MDPI
*Selections*

# Transcending Conventional Studies of Information and Computation: Dedicated to the Memory of Mark Burgin

Selected articles published by MDPI

www.mdpi.com

MDPI

# Transcending Conventional Studies of Information and Computation: Dedicated to the Memory of Mark Burgin

# Transcending Conventional Studies of Information and Computation: Dedicated to the Memory of Mark Burgin

Selected Articles Published by MDPI

MDPI • Basel • Beijing • Wuhan • Barcelona • Belgrade • Manchester • Tokyo • Cluj • Tianjin

This is a reprint of articles published online by the open access publisher MDPI (available at: www.mdpi.com). The responsibility for the book's title and preface lies with Gordana Dodig-Crnkovic and Marcin J. Schroeder, who compiled this selection.

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

LastName, A.A.; LastName, B.B.; LastName, C.C. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

© 2023 by the authors. Articles in this book are Open Access and distributed under the Creative Commons Attribution (CC BY) license, which allows users to download, copy and build upon published articles, as long as the author and publisher are properly credited, which ensures maximum dissemination and a wider impact of our publications.

# Contents

**Preface to "Transcending Conventional Studies of Information and Computation: Dedicated to the Memory of Mark Burgin"** . . . . . **ix**

**Mark Burgin**
Information Theory: a Multifaceted Model of Information
Reprinted from: *Entropy* **2003**, *5*, 146-160, doi:10.3390/e5020146 . . . . . **1**

**Mark Burgin**
Information Operators in Categorical Information Spaces
Reprinted from: *Information* **2010**, *1*, 119-152, doi:10.3390/info1020119 . . . . . **15**

**Mark Burgin**
Epistemic Information in Stratified M-Spaces
Reprinted from: *Information* **2011**, *2*, 697-726, doi:10.3390/info2040697 . . . . . **49**

**Mark Burgin**
Evolutionary Information Theory
Reprinted from: *Information* **2013**, *4*, 124-168, doi:10.3390/info4020124 . . . . . **79**

**Mark Burgin**
Weighted E-Spaces and Epistemic Information Operators
Reprinted from: *Information* **2014**, *5*, 357-388, doi:10.3390/info5020357 . . . . . **125**

**Mark Burgin and Gordana Dodig-Crnkovic**
Prolegomena to Information Taxonomy
Reprinted from: *Proceedings* **2017**, *1*, 210, doi:10.3390/IS4SI-2017-04035 . . . . . **157**

**Mark Burgin**
The General Theory of Information as a Unifying Factor for Information Studies: The Noble Eight-Fold Path
Reprinted from: *Proceedings* **2017**, *1*, 164, doi:10.3390/IS4SI-2017-04044 . . . . . **163**

**Mark Burgin and Rainer Feistel**
Structural and Symbolic Information in the Context of the General Theory of Information
Reprinted from: *Information* **2017**, *8*, 139, doi:10.3390/info8040139 . . . . . **169**

**Mark Burgin and Rainer Feistel**
Structures and Structural Information
Reprinted from: *Proceedings* **2017**, *1*, 217, doi:10.3390/IS4SI-2017-04036 . . . . . **185**

**Mark Burgin and Jaime F. Cárdenas-García**
A Dialogue Concerning the Essence and Role of Information in the World System
Reprinted from: *Information* **2020**, *11*, 406, doi:10.3390/info11090406 . . . . . **191**

**Mark Burgin**
Discerning Potential and Impact Information
Reprinted from: *Proceedings* **2020**, *47*, 4, doi:10.3390/proceedings2020047004 . . . . . **213**

**Mark Burgin and H. Roman Krzanowski**
World Structuration and Ontological Information †
Reprinted from: *Proceedings* **2022**, *81*, 93, doi:10.3390/proceedings2022081093 . . . . . **219**

**Mark Burgin**
Physical Facets of Information, Information Materialization or the Grand Illusion †
Reprinted from: *Proceedings* **2022**, *81*, 76, doi:10.3390/proceedings2022081076 . . . . . **223**

**Mark Burgin and Rao Mikkilineni**
Is Information Physical and Does It Have Mass?
Reprinted from: *Information* **2022**, *13*, 540, doi:10.3390/info13110540 . . . . . **229**

**Gordana Dodig Crnkovic and Mark Burgin**
Unconventional Algorithms: Complementarity of Axiomatics and Construction
Reprinted from: *Entropy* **2012**, *14*, 2066-2080, doi:10.3390/e14112066 . . . . . **239**

**Mark Burgin and Gordana Dodig-Crnkovic**
Prolegomena to an Operator Theory of Computation
Reprinted from: *Information* **2020**, *11*, 349, doi:10.3390/info11070349 . . . . . **255**

**Mark Burgin**
Structural Connections: Algorithms as Rituals and Rituals as Algorithms †
Reprinted from: *Proceedings* **2022**, *81*, 3, doi:10.3390/proceedings2022081003 . . . . . **271**

**Mark Burgin, Eugene Eberbach and Rao Mikkilineni**
Processing Information in the Clouds
Reprinted from: *Proceedings* **2020**, *47*, 25, doi:10.3390/proceedings2020047025 . . . . . **277**

**Rao Mikkilineni and Mark Burgin**
Structural Machines as Unconventional Knowledge Processors
Reprinted from: *Proceedings* **2020**, *47*, 26, doi:10.3390/proceedings2020047026 . . . . . **283**

**Mark Burgin and Gordana Dodig-Crnkovic**
Modeling Computing Devices and Processes by Information Operators
Reprinted from: *Proceedings* **2020**, *47*, 18, doi:10.3390/proceedings2020047018 . . . . . **289**

**Mark Burgin**
Information Processing by Symmetric Inductive Turing Machines
Reprinted from: *Proceedings* **2020**, *47*, 28, doi:10.3390/proceedings2020047028 . . . . . **295**

**Mark Burgin**
Triadic Automata and Machines as Information Transformers
Reprinted from: *Information* **2020**, *11*, 102, doi:10.3390/info11020102 . . . . . **301**

**Mark Burgin and Rao Mikkilineni**
From data Processing to Knowledge Processing: Working with Operational Schemas by Autopoietic Machines
Reprinted from: *Big Data Cogn. Comput.* **2021**, *5*, 13, doi:10.3390/bdcc5010013 . . . . . **321**

**Rao Mikkilineni and Mark Burgin**
Designing New Forms of Digital Autopoietic Machines †
Reprinted from: *Proceedings* **2022**, *81*, 59, doi:10.3390/proceedings2022081059 . . . . . **337**

**Mark Burgin and Karthik Rajagopalan**
Information Processing by Selective Machines †
Reprinted from: *Proceedings* **2022**, *81*, 122, doi:10.3390/proceedings2022081122 . . . . . **343**

**Mark Burgin**
Problem-Oriented Foundations of Intelligence in the Context of Superintelligence
Reprinted from: *Proceedings* **2020**, *47*, 21, doi:10.3390/proceedings2020047021 . . . . . **347**

**Mark Burgin**
Three Approaches to Artificial Intelligence †
Reprinted from: *Proceedings* **2022**, *81*, 147, doi:10.3390/proceedings2022081147 . . . . . **351**

**Mark Burgin and José María Díaz-Nafría**
Relational Structure of Conceptual Spaces †
Reprinted from: *Proceedings* **2021**, *81*, 39, doi:10.3390/proceedings2022081039 . . . . . **357**

**Mark Burgin**
Operations with Nested Named Sets as a Tool for Artificial Intelligence
Reprinted from: *Big Data Cogn. Comput.* **2022**, *6*, 37, doi:10.3390/bdcc6020037 . . . . . **361**

**Mark Burgin, Rao Mikkilineni and Samir Mit**
Knowledge Processing as Structure Transformation
Reprinted from: *Proceedings* **2017**, *1*, 212, doi:10.3390/IS4SI-2017-03988 . . . . . **383**

**Mark Burgin and Joseph Brenner**
Operators in Nature, Science, Technology, and Society: Mathematical, Logical, and Philosophical Issues
Reprinted from: *Philosophies* **2017**, *2*, 21, doi:10.3390/philosophies2030021 . . . . . **389**

**Mark Burgin**
Problems in the System of Scientific Knowledge
Reprinted from: *Proceedings* **2017**, *1*, 273, doi:10.3390/IS4SI-2017-03947 . . . . . **417**

**Mark Burgin**
Principles of General Ecology
Reprinted from: *Proceedings* **2017**, *1*, 148, doi:10.3390/IS4SI-2017-03996 . . . . . **423**

**Mark Burgin**
Bidirectional Named Sets as Structural Models of Interpersonal Communication
Reprinted from: *Proceedings* **2017**, *1*, 58, doi:10.3390/IS4SI-2017-03916 . . . . . **427**

**Mark Burgin and Yixin Zhong**
Information Ecology in the Context of General Ecology
Reprinted from: *Information* **2018**, *9*, 57, doi:10.3390/info9030057 . . . . . **433**

**Mark Burgin and Kees (C.N.J.) de Vey Mestdagh**
Complexity of Legal Processes and Systems
Reprinted from: *Proceedings* **2020**, *47*, 17, doi:10.3390/proceedings2020047017 . . . . . **447**

**Arkadiy Dantsker and Mark Burgin**
Monitoring Thermal Conditions and Finding Sources of Overheating †
Reprinted from: *Proceedings* **2021**, *81*, 38, doi:10.3390/proceedings2022081038 . . . . . **453**

**Mark Burgin and H. Paul Zellweger**
Named Sets as an Efficient Tool for Modeling Data Relationships in Database Models †
Reprinted from: *Proceedings* **2022**, *81*, 51, doi:10.3390/proceedings2022081051 . . . . . **457**

**Mark Burgin and Rao Mikkilineni**
Symbiotic Information Processing and Technological Progress †
Reprinted from: *Proceedings* **2022**, *81*, 84, doi:10.3390/proceedings2022081084 . . . . . **463**

**Mark Burgin and Arkadiy Dantsker**
Pollution Information Acquisition and Discovering Pollution Sources †
Reprinted from: *Proceedings* **2022**, *81*, 74, doi:10.3390/proceedings2022081074 . . . . . **469**

**Mark Burgin**
Symposium Theoretical Information Studies
Reprinted from: *Proceedings* **2017**, *1*, 223, doi:10.3390/proceedings1030223 . . . . . **473**

**Marcin J. Schroeder, GordanaDodig-Crnkovic and Mark Burgin**
IS4SI SUMMIT Berkeley 2019
Reprinted from: *Proceedings* **2020**, *47*, 1, doi:10.3390/proceedings2020047001 . . . . . **477**

**Mark Burgin**
Conference Theoretical Information Studies Berkeley 2019
Reprinted from: *Proceedings* **2020**, *47*, 2, doi:10.3390/proceedings2020047002 . . . . . **479**

**Gordana Dodig-Crnkovic and Mark Burgin**
Recent Books Delineating the Emergent Academic Filed of the Study of Information
Reprinted from: *Proceedings* **2020**, *47*, 6, doi:10.3390/proceedings2020047006 . . . . . **485**

**Marcin J. Schroeder and Mark Burgin**
2021 Summit of the International Society for the Study of Information (IS4SI) †
Reprinted from: *Proceedings* **2022**, *81*, 1, doi:10.3390/proceedings2022081001 . . . . . **489**

**Mark Burgin**
Conference Theoretical and Foundational Problems in Information Studies †
Reprinted from: *Proceedings* **2022**, *81*, 32, doi:10.3390/proceedings2022081032 . . . . . **497**

# Preface to "Transcending Conventional Studies of Information and Computation: Dedicated to the Memory of Mark Burgin"

This book stands as a tribute to the memory of our late colleague, Professor Mark Burgin, whose recent departure has left a profound void in our academic community. Within the pages of this volume, we present a collection of reprints featuring Professor Burgin's articles, previously published in MDPI venues, including journals Information, Entropy, Philosophies, Big Data and Cognitive Computing, and Proceedings.

Mark Burgin was a highly respected figure in the academic world, renowned for his exceptional contributions as a mathematician, computer scientist, and information theoretician. His profound impact resonated across a diverse spectrum of fields, encompassing the domains of information and computation, foundations of mathematics, philosophy of information and computation, theory of knowledge, and logic.

Professor Burgin's intellectual prowess and groundbreaking research have left an indelible mark on numerous scholarly endeavors, offering novel insights into information-based phenomena. Through his interdisciplinary approach, he bridged gaps between disparate disciplines, instigating fresh connections, approaches, and applications that have profoundly enriched the advancement of these specialized domains.

With this volume, we pay homage to the enduring legacy of Mark Burgin, commemorating his remarkable career and intellectual achievements. It is our aspiration that this compendium shall serve as a testament to his invaluable contributions and will inspire future generations of scholars to engage in rigorous research within the very fields that were enriched by his exceptional scholarship.

**Gordana Dodig-Crnkovic and Marcin J. Schroeder**

*Entropy* **2003**, *5*, 146-160

**Entropy**
**ISSN 1099-4300**
© 2003 by MDPI
www.mdpi.org/entropy

# Information Theory: a Multifaceted Model of Information

**Mark Burgin**

Department of Mathematics, University of California, Los Angeles, 405 Hilgard Ave. Los Angeles, CA 90095. E-mail: mburgin@math.ucla.edu

*Received: 10 January 2003 / Accepted: 12 February 2003 / Published: 30 June 2003*

---

**Abstract:** A contradictory and paradoxical situation that currently exists in information studies can be improved by the introduction of a new information approach, which is called the general theory of information. The main achievement of the general theory of information is explication of a relevant and adequate definition of information. This theory is built as a system of two classes of principles (ontological and sociological) and their consequences. Axiological principles, which explain how to measure and evaluate information and information processes, are presented in the second section of this paper. These principles systematize and unify different approaches, existing as well as possible, to construction and utilization of information measures. Examples of such measures are given by Shannon's quantity of information, algorithmic quantity of information or volume of information. It is demonstrated that all other known directions of information theory may be treated inside general theory of information as its particular cases.

---

## 1. Introduction

As with any natural phenomenon, there are two main problems related to information. The first one is to define what information is and to find what basic properties it has. The second problem is how to measure and evaluate information. In this paper, we consider the second problem.

From the beginning of the development of information theory, it was known more how to measure information than what information is. Hartley and Shannon gave effective formulas for measuring the quantity of information. However, without understanding the phenomenon of information, these formulas bring misleading results when applied to irrelevant domains.

At the same time, a variety of information definitions have been introduced. Being mostly vague and limited, these definitions have brought confusion into information studies (cf., for example, [34, 4, 19, 39]).

The existing confusion with the term information is increased when researchers call by the name "information" a measure of information or even a value of such a measure. For example, many call by the name "information" Shannon's quantity of information $\boldsymbol{I} = -\Sigma_{i=1}^{n} p_i \log p_i$ or Renyi's measure of information $\boldsymbol{H}_\alpha(X) = (1 - \alpha)^{-1}\log \Sigma_{x \in X} P^\alpha(x)$ ([29, 15]) or pragmatic measure of information $\boldsymbol{I}_M(p, q) = \Sigma_{i,m}\, p_{i/m}\, \varphi_m \log_2 (p_{i/m}/q_i)$ ([37]). At the same time, some researchers (cf., for example, [30] or [21]) never did this. That is why it is so important to explain and understand distinctions between some phenomena and their measures. This completely refers to information.

Even if we have an answer to the question what is information, it is not sufficient for practical purposes of information processing. The main problem in this perspective is how to measure or, at least, to evaluate information. The results of [8], which describe information as a general phenomenon and its basic properties by means of ontological principles, provide a base for developing a unified theory for information evaluation and measurement. This is done in the second section of the paper, which goes after Introduction. It contains the axiological component of the general theory of information. This component is developed on the base of axiomatic methodology, providing basic axiological principles for information evaluation and measurement. Basic axiological principles explain what are basic properties of measures and estimates for information. These principles systematize and unify different approaches, existing as well as possible, to construction and utilization of information measures. This, axiological aspect of the theory is not less important than the ontological one because methods of modern science emphasize importance of measurement and evaluation that are technical tools for observation and experiment in scince, as well as for engineering. In the third section of this paper, it demonstrated how the main directions of information theory as a whole are unified and systemized in the context of the general theory of information.

## 2. Basic Axiological Principles of the General Theory of Information

Basic axiological principles explain how to evaluate information and what measures of information are necessary.

According to the ontological principles [8, 11, 10], information causes changes either in the whole system ***R*** that receives information or in an inforlogical subsystem IF(***R***) of this system. Consequently, it is natural to assume that measure of information is determined by the results that are caused by reception of the corresponding portion of information. It is formulated in the first principle.

**Axiological Principle A1.** *A measure of information **I** for a system **R** is some measure of changes caused by **I** in **R*** (*for information in the strict sense, in the infological system* IF(***R***) ) .

Next principles describe what information measures reflect. This implies several classifications for information measures.

The first criterion for measure classification is the time of changes.

**Axiological Principle A2.** *According to time orientation, there are three temporal types of measures of information*: 1) *potential or perspective*; 2) *existential or synchronic*; 3) *actual or retrospective.*

**Definition 4.1.** Potential or perspective measures of information ***I*** determine (reflects) what changes (namely, their extent) may be caused by ***I*** in ***R***.

**Definition 4.2.** Existential or synchronic measures of information ***I*** determine (reflects) what changes (namely, their extent) are going in ***R*** during some fixed interval of time after receiving ***I***. This interval of time may be considered as the present time.

**Definition 4.3.** Actual or retrospective measures of information **I** determine (reflects) what changes (namely, their extent) were actually caused by ***I*** in the system ***R***.

Let us consider the following example. Some student ***R*** studies a textbook ***C***. After two semesters she acquires no new knowledge from ***C*** and finishes to use it. At this time an existential measure of information contained in ***C*** for ***R*** is equal to zero. The actual measure of information in ***C*** for ***R*** is very big if ***R*** is a good student and ***C*** is a good textbook. But the potential measure of information in ***C*** for ***R*** may be also bigger than zero if in future ***R*** returns to ***C*** and finds in ***C*** such things that she did not understand in her youth.

Different types of information measures can estimate information in separate infological systems. For example, synchronic measures reflect the changes of the short-term memory, while retrospective measures represent transformations in the long-term memory of a human being.

The second criterion for measure classification is derived from the system separation triad:

$$(\boldsymbol{R}, \boldsymbol{l}, \boldsymbol{W})$$

Here, ***R*** is a system, ***W*** is the environment of this system and ***l*** represent different links between ***R*** and ***W***.

**Axiological Principle A3.** *There are three structural types of measures of information: external, intermediate, and internal.*

**Definition 4.4.** An *internal* information measure reflects the extent of inner changes caused by ***I*** . Examples are given by the change of the length (the extent) of a thesaurus.

**Definition 4.5.** An *intermediate* information measure reflects the extent of changes caused by ***I*** in the links between ***R*** and ***W***.

Examples are given by the change of the probability p(***R***, *g*) of achievement of a particular goal *g* by the system ***R***. This information measure was suggested by [20] and is called the quality of information.

**Definition 4.6.** An *external* information measure reflects the extent of outer changes caused by ***I***, i.e., the extent of changes in ***W***.

Examples are given by the change of the dynamics (functioning, behavior) of the system ***R*** or by the complexity of changing ***R***.

**Axiological Principle A4.** *There are three constructive types of measures of information: abstract, realistic, and experiential.*

**Definition 4.7.** An *abstract* information measure is determined theoretically under general assumptions.

Examples are given by the change of the length (the extent) of a thesaurus.

**Definition 4.8.** A *realistic* information measure is determined theoretically subject to realistic conditions.

Quality of information [20] is an example of such measure.

Those people who worked with information technology and dealt with problems of information security and reliability have discovered the difference between abstract and realistic measures of information. They found that if you have an encrypted message, you know that information contained in this message is available. Those who know the cipher can get it. However, if you do not possess this cipher and do not have working algorithms for deciphering, then this information is inaccessible to you. To reflect this situation, exact concepts of available and *acceptable* information have been introduced. Available information is measured by abstract information measures, while acceptable information is measured by realistic information measures.

The third type of measures from the Principle A4 is defined as follows.

**Definition 4.9.** An *experiential* information measure is obtained through experimentation.

**Remark 4.1.** In some cases, one information measure may belong to different types. In other words, classes of information measures overlap.

As an example of such a measure, we may take the measure that is used to estimate the computer memory content as well as the extent of a free memory in computer or on the disk. Namely, information in computers is represented as strings of binary symbols and the measure of such a string is the number $n$ of these symbols. The length of the string is taken as the value of its information measure. The unit of such a measure is called a bit. Computer memory is measured in bits, bytes, kilobytes, which contain 1,000 bytes, megabytes, which contain 1,000,000 bytes and so on. This reflects the length of the strings of symbols can be stored in a memory. This is the simplest measure of symbolic information. However, this measure is necessary because storage devices (such as computer disks) have to be relevant to needs in information storage. For example, if you have a file containing five megabytes and a floppy disk of 1.4 megabytes, then cannot store this file on this floppy disk.

Moreover, some authors consider information in such a simplistic way. For example, in one article, it is assumed that information is a one-dimensional *string* comprising a sequence of atomic *symbols*. Each such symbol is a token drawn, with replacement, from a set of available symbol *types*. Sets of symbol types may be the binary digits (0 and 1) or the alphanumeric characters or some other convenient set.

It is necessary to remark that different information measures may give different values for the same string. For example, according to the measures used in the algorithmic information theory, the algorithmic measure of this string having length $n$ may be much less that $n$ (cf., [21, 13]).

Let us look how this measure relates to the axiological principles of the general theory of information. When such a string is written into the computer memory, it means that some information is stored in the memory. Changes in the memory content might be measured in a different way. The

simplest is to measure the work that has been performed when the string has been written. The simplest way to do this is to count how many elementary actions of writing unit symbols have been performed. However, this number is just the number of bits in this string. So, conventional measure of the size of a memory and its information content correlates with the axiological principles of the general theory of information.

Let us take classifications of measures that are presented in the axiological principles A2-A4 and apply it to the conventional measure of the size of a memory. We see that it is an internal measure (cf. Principle A3), both abstract and realistic measure (cf. Principle A4), and belong to all three classes of potential, existential and actual measures (cf. Principle A2).

The axiological principles A2-A4 have the following consequences.

*A unique measure of information exists only for oversimplified system.* Any complex system ***R*** with a developed infological subsystem IF(***R***) has many parameters that may be changed. So, such systems demand many different measures of information in order to reflect the full variety of these systems properties as well as of conditions in which these systems function. Thus, *the problem of finding one universal measure for information is unrealistic.*

Uncertainty elimination (which is measured by the Shannon's quantity of information, cf. Section 3) is only one of the possible changes, which are useful to measure for information. Another important property is a possibility to obtain a better solution of a problem (which is more complete, more adequate, demands less resources, for example, time, for achievement a goal). Changes of this possibility reflect the utility of information. Different kinds of such measures of information are introduced in the theory of information utility [20] and in the algorithmic approach in the theory of information [12, 13, 21].

**Axiological Principle A5.** *Measure of information **I**, which is transmitted from **C** to a system **R**, depends on interaction between **C** and **R**.*

Stone [33] gives an interesting example of this property. Distortions of human voice, on one hand, are tolerable in an extremely wide spectrum, but on the other hand, even small amounts of distortion create changes in interactive styles.

The next principle is a clarification of Principle A4.

**Axiological Principle A6.** *Measure of information transmission reflects a relation (like ratio, difference etc.) between measures of information that is accepted by the system **R** in the process of transmission and information that is presented by **C** in the same process.*

It is known that the receiver accepts not all information that is transmitted by a sender. Besides, there are different distortions of transmitted information. For example, there is a myth that the intended understanding may be transmitted whole from a sender to a receiver. In almost every process of information transmission the characteristic attitudes of the receiver "interfere" in the process of comprehension. People make things meaningful for themselves by fitting them into their preconceptions. Ideas come to us raw, and we dress and cook them. The standard term for this process is *selective perception*. We see what we wish to see, and we twist messages around to suit ourselves. All this is demonstrated explicitly in the well-known 'Mr. Biggott' studies [35]. An audience was

shown a series of anti-prejudice cartoons featuring the highly prejudiced Mr. Biggott. Then people from the audience were subjected to detailed interviews. The main result was that about two thirds of the sample clearly misunderstood the anti-prejudice intention of the cartoons. The major factors accounting for this selective perception, according to the researchers, were the predispositions of the audience. Those who were already prejudiced saw the cartoons as supporting their position. Even those from them who understood the intentions of the cartoons found ways of evading the anti-prejudice 'effect.' Only those with a predisposition toward the message interpreted the films in line with the intended meanings of the communicators.

## 3. Systematizing Theoretical Approaches in Information Science

Other developed theoretical approaches in information science are particular cases of the general theory of information because they explicitly or implicitly consider treat information from the functional point of view as a kind of transformations in a system. To prove this, we find in each case a relevant infological system IF(***R***) and demonstrate that in each of these approaches information is what changes this system.

It is necessary to remark that there are some approaches, which consider information as some kind of knowledge. More exactly, in modern information theory, according to [17], a distinction is made between structural-attributive and functional-cybernetic types of theories. While representatives of the former approach conceive information as structure, like knowledge or data, variety, order, and so on; members of the latter understand information as functionality, functional meaning or as a property of organized systems. The general theory of information treats information from the functional and, more exactly, dynamic perspective. As it is demonstrated in [10, 11], structural-attributive interpretation does not represent information itself but relates to information carriers. Consequently, structural-attributive types of information theories are also included in the scope of the general theory of information because structures and attributes are represented in this theory by infological elements and their properties and systems.

### 3.1. Shannon's Information Theory

The **statistical** approach is now the most popular direction in the information sciences. It is traditionally called Shannon's information theory or, as it was at first named by Shannon, the theory of communication [30]. It is a mathematical theory formulated principally by the American scientist Claude E. Shannon to explain aspects and problems of information and communication.

The basic problem that needed to be solved, according to Shannon, was the reproduction at one point of a message produced at another point. He deliberately excluded from his investigation the question of the meaning of a message, i.e., the reference of the message to the things of the real world. He wrote [30]:

"*Frequently the messages have meaning; that is they refer to or are correlated according to some system with physical or conceptual entities. These semantic aspects of communication are irrelevant to*

*the engineering problem. The significant aspect is that the actual message is one selected from a set of possible messages. The system must be designed to operate for each possible selection, not just the one which will actually be chosen since this is unknown at the time of design*."

While the statistical theory of information is not specific in many respects, it proves the existence of optimum coding schemes without showing how to find them. For example, it succeeds remarkably in outlining the engineering requirements of communication systems and the limitations of such systems.

In the statistical theory of information, the term information is used in a special sense: it is a measure of the freedom of choice with which a message is selected from the set of all possible messages. Information is thus distinct from meaning, since it is entirely possible for a string of nonsense words and a meaningful sentence to be equivalent with respect to information content.

In the statistical theory of information, information is measured in bits (short for *binary digit*). One bit is equivalent to the choice between two equally likely choices. For example, if we know that a coin is to be tossed but are unable to see it as it falls, a message telling whether the coin came up heads or tails gives us one bit of information. A special measure that is called *quantity of information* of a message about some situation is defined by the formula $\boldsymbol{I} = -\Sigma_{i=1}^{n} p_i \log p_i$ where $n$ is the number of possible cases of the situation in question and $p_i$ is the probability of the case $i$ . So, information is considered as elimination of uncertainty, i.e., as a definite change in the knowledge system that is the infological system IF($\boldsymbol{R}$) of the receptor of information. Consequently, we have the following result.

**Proposition 3.1.** *The statistical information theory is a subtheory the general theory of information.*

Interestingly, the mathematical expression for information content closely resembles the expression for entropy in thermodynamics. The greater the information in a message, the lower its randomness, or "noisiness," and hence the smaller its entropy. Since the information content is, in general, associated with a source that generates messages, it is often called the entropy of the source.

However, many saw limitations of Shannon's information theory, especially when it was applied outside technical areas. As a result, other directions have been suggested in information science. However, to this day, there is no measure in information theory that is as well-supported and as generally accepted as Shannon's quantity of information. On the other hand, Shannon's work is rightly seen as lacking indications for a conceptual clarification of information.

### 3.2. A Semantic Theory of Information

The semantic theory of information tries to encompass semantic aspects of information because some experts in information sciences consider meaning as the essence of information. This approach is based on the assumption (cf., for example, [2]), that every piece of information has the characteristic that it makes a positive assertion and at the same time makes a denial of the opposite of that assertion. In the semantic information theory**,** transformations of such infological system as thesaurus, or system of knowledge, are treated as information. The founders of this approach, Bar-Hillel and Carnap [1] build it as a logical system. Thus, the infological systems in this theory are sets of logical propositions. These proposition, are used to describe and represent the state of an arbitrary system. The corresponding measure of information is defined for a separate proposition as the probability that this

proposition is a part of the description of the real state of a system under consideration. Thus, information causes change in knowledge about this state. Consequently, we have the following result.

**Proposition 3.2.** *The semantic information theory is a subtheory the general theory of information.*

A similar concept of information is utilized in the approach that was developed by Shreider [31] and also called the **semantic theory of information**. In his theory, the notion of a thesaurus is more imprecise. At the same time, the notion of information is more general than in [1] being a transformation of a thesaurus. As any thesaurus is a kind of infological systems, this approach is also included in the general theory of information.

### 3.3. Fisher Information

Fisher information is also based on mathematical statistics. It was introduced by Ronald A. Fisher, who developed classical measurement theory [16]. According to this theory, the quality of any measurement may be specified by a form of information that has come to be called Fisher information.

Consider the class of "unbiased" estimates, obeying the law $<\hat{\theta}(y)> = \theta$, where $\theta$ is a value of a given parameter and $\hat{\theta}(y)$ is an estimate of $\theta$, which is an optimal function for a measured data $y = \theta + x$. The mean-square error $e^2$ in such an estimate obeys a relation $e^2 \mathrm{I} \geq 1$. Here I is called the Fisher information. In particular case, which is important for physics [18], we have

$$\mathrm{I} = \int p'^{2}(x) / p(x)\, \mathrm{d}x$$

Here $p(x)$ denotes the probability density function for the noise value $x$, and $p'(x) = \mathrm{d}p(x)/\mathrm{d}x$.

This demonstrates that Fisher information I is the quality metric for the estimation/measurement procedure. Consequently, I measures an adequate change in knowledge about the parameter $\theta$, and is information in the sense of the general theory of information. Thus, we have the following result.

**Proposition 3.3.** *The Fisher information theory is a subtheory the general theory of information.*

It is necessary to remark that there is an approach in physics that is based on the assumption that information has to be the base for the whole physics because physics is knowledge about the universe and all knowledge is acquired through information [18]. By this theory, the observer is included into the phenomenon of measurement. According to the classical quantum mechanical philosophy, the observer becomes a collector of data, which are influenced by the measurement. More radical ideas, which are stated, for example, by J.A. Wheeler [36], represent the observer as an activator of the physical phenomenon that gives rise to the data.

### 3.4. The Qualitative Information Theory

In the qualitative information theory [28], the following definition is given: *information is a transformation of one communication of an information association into another communication of the same association*. So, here the infological system IF(***R***) is some information association and information is a transformation of ***T***. Consequently, we have the following result.

**Proposition 3.4.** *The qualitative information theory is a subtheory the general theory of information.*

### 3.5. The Algorithmic Theory of Information

The principal aim of the algorithmic theory of information is the development of more realistic concepts of randomness and probability than those that are in the traditional theory of probability. This new approach was suggested in the works of three authors: Solomonoff [32], Kolmogorov [21], and Chaitin [13]. The main idea was that algorithms play a leading role in many processes, including people behavior. Consequently, theory of algorithms has to be central in a study of such processes.

In the algorithmic information theory, there are two kinds of measures of information [32, 21, 12, 5, 7]. The first one is called the entropy, or information content, or complexity of finite constructive objects (like strings of symbols). Such algorithmic measure of information is defined to be the number of bits (or symbols) needed to specify the object in question so effectively that it can be constructed. This measure is called the Kolmogorov complexity of $x$ and is denoted by $\boldsymbol{K}(x)$. By the definition, this is the measure of such a transformation as a construction of some object. It is an external measure of information that acts on such infological system IF($\boldsymbol{R}$) as thesaurus, or the system of knowledge. Knowledge in this system is represented by strings of symbols (texts) that have meaning.

It is possible to define a more general measure of information of the first type. It is called a generalized Kolmogorov complexity measure [5, 7]. Generalized Kolmogorov complexity measures reflects a measure of resources needed to construct the object in question. According to [5], generalized Kolmogorov complexity measures are dual to computational complexity measures.

To get the second algorithmic measure of information, we consider two texts or simply, sequences of symbols $x$ and $y$. Then in the algorithmic information theory, the second, relative measure of information is defined as information in $y$ about $x$. This measure is given by the formula $\boldsymbol{I}(y, x) = \boldsymbol{K}(x) - \boldsymbol{K}(x/y)$ where $\boldsymbol{K}(x)$ is the Kolmogorov complexity of $x$ and $\boldsymbol{K}(x/y)$ is the Kolmogorov complexity of construction of $x$ when $y$ is given. The Kolmogorov complexity $\boldsymbol{K}(x)$ or other dual measures of complexity studied in [6] may be also treated as absolute measures of information which are necessary for constructing some object. So, information in $y$ about $x$ changes the system of algorithms that are used for the computation (or construction) of $x$. In this case, the infological system IF($\boldsymbol{R}$) that is changed by the information is the system of algorithms which compute (or construct) $x$. Consequently, we have the following result.

**Proposition 3.5.** *The algorithmic information theory is a subtheory the general theory of information, and* $\boldsymbol{I}(y, x)$ *is an internal measure of information.*

### 3.6. The Pragmatic Theory of Information

The concept of pragmatic information was introduced by Ernst Ulrich von Weizsäcker [38] and further developed by Von Lucadou [24, 25], Kornwachs [22], Weinberger [37] and some others. This concept relies on the two notions: firsteness ("Erstmaligkeit") or originality or novelty, and confirmation ("Bestatigung") or redundancy to already known. Weizsäcker suggests that a message that does nothing but confirm the prior knowledge of a receiver will not change its structure or behaviour. Thus, with confirmation up to 100%, a message gives no pragmatic information. On the other hand, a message providing only original/novel material completely unrelated to any prior

knowledge also will not change structure or behaviour of the receiver, because the receiver will not understand this message. Thus, with firstness up to 100%, a message gives no pragmatic information. Only a relevant mixture of firstness and confirmation allows the receiver to get pragmatic information from the message.

Thus, pragmatic information, according to Weizsäcker, is related to changes of structure or behaviour of the receiver. Thus, the theory of pragmatic information is a special case of the general theory of information. Taking thesaurus, i.e., the system of knowledge, as an infological system, adding some principles and developing them further, we obtain the theory of pragmatic information.

The psychologist von Lucadou [24, 25] applied pragmatic information to problems of psychology. He writes:

"The model of pragmatic information (MPI), which is a candidate for a non-classical model of psychology, predicts that the behavior of a non-classical system depends on the conditions of its observation. The exchanged pragmatic information (meaningful information) ties the "observer" (e.g., a person) and the "observed" (e.g., a machine) together and creates an "organizational closed system." It is assumed that this process produces non-local correlations between the observer and the observed."

In such a way, pragmatic information acts on the observed system. This entirely corresponds to the concept of information in the general theory of information.

However, the general theory of information allows us to go further and to consider two opportunities of change. In one case, behavior of the system under observation changes. It corresponds to an action on the behavioral infological system. In the other case, changes only information that we get from the system under observation. It corresponds to an action on the representational infological system.

### 3.7. Social Information

Goguen (1997) suggests to study first of all social information because it is very important to the development of information processing systems. He introduces the following definition:

*An item of information is an interpretation of a configuration of signs for which members of some social group are accountable.*

According to the general theory of information, a configuration of signs is a carrier of information, while interpretation as an action is the change caused in some social group. At the same time, interpretation as some text (symbolic configuration) is an element of the cognitive infological system of this group.

### 3.8. The Utility Theory of Information

In the utility theory of information [20], the measure of information is called the quality of information. It is defined for mission oriented systems $\boldsymbol{R}$ . If $\boldsymbol{I}$ is some portion of information, then the quality of this information is equal to the caused by $\boldsymbol{I}$ change of the probability $p(\boldsymbol{R},g)$ of achievement of a particular goal $g$ by the system $\boldsymbol{R}$ . If we consider objective probability, then the corresponding infological system is the state space of the world in which system $\boldsymbol{R}$ functions. If we consider

subjective probability, then the corresponding infological system is the belief space in which probabilities for different events involving system ***R*** are represented. In both cases, information appears as a change in the corresponding infological system. Consequently, we have the following result.

**Proposition 3.6.** *The utility information theory is a subtheory the general theory of information.*

### 3.9. An Economic Theory of Information

The economic theory of information [26, 27] appeared aiming to represent economical aspects information processes in society. It reflected some changes in the thinking habits of economists that has resulted in a broadening of the concept of `economics'. As time advanced, problems of decision, information and organization came to the center of many economical theories. The actions considered and their outcomes may, but need not, be inputs and outputs of quantifiable and marketable production factors and products, or their prices. Nowadays, broader `optimization problems' are pigeonholed as belonging to `operations research', `management science', or `systems analysis'. They occupy economists, engineers, and (to the extent that applied probabilities are involved) statisticians as well. As `benefit-cost analysis', these tools are also applied to problems of social policy no less than to military or medical planning. There is a promise of cross-fertilization with the evolutionary theories of life science and anthropology. Moreover, the decision-theoretic approach has recently taken a foothold in the philosophy of science.

In the economic information theory, information $x$ about the state of environment is considered with respect to a person's action $a$. Person's profit $u(a, x)$ is taken as a utility function. It makes possible to consider expectation $U_0 = \max_a \mathrm{E}\, u(a, x)$ of the profit without knowing $x$ as well as expectation $U_1 = \mathrm{E} \max_a u(a, x)$ of the profit when $x$ is known. Then the value of information $v(I_x)$ about $x$ is defined as $v(I_x) = U_1 - U_0$ . Thus, information is also treated as a change in the system of knowledge that is the infological system IF(***R***) in this case. Consequently, we have the following result.

**Proposition 3.7.** *The economic information theory is a subtheory the general theory of information.*

According to the principles of general theory of information, while in Shannon's theory quantity of information is an internal measure, value of information in the sense of the economic information theory is an external measure of information.

### 3.10. A Dynamic Theory of Information

The principal aim for the development of the dynamic theory of information was investigation of biological information and information processes in living systems [14]. In the dynamic information theory, the notion of information is taken as a basic one. Information is considered as a sequence of two operations: choice of one alternative from a collection of possible alternatives and saving the chosen alternative. Thus, information is also treated as an action causing transformations. Consequently, we have the following result.

**Proposition 3.8.** *The dynamic information theory is a subtheory the general theory of information.*

## Conclusion

Thus, we have demonstrated that the general theory of information makes it possible to solve many open problems related to information and achieve a new profound understanding of information as a natural, technological, and social phenomenon. It provides for systematization of all other approaches in information sciences, eliminating many of their shortcomings. In particular, a new, more adequate definition of information is obtained. In addition, it allows us to discover new types of information: *cognitive*, *conditioning*, and *regulative* information. Cognitive information gives knowledge and thus, it is what people know under the name of information. Two other types are new and help to understand many phenomena in system functioning. This discovery of new types of information makes possible to determine the role of information for system functioning. Conditioning information is basic for any system from the perspective of its inner evolution. Regulative information is basic for any system from the perspective of its interaction with environment. Cognitive information appears only on higher levels of system development. The role of cognitive information increases with the development of a system and becomes decisive on some level of this development.

## References


1. Bar-Hillel, Y. and Carnap, R. Semantic Information, *British J. of Philosophical Sciences* **1958**, *4(3)*, 147-157.
2. Bateson, G. and Ruesch, J. *Communication*: *The Social Matrix of Psychiatry*. W.W.Norton and Co., New York, **1951**.
3. Beck, J. *Cognitive Therapy: Basics and Beyond*. New York, Guilford, **1995**.
4. Bosma, H. Information Quality rather then Information Quantity, in "Information Policy and Scientific Research", Elsevier, Amsterdam, **1983**, 99-106.
5. Burgin, M.S. Generalized Kolmogorov complexity and duality in theory of computations. *Notices of the Russian Academy of Sciences* **1982**, *25(3)*, 19-23.
6. Burgin, M. S. The Generalized Kolmogorov Complexity and other Dual Complexity Measures. *Cybernetics* **1990**, *4*, 21-29 (translated from Russian) .
7. Burgin, M.S. The Algorithmic Approach in the Dynamic Theory of Information. *Notices of the Russian Academy of Sciences*, **1995**, *342(1)*, 7-10.
8. Burgin, M.S. Fundamental Structures of Knowledge and Information. Academy for Information Sciences, Kiev, **1997** (in Russian).
9. Burgin, M.S. Information Algebras, Control Systems and Machines, **1997**, *6*, 5-16 (in Russian).
10. Burgin, M. Information in the Context of Education. *The Journal of Interdisciplinary Studies* **2001**, *14(Fall)*, 155-166.
11. Burgin, M. Information, Organization, and System Functioning. *Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics* **2002**, *2*, 155-160.
12. Chaitin, G.J. On the Length of Programs for Computing Finite Binary Sequences. *J. Association for Computing Machinery* **1966**, *13(4)*, 547-569.

13. Chaitin, G.J. Algorithmic information theory. *IBM Journal of Research and Development* **1977**, *21(4)*, 350-359.
14. Chernavsky, D.S. *Synergetics and Information*, Znaniye, Moscow, **1990** (in Russian).
15. Crutchfield, J.P. Information and its Metric, in *Nonlinear Structures in Physical Systems – Pattern Formation, Chaos, and Waves*; Springer-Verlag, New York, **1990**, 119-130.
16. Fisher, R. A. *Contributions to Mathematical Statistics*; New York, Wiley, **1950**.
17. Flükiger, D.F. *Contributions towards a Unified Concept of Information*, Doctoral Thesis, University of Berne, **1995**.
18. Frieden, R.B. *Physics from Fisher Information*; Cambridge University Press, Cambridge, **1998**.
19. Giuliano, V.E. The United States of America in the Information Age*,* in "*Information Policy and Scientific Research*", Elsevier, Amsterdam, **1983**, 59-76.
20. Harkevich, A. On the value of information. *Problems of Cybernetics* **1960**, *4*, 2-10.
21. Kolmogorov, A.N. Three approaches to the definition of the quantity of information. *Problems of Information Transmission* **1965**, *1*, 3-11.
22. Kornwachs, K. Pragmatic Information and the Generation of Knowledge, in Interdisciplinary Approaches to a New Understanding of Cognition and Consciousness, Ulmer Universitätsverlag, Ulm, **1996**, 33-74.
23. Laudon, K.C. *Information Technology and Society*; Wadsworth P.C., Belmont, California, **1996**.
24. Von Lucadou, W. The Model of Pragmatic Information. *Proceedings of the 30th Parapsychological Association Convention* **1987**, 236-254.
25. Von Lucadou, W. The Endo- Exo-Perspective - Heaven and Hell of Parapsychology. *Proceedings of the 37th Parapsychological Association Convention* **1994**, 242-252.
26. Marschak, J. Economic information, decision, and prediction. *Selected essays*, **1980**, *II(II)*. *Economics of information and organization. Theory and Decision Library*, v. 7. D. Reidel P.C., Dordrecht/Boston, Mass.
27. Marschak, J. Economics of information systems. *J. Amer. Statist. Assoc*. **1971**, *66*, 192-219.
28. Mazur, M. *Jakosciowa Teoria Informacji*, Warszawa, PAN , **1970** (in Polish).
29. Renyi, A. Some Fundamental Questions of Information Theory, in *Selected Papers of Alfred Renyi*, v. 2 Akademii Kiado, Budapest, **1976**.
30. Shannon, C.E. The Mathematical Theory of Communication. *Bell System Technical Journal* **1948**, *27(1)*, 379-423, *(3)*, 623-656.
31. Shreider, Yu.A. On Semantic Aspects of Information Theory. *Information and Cybernetics*, Moscow, Radio, **1967** (in Russian).
32. Solomonoff, R. A Formal Theory of Inductive Inference. *Information and Control* **1964**, *7(1)*, 1-22.
33. Stone, A.R. *The War of Desire and Technology at the Close of the Mechanical Age*; The MIT Press, Cambridge, Massachussetts, **1997**.
34. Tohm, R. *Structural Stability and Morphogenesis,* Benjamin, New York, **1975**.

35. Tudor, A. *Image and Influence*; Geoge Allen&Unwin Ltd, London, **1950**.
36. Wheeler, J.A. Information, Physics, Quantum: The Search for Links. In "*Complexity, Entropy, and the Physics of Information*" Redwood City, CA: Addison-Wesley, **1990**.
37. Weinberger, E.D. A Theory of Pragmatic Information and Its Application to the Quasispecies Model of Biological Evolution, nlin.AO/0105030 (electronic edition: http://arXiv.org), **2001**.
38. von Weizsäcker, E.U. Erstmaligkeit und Bestatigung als Komponenten der Pragmatischen Information, im *Offene Systeme* I, *Beitrage zur Zeitstruktur von Information, Entropie und Evolution*, Klett, Stuttgart, **1974**, 82-113.
39. Wilson, T. Trends and Issues in Information Science – a General Survey, in "*Media, Knowledge and Power,*" Groom Helm, London/Sydney, **1993**, 407-422.

© 2003 by MDPI (http://www.mdpi.org). Reproduction for noncommercial purposes permitted.

*Information* **2010**, *1*, 119-152; doi:10.3390/info1020119

OPEN ACCESS

*information*

**ISSN 2075-1710**

www.mdpi.com/journal/information

*Article*

# Information Operators in Categorical Information Spaces

**Mark Burgin**

Department of Mathematics, University of California, Los Angeles, 405 Hilgard Ave., Los Angeles, CA 90095, USA; E-Mail: mburgin@math.ucla.edu

*Received: 4 October 2010; in revised form: 28 October 2010 / Accepted: 29 October 2010 / Published: 18 November 2010*

---

**Abstract:** The general theory of information (GTI) is a synthetic approach, which reveals the essence of information, organizing and encompassing all main directions in information theory. On the methodological level, it is formulated as system of principles explaining what information is and how to measure information. The goal of this paper is the further development of a mathematical stratum of the general theory of information based on category theory. Abstract categories allow us to construct flexible models for information and its flow. Now category theory is also used as unifying framework for physics, biology, topology, and logic, as well as for the whole mathematics, providing a base for analyzing physical and information systems and processes by means of categorical structures and methods. There are two types of representation of information dynamics, *i.e.*, regularities of information processes, in categories: the categorical representation and functorial representation. Here we study the categorical representations of information dynamics, which preserve internal structures of information spaces associated with infological systems as their state/phase spaces. Various relations between information operators are introduced and studied in this paper. These relations describe intrinsic features of information, such as decomposition and complementarity of information, reflecting regularities of information processes.

**Keywords:** information; information operator; category; functor; composition; knowledge; cognition; order

---

## 1. Introduction

Modern science, and especially, physics, is extensively based on mathematics. Mathematical models give the most exact representation for diverse objects in nature and society. At the same time, when people try to describe complex natural phenomena or systems beyond perception by ordinary senses, such as vision or hearing, their descriptions become very vague, imprecise and sometimes misleading. That is why, in such cases, the best description and definition of a natural phenomenon/essence is its mathematical model. For instance, for a long time, people tried to answer the question of what an electron is by saying that it is a very small particle, which has a negative charge and several other properties. However, it was discovered that in many situations electrons behave as waves and not as particles. As a result, informal descriptions of the *electron* have become extremely vague. For instance, it is described as some entity in nature that is very small, sometimes behaves as a particle, while in other cases, it behaves as a wave. Whereas many other properties of electrons have been discovered, such descriptions have become too confusing because people have nothing of this kind in their everyday experience. In addition, these descriptions, which use natural languages and mundane images, are too inexact and as a result, physicists came to the conclusion that the best answer to the above question is the statement that the *electron* is an entity in nature that behaves according to the laws of quantum mechanics.

To answer here the question of what information is, we simply imply that information is a phenomenon described by the theory of information. In the past, this answer was unsatisfactory because there were many theories of information, which gave different representations of this basic phenomenon. Researches have found shortcomings of all such directions and even started to believe that it is impossible to give a comprehensive definition of information, building a unified information theory. They argued this diversity of information uses formed an insurmountable obstacle to creation of a unified comprehensible information theory (*cf.* for example, [1-4]). Moreover, it has been argued, for example, by Gofman [5] and Gilligan [6], that the term *information* has been used in so many different and sometimes incommensurable ways, forms and contexts that it is not even worthwhile to elaborate a single conceptualization achieving general agreement.

However, it has become possible to synthesize all directions and approaches in information studies and to find a solution to the important problem of understanding what information is. This was achieved by utilization of a new definition type in the general theory of information [7]. Namely, to overcome limitations of the conventional approaches and to solve the problem of information definition a parametric definition is used.

The general theory of information has different directions: statistic, semantic, algorithmic, *etc.* Here we continue to develop the categorical direction in the general theory of information. In this direction, information is modeled by categorical and functorial information operators where each portion of information is represented by a categorical or functorial information operator.

Abstract categories emerged in algebra, becoming one of the most popular and efficient tools of contemporary mathematics. Now category theory is also used as a unifying framework for physics (*cf.* for example, [8]), for biology ([9-11]), for computation ([12-19]), for topology, and logic ([20-23]), as well as for the whole mathematics ([24,22]). This provides a base for analyzing physical and information systems and processes by means of categorical structures and methods. Categories work

well as models of information spaces because categorical technology is sufficiently powerful and flexible for information dynamics modeling and exploration.

Information is intrinsically related to transformations [25]. That is why portions of information are modeled by information operators in infological system representation spaces. There are two types of information dynamics depiction in categories: The categorical and functorial representations. The categorical representation of information dynamics, which describes regularities of information processes, preserves internal structures of information spaces associated with infological systems as their state or phase spaces. In it, portions of information are modeled by categorical information operators. The functorial representation of information dynamics preserves external structures of information spaces associated with infological systems as their state or phase spaces. In it, portions of information are modeled by functorial information operators. Here we study categorical representation of information dynamics.

Section 2 provides the reader with knowledge on categories that is necessary for understanding results obtained in this work. Section 3 gives a compressed description of the phenomenological stratum of the general theory of information. In Section 4, we study decompositions of categorical information operators and relations between categorical information operators, such as complementarity of categorical information operators. Section 5 contains open problems in the mathematical theory of information.

## 2. Elements of Category Theory

There are two approaches to the mathematical structure called a *category*. One approach treats categories in the framework of the general set-theoretical mathematics. Another approach establishes categories independently of sets and uses them as a foundation of mathematics different from set theory. It is possible to build the whole mathematics in the framework of categories. For instance, such a basic mathematical concept as a binary relation is frequently studied in categories (*cf.* for example, [26]). Toposes allow one to reconstruct set theory as a subtheory of category theory (*cf.* for example, [24]). According to the first approach, we have the following definition of a category.

**Definition 2.1.** A *category* **C** consists of two collections: Ob **C**, the *objects* of **C**, and Hom **C**, the *morphisms* (also called *arrows*) of **C** that satisfy the following three axioms:

**A1.** For every pair $A$, $B$ of objects, there is a set $\mathrm{Hom}_{\mathbf{C}}(A, B)$, also denoted by $\mathrm{H}_{\mathbf{C}}(A, B)$ or $\mathrm{Mor}_{\mathbf{C}}(A, B)$, called morphisms from $A$ to $B$ in **C.** When $f$ is a morphism from $A$ to $B$, it is denoted by $f\colon A \to B$. The object $A$ is called the domain of $f$ and object $B$ is called the codomain of $f$.

**A2.** For every three objects $A$, $B$ and $C$ from Ob **C**, there is a binary partial operation, which is a partial function from pairs of morphisms that belong to the direct product $\mathrm{Hom}_{\mathbf{C}}(A, B) \times \mathrm{Hom}_{\mathbf{C}}(B, C)$ to morphisms in $\mathrm{Hom}_{\mathbf{C}}(A, C)$ . In other words, when $f\colon A \to B$ and $g\colon B \to C$, there is a morphism $g \circ f\colon A \to C$ called the composition of morphisms $g$ and $f$ in **C**, and also denoted by $gf$. This composition is associative, that is, if $f\colon A \to B$, $g\colon B \to C$ and $h\colon C \to D$, then $h \circ (g \circ f) = (h \circ g) \circ f$.

**A3.** For every object $A$, there is a morphism $1_A$ in $\mathrm{Hom}_{\mathbf{C}}(A, A)$, called the identity on $A$, for which if $f\colon A \to B$, then $1_B \circ f = f$ and $f \circ 1_A = f$.

If $f\colon A \to B$ is a morphism from **C**, then $B = \mathrm{Im}\, f$ and $A = \mathrm{Dom}\, f$.

**Example 2.1.** The category of sets **SET**: objects are arbitrary sets and morphisms are mappings of these sets.

**Example 2.2.** The category of groups **GRP**: objects are arbitrary groups and morphisms are homomorphisms of these groups.

**Example 2.3.** The category of topological spaces **TOP**: objects are arbitrary topological spaces and morphisms are continuous mappings of these topological spaces.

**Remark 2.1.** In the case when a category **C** has a set of objects, it may be treated as a graph where objects are vertices and morphisms are edges.

**Definition 2.2.** A morphism $f$: $A \to B$ from the category **C** is called:

a) A *monomorphism* if for all morphisms $h$ and $k$ in **C** such that the compositions $f \circ h$ and $f \circ k$ exist and are equal, it follows that $h = k$.

b) An *epimorphism* if for all morphisms $h$ and $k$ in **C** such that the compositions $h \circ f$ and $k \circ f$ exist and are equal, it follows that $h = k$.

c) A *bimorphism* if it is both a monomorphism and epimorphism.

d) A *section* if there is a morphism $g$: $B \to A$ from the category **C** such that $g \circ f = 1_A$.

e) A *retraction* if there is a morphism $g$: $B \to A$ from the category **C** such that $f \circ g = 1_B$.

f) An *isomorphism* if it is both a section and retraction.

g) A *constant morphism* if for any morphisms $g$: $C \to A$ and $h$: $C \to A$ from the category **C**, we have $f \circ g = f \circ h$.

h) A *co-constant morphism* if for any morphisms $g$: $B \to D$ and $h$: $B \to D$ from the category **C**, we have $g \circ f = h \circ f$.

i) A *zero morphism* if it is a constant and co-constant morphism.

An *initial object* also called a *coterminal object* of a category **C** is an object $I$ in **C** such that for every object $X$ in **C**, there exists a single morphism $I \to X$. A *terminal object* also called a *final object* of a category **C** is an object $T$ in **C** such that for every object $X$ in **C** there exists a single morphism $X \to T$. Initial objects and terminal objects are dual concepts.

A *zero object* in a category **C** is an object 0 that is both an initial object and a terminal object**.**

Mapping of categories that preserve their structure are called functors. There are functors of two types: covariant functors and contravariant functors.

**Definition 2.3.** A *covariant functor* $F$: **C** $\to$ **K**, also called a *functor*, from a category **C** to a category **K** is a mapping that is stratified into two related mappings $F_{\mathrm{Ob}\mathbf{C}}$ : Ob **C** $\to$ Ob **K** and $F_{\mathrm{Mor}\mathbf{C}}$ : Mor **C** $\to$ Mor **K** , *i.e.*, $F_{\mathrm{Ob}\mathbf{C}}$ associates an object $F(A)$ from the category **K** to each object $A$ from the category **C** and $F_{\mathrm{Mor}\mathbf{C}}$ associates a morphism $F(f)$: $F(A) \to F(B)$ from the category **K** to each morphism $f$: $A \to B$ from the category **C**. In addition, $F$ satisfies the following two conditions:

1. $F(1_A) = 1_{F(A)}$ for every object $A$ from the category **C**;
2. $F(f \circ g) = F(f) \circ F(g)$ for all morphisms $f$ and $g$ from the category **C** when their composition $f \circ g$ exists.

That is, functors preserve identity morphisms and composition of morphisms.

**Definition 2.4.** A *contravariant functor* $F$: **C** $\to$ **K** from a category **C** to a category **K** consists of two mappings $F_{\mathrm{Ob}\mathbf{C}}$ : Ob **C** $\to$ Ob **K** and $F_{\mathrm{Mor}\mathbf{C}}$ : Mor **C** $\to$ Mor **K**, *i.e.*, $F_{\mathrm{Ob}\mathbf{C}}$ associates an object $F(A)$

from the category **K** to each object $A$ from the category **C** and $F_{\mathrm{Mor}\mathbf{C}}$ associates a morphism $F(f)$: $F(A) \to F(B)$ from the category **K** to each morphism $f$: $A \to B$ from the category **C**, that satisfy the following two conditions:

1. $F(1_A) = 1_{F(A)}$ for every object $A$ from the category **C**;
2. $F(f \circ g) = F(g) \circ F(f)$ for all morphisms $f$ and $g$ from the category **C** when their composition $f \circ g$ exists.

It is possible to define a contravariant functor as a covariant functor on the dual category $\mathbf{C}^{\mathrm{op}}$. In what follows, we consider only covariant functors.

**Definition 2.5.** A functor $F$: **C** → **K** is called:

a) *embedding* if $F_{\mathrm{Mor}\mathbf{C}}$ : Mor **C** → Mor **K** is an embedding;

b) *dense*, or *representative*, if for each object $A \in$ Ob **K**, there is some object $B \in$ Ob **C** such that $F(B)$ is isomorphic to $A$;

c) *full* if each Hom-set restriction $F|_{\mathrm{Hom}(A,B)}{}^{\mathrm{Hom}(F(A),F(B))}$ of $F$ is surjective.

d) *faithful* if each Hom-set restriction $F|_{\mathrm{Hom}(A,B)}{}^{\mathrm{Hom}(F(A),F(B))}$ of $F$ is injective.

It is possible to read more about categories, functors and their properties, for example, in [24,27].

## 3. The Phenomenological Stratum of the General Theory of Information

The general theory of information is constructed as an axiomatic theory and has three levels: *Conceptual*, *methodological* (also called *meta-theoretical*) and *theoretical*.

On the *conceptual level*, the essence of information as a dynamic object playing a pivotal role in all walks of reality is explicated. This allows clarifying a quantity of misconceptions, fallacies and illusions.

*Methodological* (*meta-theoretical*) *level* is based on two classes of principles and their relations. The first class contains ontological principles, which bring to light general properties and regularities of information and its functioning. Principles from the second class explain how to measure information and are called axiological principles.

On the *theoretical level*, axioms of theoretical structures and axioms reflecting features of information are introduced and utilized for building models of information and related phenomena, e.g., information flow or information processing. These models are employed in studies of information and various related systems and phenomena, e.g., information flow in society or information processing systems, such as computers and networks.

To clarify the concept of information, we consider here the basic ontological principles. The first of them separates local and global approaches to information definition, *i.e.*, in what context information is defined.

**Ontological Principle O1** (**the *Locality Principle***). It is necessary to separate information in general from information (or a portion of information) for a system $R$.

In other words, empirically, it is possible to speak only about information (or a portion of information) for a system. In the mathematical model studied in this paper, portions of information are formalized as information operators in infological system representation spaces.

**Definition 3.1.** The system ***R*** with respect to which some information is considered is called the *receiver*, *receptor* or *recipient* of this information.

Such a receiver/recipient can be a person, community, class of students, audience in a theater, animal, bird, fish, computer, network, database and so on.

The Locality Principle explicates an important property of information, but does not answer the question "What is information?" The essence of information is described by the second ontological principle, which has several forms.

**Ontological Principle O2 (the *General Transformation Principle*).** In a broad sense, *information* for a system *R* is a capacity to cause changes in the system *R*.

Thus, we may understand information in a broad sense as a capacity (ability or potency) of things, both material and abstract, to change other things. Information exists in the form of *portions of information*. Informally, a portion of information is such information that can be separated from other information. Information is, as a rule, about something. What information is about is called the *object* of this information.

The Ontological Principle O2 has several consequences. First, it demonstrates that information is closely connected to transformation. Namely, it means that information and transformation are functionally similar because they both point to changes in a system. At the same time, they are different because information is potency for (or in some sense, cause of) change, while transformation is the change itself, or in other words, transformation is an operation, while information is what induces this operation.

Second, the Ontological Principle O2 explains *why* information influences society and individuals all the time, as well as why this influence grows with the development of society. Namely, reception of information by individuals and social groups induces transformation. In this sense, information is similar to energy. Moreover, according to the Ontological Principle O2, energy is a kind of information in a broad sense. This well correlates with the von Weizsäcker's idea (*cf.* [3]) that *energy might in the end turn out to be information*.

Third, the Ontological Principle O2 makes it possible to separate different kinds of information. For instance, people, as well as any computer, have many kinds of memory. It is even supposed that each part of the brain has several types of memory agencies that work in somewhat different ways to suit particular purposes [28]. Thus, it is possible to consider each of these memory agencies as a separate system and to study differences between information that changes each type of memory. This might help to understand the interplay between stability and flexibility of mind, in general, and memory, in particular.

In essence, we can see that all kinds and types of information are encompassed by the Ontological Principle O2. In the most concise form, it is demonstrated in [7].

However, the common usage of the word information does not imply such wide generalizations as the Ontological Principle O2 implies. Thus, we need a more restricted theoretical meaning because an adequate theory, whether of information or of anything else, must be in significant accord with our common ways of thinking and talking about what the theory is about, else there is the danger that theory is not about what it purports to be about.

Information in a proper sense is defined based on the concept of structural infological systems. In essence, any subsystem of a system may be considered as its infological system. However, information in a proper sense acts on structural infological systems. An infological system is *structural* if all its elements are structures. For example, systems of knowledge, which are paradigmatic infological systems, are structural because their elements are structures. Other examples of structural infological systems are: systems of beliefs, systems of values, systems of goals, and systems of ideas.

To achieve precision in the information definition, we do two conceptual steps. At first, we make the concept of information relative to the chosen infological system IF($R$) of the system $R$ and then we select a specific class of infological systems to specify information in the strict sense. That is why it is impossible and, as well as, counterproductive to give an exact and thus, too rigid and restricted definition of an infological system. Indeed, information is a very rich and widespread phenomenon to be reflected by a restricted rigid definition (*cf.* for example, [2,7,29,30]).

Infological system plays the role of a free parameter in the general theory of information, providing for representation of different kinds and types of information in this theory. That is why the concept of *infological system*, in general, should not be limited by boundaries of exact definitions. A free parameter must really be free. Identifying an infological system IF($R$) of a system $R$, we can define information relative to this system. This definition is expressed in the following principle.

**Ontological Principle O2g** (**the *Relativized Transformation Principle***). *Information* for a system *R relative to the infological system* IF($R$) is a capacity to cause changes in the system IF($R$).

As a model example of an infological system IF($R$) of an intelligent system $R$, we take the system of knowledge of $R$. In cybernetics, it is called the *thesaurus* Th($R$) of the system $R$. Another example of an infological system is the memory of a computer. Such a memory is a place in which data and programs are stored and is a complex system of diverse components and processes.

Elements from IF($R$) are called *infological elements*.

There is no exact definition of infological elements although there are various entities that are naturally considered as infological elements as they allow one to build theories of information that inherit conventional meanings of the word *information*. For instance, knowledge, data, images, ideas, algorithms, procedures, scenarios, schemas, values, goals, ideals, fantasies, abstractions, beliefs, and similar objects are standard examples of infological elements.

When we take a physical system $D$ as the infological system and allow only for physical changes, information with respect to $D$ coincides with energy.

Taking a mental system $B$ as the infological system and considering only mental changes, we see that information with respect to $B$ coincides with mental energy.

These ideas are crystallized in the following principle.

**Ontological Principle O2a** (**the *Special Transformation Principle***). *Information in the strict sense* or *proper information* or, simply, *information* for a system $R$, is a capacity to change structural infological elements from an infological system IF($R$) of the system $R$.

To better understand how infological systems can help explicating the concept of information in the strict (conventional) sense, we consider cognitive infological systems.

An infological system IF($R$) of the system $R$ is called *cognitive* if IF($R$) contains (stores) elements or constituents of cognition, such as knowledge, data, ideas, fantasies, abstractions, beliefs, *etc.* A cognitive infological system of a system $R$ is denoted by CIF($R$) and is related to cognitive information.

In this case, it looks like it is possible to give an exact definition of a cognitive infological system. However, now cognitive sciences do not know all structural elements involved in cognition. A straightforward definition specifies cognition as an activity (process) that gives knowledge. At the same time, we know that knowledge, as a rule, comes through data and with data. So, data are also involved in cognition and thus, have to be included in cognitive infological systems. Besides, cognitive processes utilize such structures as ideas, algorithms, procedures, scenarios, images, beliefs, values, measures, problems, tasks, *etc.* Thus, to comprehensively represent cognitive information, it is imperative to include all such objects in cognitive infological systems.

For those who prefer to have an exact definition contrary to a broader perspective, it is possible to define a cognitive infological system as the system of knowledge. This approach was used by [31,32].

Cognitive infological systems are standard examples of infological systems, while their elements, such as knowledge, data, images, ideas, fantasies, abstractions, and beliefs, are standard examples of infological elements. Cognitive infological systems are very important, especially, for intelligent systems as the majority of researchers believe that information is intrinsically connected to knowledge.

**Ontological Principle O2c** (**the *Cognitive Transformation Principle***). *Cognitive information* for a system $R$, is a capacity to cause changes in the cognitive infological system CIF($R$) of the system $R$.

As the cognitive infological system contains knowledge of the system it belongs to, cognitive information is the source of knowledge changes.

It is useful to understand that in the definition of cognitive information, as well as of other types of information in the strict sense, it is assumed that an infological system IF($R$) of the system $R$ is a part (subsystem) of the system $R$. However, people have always tried to extend their cognitive tools using different things from their environment. In ancient times, people made marks on stones and sticks. Then they used paper. Now they use computers and computer networks.

There are two ways to take this peculiarity into consideration. In one approach, it is suggested to consider *extended infological systems* that do not completely belong to the primary system $R$ that receives information. For instance, taking an individual $A$, it is possible to include in the extended cognitive infological system IF($A$) of $A$ not only the mind of $A$ but also memory of the computer that $A$ uses, books that $A$ reads and cognitive objects used by $A$.

Another approach extends the primary system $R$ as a cognitive object, including all objects used for cognitive purposes. In this case, when we regard an individual $A$ as a cognitive system $R$, we have to include (in $R$) all cognitive tools used by $A$. The second approach does not demand to consider extended infological systems. In this case, all infological systems of $R$ are parts (subsystems) of the primary system $R$.

As a result, we come to the situation where the concept of information is considered on three basic levels of generality:

1. *Information in a broad sense* is considered when there are no restrictions on the infological system (*cf.* Ontological Principle O2).

2. *Information in the strict sense* is considered when the infological system consists of structural elements (*cf.* Ontological Principle O2a).

3. *Cognitive information* is considered when the infological system consists of cognitive structures, such as knowledge, beliefs, ideas, images, *etc.* (*cf.* Ontological Principle O2c).

As a result, we come to three levels of information understanding:

1. *Information in a broad sense* for a system *R* is a capability (potential) to change (transform) this system in any way.

2. *Information in the strict sense* for a system *R* is a capability (potential) to change (transform) structural components of this system, e.g., cognitive information changes knowledge of the system, affective information changes the state of the system, while effective information changes system orientation [7].

3. *Cognitive information* for a system *R* is a capability (potential) to change (transform) the cognitive subsystem of this system.

Let us explicate other properties of information, taking into consideration a portion *I* of information for a system *R*.

**Ontological Principle O3** (**the *Embodiment Principle***). *For any portion of information I, there is always a carrier C of this portion of information for a system R.*

People get information from books, magazines, TV and radio sets, computers, and from other people. To store information people use their brains, paper, tapes, and computer disks. All these entities are carriers of information.

For adherents of the materialistic approach, the Ontological Principle O3 must be changed to its stronger version.

**Ontological Principle OM3** (**the *Material Embodiment Principle***). *For any portion of information I, there is some substance C that contains I.*

The substance (material object) ***C*** that is a carrier of the portion of information *I* is called the *physical*, or *material*, *carrier* of *I*.

**Ontological Principle O4** (**the *Representability Principle***). *For any portion of information I, there is always a representation C of this portion of information for a system R.*

As any information representation is, in some sense, its carrier, the Ontological Principle O4 implies the Ontological Principle O3.

The first four ontological principles ((O1)-(O4) or (O1)-(OM4)) imply that information connects the carrier *C* with the system *R* and thus, information *I* is a component of the following fundamental triad [7],

$$(C, I, R)$$

People empirically observed that for information to become available, the carrier must interact with a receptor that was capable of detecting information the carrier contained. This empirical fact is represented by the following principle.

**Ontological Principle O5** (**the *Interaction Principle***). *A transaction/transition/transmission of information goes on only in some interaction of C with R.*

However, being necessary, interaction is not sufficient for information transmission. Thus, we need one more principle.

**Ontological Principle O6** (**the *Actuality Principle***). *A system R accepts a portion of information I only if the transaction/transition/transmission causes corresponding transformations.*

For instance, if after reading this paper, your knowledge remains the same, you do not accept cognitive information from this text. In a general case, when the recipient's knowledge structure was not changed, there is no cognitive information reception.

**Ontological Principle O7** (**the *Multiplicity Principle***). *One and the same carrier C can contain different portions of information for one and the same system R.*

Thus, all ontological principles form three groups:

- Ontological Principles O1 and O2 reflect intrinsic substantial properties of information.
- Ontological Principles O3, O4 and O7 reflect representation properties of information.
- Ontological Principles O5 and O6 reflect dynamic properties of information.

## 4. Categorical Representation of Information Dynamics

To build a categorical representation of information dynamics, *i.e.*, of regularities of information processes, we start with constructing a categorical representation of the information receptor/receiver $R$ because according to the Principles O1 and O2, information is determined by its action on the information receptor/receiver. When we are interested in information in the strict sense, we consider only an infological system IF($R$) of $R$.

Let us take a category **C**. Objects from the category **C** can represent different infological systems IF, the same infological system IF in different states or different infological systems IF, the same infological system IF in different states. In all cases, the category **C** is called a *categorical information space*.

**Definition 4.1.** a) A *categorical representation* of the infological system IF($R$) (or $R$) assigns objects from **C** to the states of IF($R$) (correspondingly, $R$) and morphisms from **C** to transformations of IF($R$) (or $R$).

b) A categorical representation of the infological system IF($R$) (or $R$) is called *complete* if all states of IF($R$) (correspondingly, $R$) are represented and each object in Ob **C** is a representation of some state of IF($R$) (or $R$).

c) The category **C** is called a *categorical information state space* for the infological system IF($R$) (for the system $R$).

In this case, $\mathrm{Mor}_{\mathrm{IF}(R)}\ \mathbf{C}$ ($\mathrm{Mor}_R\ \mathbf{C}$) denotes the set of all morphisms from **C** representing transformations of IF($R$) (correspondingly, $R$) and if $A, B \in \mathrm{Ob}\ \mathbf{C}$, then $\mathrm{Hom}_{\mathrm{IF}(R)}(A, B) = \mathrm{Mor}_{\mathrm{IF}(R)}\ \mathbf{C} \cap \mathrm{Hom}_{\mathbf{C}}(A, B)$ ($\mathrm{Hom}_R\ (A, B) = \mathrm{Mor}_R\ \mathbf{C} \cap \mathrm{Hom}_{\mathbf{C}}(A, B)$).

In what follows, we consider only complete representations.

**Example 4.1.** Objects of the category **C** are linear spaces, while morphisms are linear mappings (linear operators) of these spaces.

**Example 4.2.** Objects of the category **C** are sets, e.g., sets of knowledge items, such as considered in [31], or of propositions, such as in [33,34], while morphisms are mappings (transformations) of these sets.

**Example 4.3.** Words, or more generally, texts are information carriers, as well as information representations. Thus, it is natural to take words (texts) as objects of the category **C**. Then morphisms of this category are computations that transform one word (system of words) into another one. These computations may be restricted to computations of some class of abstract automata, such as finite automata, Turing machines, inductive Turing machines or neural networks (*cf.* e.g., [35]. It is possible to consider information automata in the sense of Cooper [36] as devices that perform computations.

**Example 4.4.** Thesaurus is a natural infological system [7,32]. An efficient representation of a thesaurus is a set of words, or more generally, of texts in some languages, which are information representations, as well as information carriers when these texts are in a material form, e.g., strings of symbols on paper or states of a computer memory [32]. Thus, it is natural to take sets of words (or texts) as objects of the category **C**. Then morphisms of this category are multiple computations ([37]) performed by systems/automata from some class, e.g., Turing machines, inductive Turing machines or finite automata.

**Example 4.5.** Classifications, which are also infological systems used in a variety of areas, and their infomorphisms in the sense of [38] form a category in which classifications are objects and infomorphism are morphisms.

Informally, a portion of information $I$ is a potency to cause changes in (infological) systems, *i.e.,* to change the state of this system. Assuming that all systems involved in such changes are represented in a category **C**, we see that a change in a system may by represented by a morphism from this category. This gives us a transformation of the system that receives information. As a result, the portion of information $I$ is represented by a categorical information operator Op($I$). When it does not cause misunderstanding, it is possible to denote the information portion $I$ and the information operator Op($I$) by the same letter $I$.

**Definition 4.2.** a) A *pure categorical representation* of a collection **IF** = { IF($R_i$); $i \in I$ } of infological systems assigns objects from **C** to the infological systems IF($R_i$) and morphisms from **C** to transformations of one infological system into another.

b) A pure categorical representation of the collection **IF** is called *complete* if all systems from **IF** are represented and each object in Ob **C** is a representation of some system from **IF**.

c) The category **C** is called a *pure categorical information system space* for the collection **IF**.

In this case, $\mathrm{Mor}_{\mathbf{IF}}\,\mathbf{C}$ denotes the set of all morphisms from **C** represent transformations of systems from **IF** and if $A, B \in \mathrm{Ob}\,\mathbf{C}$, then $\mathrm{Hom}_{\mathbf{IF}}(A, B) = \mathrm{Mor}_{\mathbf{IF}}\,\mathbf{C} \cap \mathrm{Hom}_{\mathbf{C}}(A, B)$.

In what follows we consider only complete representations.

**Example 4.6.** Information operators can represent information that changes DNA molecules to RNA molecules and RNA molecules to proteins.

**Definition 4.3.** a) An *enriched categorical representation* of the collection **IF** = { IF($R_i$); $i \in I$) of marked infological systems assigns objects from **C** to the states of infological systems IF($R_i$) and

morphisms from **C** to transformations of one infological system in some state into another infological system in some state.

b) An enriched categorical representation of the collection **IF** is called *complete* if all systems from **IF** and all their states are represented and each object in Ob **C** is a representation of some system from **IF** and its state.

c) The category **C** is called an *enriched categorical information extended system space* for the collection **IF**.

Definitions 4.1 and 4.2 describe particular cases of categorical representations from the Definition 4.3 because a categorical representation of IF($R$) is an enriched categorical representation of one infological system, while a pure categorical representation of a collection **IF** is enriched categorical representation of **IF** assuming that each infological system from **IF** has only one state.

In this case, objects from **C** represent pairs (IF($R$), $r$) where $r$ is a state of the infological system IF($R$). $\mathrm{Mor}_{\mathrm{IF}(R)}$ **C** ($\mathrm{Mor}_R$ **C**) denotes the set of all morphisms from **C** represent transformations of IF($R$) (correspondingly, $R$) and if $A$, $B \in$ Ob **C**, then $\mathrm{Hom}_{\mathrm{IF}(R)}(A, B) = \mathrm{Mor}_{\mathrm{IF}(R)}\ \mathbf{C} \cap \mathrm{Hom}_{\mathbf{C}}(A, B)$ ($\mathrm{Hom}_R\,(A, B) = \mathrm{Mor}_R\ \mathbf{C} \cap \mathrm{Hom}_{\mathbf{C}}(A, B)$).

Note that two different states of a system may be considered as different systems. This allows us to reduce categorical representations from Definitions 4.1 and 4.3 to categorical representations from Definition 4.2. At the same time, it is also possible (at least formally) to consider different systems as the states of one universal system. This allows us to reduce categorical representations from Definitions 4.2 and 4.3 to categorical representations from Definition 4.1. It means that although Definition 4.3 looks as the most general, on the abstract level, it is possible to reduce it to any of the other two definitions.

In what follows we consider only complete representations.

Let us consider a categorical information space **C**.

**Definition 4.4.** A *categorical information operator* Op($I$) over the categorical information space **C** is a mapping Op($I$): Ob **C** $\rightarrow$ Mor **C** such that for any $A \in$ Ob **C**, its image $\mathrm{Op}(I)(A) \in \mathrm{Hom}_{\mathrm{IF}(R)}\,(A, X)$ for some $X \in$ Ob **C**. The morphism Op($I$)($A$) is called the *component* of the categorical information operator Op($I$) at $A$.

Informally, a categorical information operator shows how each object (infological system) changes when a portion of information is received.

Categorical operators can be total or partial.

Note that it is possible that $X$ coincides with the same object $A$. When $\mathrm{Op}(I)(A) = 1_A$, it means that the information portion $I$ does not change $A$ or the state of $A$.

**Example 4.7.** Taking the category **C** from the Example 4.1, in which objects of the category **C** are linear spaces and morphisms are linear operators in these spaces, we can build categorical information operators. These operators assign a linear operator to each linear space from **C**. Such operators represent, for example, information extraction by measurement in physics (*cf.* for example, [39]).

**Example 4.8.** Taking the category **C** from the Example 4.3, in which words (or texts) are objects and morphisms are computations performed by automata from some class, e.g., Turing machines,

inductive Turing machines or finite automata, we obtain computational information operators. These operators assign a computation to each word (text) where this word (text) is the input.

**Example 4.9.** Taking the category **C** from the Example 4.4, in which an object is a thesaurus, e.g., represented by systems of words (or texts), and morphisms are transformations of these thesauruses, we obtain a cognitive information operator, which assigns to each thesaurus its transformation. There are different types of cognitive information operators:

1. computational cognitive information operators;
2. analytical cognitive information operators;
3. matrix cognitive information operators;
4. set-theoretical cognitive information operators;
5. named-set-theoretical cognitive information operators.

For instance, transformations of knowledge states studied by Mizzaro [31] form set-theoretical cognitive information operators, while infomorphisms studied by Barwise and Seligman [38] shape named-set-theoretical cognitive information operators and transformations studied by Shreider [32] bring into being analytical cognitive information operators.

Information is often connected to meaning. Some researchers even cannot imagine information without meaning, understanding meaning in the conventional sense (*cf.* for example, [40]). However, it was demonstrated that meaning is not a necessary attribute of information (*cf.* for example, [41]). For instance, Shannon's information theory is very useful although it ignores meaning

However, as the general theory of information (GTI) encompasses all other approaches in information theory, it also includes semantic theories of information (*cf.* [7]). As a result, the GTI allows us to reflect meaning of information by the following model. In a formal context, meaning is represented by formal structures, e.g., it is possible to represent meaning by propositions, by semantic networks or by frames. There are also other formal representations of meaning, such as the Universal grammar of Montague, Discourse Representation Semantics, and File Change Semantics. Formal structures that represent meaning for a given system $R$ constitute a semantic system $M$. It is possible to take this semantic system $M$ as the infological system of the system $R$. This incorporates semantic approach to information into the GTI framework.

In this formal context, acceptance of meaningful information $I$ changes the semantic system $M$ by integrating the meaning of $I$ into $M$. When we employ a categorical model, representing states of the semantic system $M$ by objects of a category $\boldsymbol{C}$, information is naturally represented by a categorical operator, which changes the states of $M$, reflecting adaptation of the meaning of $I$.

There are different operations with categorical information operators. One of the most important is sequential composition.

**Definition 4.5.** The *sequential composition* (often called simply *composition*) of categorical information operators $O_1$ and $O_2$ is a mapping $O_2 \circ O_1$: Ob **C** $\rightarrow$ Mor **C** such that for any $A \in$ Ob **C**, if $O_1(A) \in \mathrm{Hom}_{\mathbf{C}}(A, B)$, then $[O_2 \circ O_1](A) = O_2(B) \circ O_1(A)$.

By definition, the (sequential) composition of categorical information operators also is a categorical information operator. It models sequential reception of information by systems.

We do not use the shorter name *composition* instead of the name *sequential composition* because there other types of compositions of categorical information operators, for example, parallel composition of categorical information operators or concurrent composition of categorical information operators.

Let us consider some relations between categorical information operators.

**Definition 4.6.** a) If $O_1$ and $O_2$ are categorical information operators, then $O_1 \preccurlyeq O_2$ (we say, $O_1$ is an *information suboperator* of $O_2$) if for any object $A$ from **C**, we have

$$O_2(A) = f \circ O_1(A) \text{ for some morphism } f$$

This condition is equivalent to the following commutative diagram:

$$\begin{tikzcd} A \arrow[d, "O_1(A)"'] \arrow[dr, "O_2(A)"] & \\ B \arrow[r, "f"'] & C \end{tikzcd} \tag{4.1}$$

b) If $O_1$ and $O_2$ are categorical information operators and K is a class (type) of morphisms, e.g., K consists of all monomorphisms, then $O_1$ is a K-*uniform information suboperator* of $O_2$ ($O_1 \preccurlyeq_K O_2$) if for any object $A$ from **C**, we have

$$O_2(A) = f \circ O_1(A) \text{ for some morphism } f \text{ from K}$$

c) If $O_1$ and $O_2$ are categorical information operators, then $O_1$ is an *information net suboperator* of $O_2$ ($O_1 \subseteq O_2$) if for any object $A$ from **C**, we have

$$O_2(A) = O_3(\mathrm{Im}\, A) \circ O_1(A) \text{ for some categorical information operator } O_3$$

d) If $O_1$ and $O_2$ are categorical information operators, then $O_1 \asymp O_2$ (we say, $O_1$ is a *equivalent* to $O_2$) if $O_1 \preccurlyeq_K O_2$ and $O_2 \preccurlyeq_K O_1$.

e) If $O_1$ and $O_2$ are categorical information operators and K is a class (type) of morphisms, then $O_1 \asymp_K O_2$ (we say, $O_1$ is a K-*uniformly equivalent* to $O_2$) if $O_1 \preccurlyeq_K O_2$ and $O_2 \preccurlyeq_K O_1$ .

f) If $O_1$ and $O_2$ are categorical information operators, then $O_1 \equiv O_2$ (we say, $O_1$ is a *net equivalent* to $O_2$) if $O_1 \subseteq O_2$ and $O_2 \subseteq O_1$.

The concept of a information suboperator models the situation when one portion of information can be converted to another portion of information by transformations in the information space, *i.e.*, $O_1 \preccurlyeq O_2$ means that information represented by the information operator $O_1$ can be converted to information represented by the information operator $O_2$.

The concept of a uniform information suboperator models the situation when one portion of information can be converted to another portion of information by transformations in the information space that belong to a given class of transformations, *i.e.*, $O_1 \preccurlyeq_K O_2$ means that information represented by the information operator $O_1$ can be converted to information represented by the information operator $O_2$ by transformations that belong to the class K.

The concept of a net information suboperator models the situation when one portion of information can be converted to (complemented by) another portion of information by some information operator.

As it is usual in mathematics, all three relations generate corresponding equivalence relations: equivalence of information operators $\asymp$, uniform equivalence of information operators $\asymp_K$ and net equivalence of information operators $\equiv$.

**Example 4.10.** Objects of the category **C** are sets of well formed formulas of a logical calculus, e.g., sets of of propositions, such as considered in [33,34], while morphisms are finite deductions, which add deduced formulas to the initial set. It means that reception of information initiates some deductive process.

Let us assume that $O_1$ and $O_2$ are categorical information operators in **C**, while $O_1 \preccurlyeq O_2$. In this interpretation, the set $B = \mathrm{Im}\ O_1(A)$ of well formed formulas is less than or equal to the set $C = \mathrm{Im}\ O_2(A)$ of well formed formulas. It means that if $O_1 \preccurlyeq O_2$, then the categorical information operator $O_1(A)$ adds less formulas than the categorical information operator $O_1(A)$ does. This is true for all objects $A$ from **C**, giving the meaning of the relation $\preccurlyeq$.

**Example 4.11.** Objects of the category **C** are sets of knowledge items, such as considered in [31] or in [7], while morphisms are transformations of these sets that only add new knowledge items but never delete them.

Let us assume that $O_1$ and $O_2$ are categorical information operators in **C**, while $O_1 \preccurlyeq O_2$. In this interpretation, the set $B = \mathrm{Im}\ O_1(A)$ of knowledge items is less than or equal to the set $C = \mathrm{Im}\ O_2(A)$ of knowledge items. It means that if $O_1 \preccurlyeq O_2$, then the categorical information operator $O_1(A)$ adds less knowledge items than the categorical information operator $O_1(A)$ does. This is true for all objects $A$ from **C**, giving the meaning of the relation $\preccurlyeq$.

**Lemma 4.1.** The relation $\subseteq$ is stronger than the relation $\preccurlyeq$, *i.e.*, $O_1 \subseteq O_2$ implies $O_1 \preccurlyeq O_2$ for any categorical information operators $O_1$ and $O_2$.

**Corllary 4.1.** The relation $\equiv$ is stronger than the relation $\asymp$, *i.e.*, $O_1 \equiv O_2$ implies $O_1 \asymp O_2$ for any categorical information operators $O_1$ and $O_2$.

**Lemma 4.2.** If the class K is closed with respect to all identity morphisms$1_A$ form Mor **C**, then the relation $\preccurlyeq_K$ is reflexive.

Indeed, for any categorical information operator O, we have $O \preccurlyeq_K O$ because for any object $A$ from **C**, $O(A) = 1_B \circ O(A)$ where $B = \mathrm{Im}\ A$ (*cf.* Diagram (4.1)).

**Lemma 4.3.** If the class K is closed with respect to composition of morphisms, then the relation $\preccurlyeq_K$ is transitive.

<u>Proof</u>. Let us consider categorical information operators $O_1$, $O_2$ and $O_3$. If $O_1 \preccurlyeq_K O_2$ and $O_2 \preccurlyeq_K O_3$, then $O_2(A) = f \circ O_1(A)$ for some morphism $f$ from K and $O_3(A) = g \circ O_2(A)$ for some morphism $g$ form K.

These conditions are equivalent to the following commutative diagram:

$$\begin{tikzcd} A \arrow[d, "O_1(A)"'] \arrow[dr, "O_2(A)"] \arrow[drr, "O_3(A)"] & & \\ B \arrow[r, "f"'] & C \arrow[r, "g"'] & D \end{tikzcd} \qquad (4.2)$$

Consequently, we have

$$O_3(A) = g \circ O_2(A) = g \circ (f \circ O_1(A)) = (g \circ f) \circ O_1(A) = h \circ O_1(A)$$

where $h = g \circ f$ belongs to K. Consequently, $O_1 \preccurlyeq_K O_3$.

Lemma is proved.

Lemmas 4.2 and 4.3 imply the following result.

**Proposition 4.1.** If the class K is closed with respect to composition of morphisms and all identity morphisms$1_A$ form Mor **C**, then the relation $\preccurlyeq_K$ is a preorder on the class of all information operators, *i.e.*, the relation $\preccurlyeq_K$ is reflexive and transitive.

As the relation $\preccurlyeq$ is a particular case of the relation $\preccurlyeq_K$ when K = Mor **C** and the class Mor **C** is closed with respect to composition of morphisms, Proposition 4.1 implies the following result.

**Corollary 4.2.** The relation $\preccurlyeq$ is a preorder on the class of all information operators, *i.e.*, the relation $\preccurlyeq$ is reflexive and transitive.

**Proposition 4.2.** If the class K is closed with respect to composition of morphisms and all identity morphisms$1_A$ form Mor **C**, then the relation $\asymp_K$ is an equivalence on the class of all information operators, *i.e.*, the relation $\asymp_K$ is reflexive, symmetric and transitive.

Indeed, by definition, the relation $\asymp_K$ is symmetric and by Proposition 4.1, it is reflexive and transitive.

As the relation $\asymp$ is a particular case of the relation $\asymp_K$ when K = Mor **C** and the class Mor **C** is closed with respect to composition of morphisms, Proposition 4.1 implies the following result.

**Corollary 4.3.** The relation $\asymp$ is an equivalence relation on the class of all information operators, *i.e.*, the relation $\asymp$ is reflexive, symmetric and transitive.

Let us consider the class ISO of all isomorphisms of the category **C**.

**Proposition 4.3.** The relation $\preccurlyeq_{ISO}$ is an equivalence relation on the class of all information operators, *i.e.*, the relation $\preccurlyeq_{ISO}$ is reflexive, symmetric and transitive.

Proof. By Proposition 4.1, the relation $\preccurlyeq_{ISO}$ is a preorder on the class of all information operators because any identity morphism $1_X$ is an isomorphism and composition of two isomorphisms is also an isomorphism [27]. So, it is necessary to show that the relation $\preccurlyeq_{ISO}$ is symmetric.

Let us consider two categorical information operators $O_1$ and $O_2$ such that $O_1 \preccurlyeq_{ISO} O_2$. By Definition 4.6, it means that there is a commutative Diagram (4.1) in which $f$ is an isomorphism. As $f$ is an isomorphism, there is a morphism $g$: $C \to B$, such that $g \circ f = 1_B$.

Thus, for $A$ is an arbitrary object from the category **C**, the relation $O_1 \preccurlyeq_{ISO} O_2$ implies $O_2(A) = f \circ O_1(A)$, which in turn, implies the following sequence of equalities:

$$g \circ O_2(A) = g \circ (f \circ O_1(A)) = (g \circ f) \circ O_1(A) = 1_B \circ O_1(A) = O_1(A)$$

As $A$ is an arbitrary object from the category **C**, it means that $O_2 \preccurlyeq_{ISO} O_1$, *i.e.*, the relation $\preccurlyeq_{ISO}$ is symmetric.

Proposition is proved.

**Proposition 4.4.** The relation $\subseteq$ is a preorder on the class of all information operators, *i.e.*, the relation is reflexive and transitive.

Proof. (a) For any categorical information operator O, we have $O \subseteq O$ because there is the identity categorical information operator Id such that $\mathrm{Id}(A) = 1_A$ for any object $A$ from **C** and $O_1(A) = 1_B \circ O_1(A) = \mathrm{Id}(B) \circ O(A)$ where $B = \mathrm{Im}\, O(A)$ (*cf.* Diagram (4.1)).

(b) Let us consider categorical information operators $O_1$, $O_2$ and $O_3$. If $O_1 \subseteq O_2$ and $O_2 \subseteq O_3$, then $O_2(A) = O_4(\mathrm{Im}\, O_1(A)) \circ O_1(A)$ for some categorical information operators $O_4$ and $O_3(A) = O_5(\mathrm{Im}\, O_2(A)) \circ O_2(A)$ for some categorical information operators $O_5$.

These conditions are equivalent to the following commutative diagram:

$$\begin{tikzcd} A \arrow[d, "O_1(A)"'] \arrow[dr, "O_2(A)"] \arrow[drr, "O_3(A)"] & & \\ B \arrow[r, "O_4(\mathrm{Im}\, O_1(A))"'] & C \arrow[r, "O_5(\mathrm{Im}\, O_2(A))"'] & D \end{tikzcd} \tag{4.3}$$

As $\mathrm{Im}\, O_4(\mathrm{Im}\, O_1(A)) = \mathrm{Im}\, O_2(A)$, we have

$$O_3(A) = O_5(\mathrm{Im}\, O_2(A)) \circ O_2(A) = O_5(\mathrm{Im}\, O_2(A)) \circ (O_4(\mathrm{Im}\, O_1(A)) \circ O_1(A)) =$$
$$(O_5(\mathrm{Im}\, O_2(A)) \circ O_4(\mathrm{Im}\, O_1(A))) \circ O_1(A) =$$
$$(O_5(\mathrm{Im}\, O_4(\mathrm{Im}\, O_1(A)))) \circ O_4(\mathrm{Im}\, O_1(A))) \circ O_1(A) = h \circ O_1(A)$$

where $h = (O_5(\mathrm{Im}\, O_4(\mathrm{Im}\, O_1(A)))) \circ O_4(\mathrm{Im}\, O_1(A)))$. Consequently, $O_1 \subseteq O_3$.

Proposition is proved.

**Proposition 4.5.** The relation $\equiv$ is an equivalence relation on the class of all information operators, *i.e.*, the relation $\equiv$ is reflexive, symmetric and transitive.

Indeed, by definition, the relation $\equiv$ is symmetric and by Proposition 4.4, it is reflexive and transitive.

**Proposition 4.6.** The following conditions are equivalent:

(a) $O_1 \preccurlyeq_{ISO} O_2$;
(b) $O_2 \preccurlyeq_{ISO} O_1$;
(c) $O_1 \asymp_{ISO} O_2$;

Proof. Let us consider two categorical information operators $O_1$ and $O_2$ such that $O_1 \preccurlyeq_{ISO} O_2$. By Definition 4.6, it means that there is a commutative Diagram (4.1) in which $f$ is an isomorphism. As $f$ is an isomorphism, there is a morphism $g$: $C \to B$, such that $g \circ f = 1_B$.

Thus, for $A$ is an arbitrary object from the category **C**, the relation $O_1 \preccurlyeq_{ISO} O_2$ implies $O_2(A) = f \circ O_1(A)$, which in turn, implies the following sequence of equalities:

$$g \circ O_2(A) = g \circ (f \circ O_1(A)) = (g \circ f) \circ O_1(A) = 1_B \circ O_1(A) = O_1(A)$$

As $A$ is an arbitrary object from the category **C**, it means that $O_2 \preccurlyeq_{ISO} O_1$.

In a similar way, $O_2 \preccurlyeq_{ISO} O_1$ implies $O_1 \preccurlyeq_{ISO} O_2$.

Consequently, each of the relations $O_1 \preccurlyeq_{ISO} O_2$ or $O_2 \preccurlyeq_{ISO} O_1$ implies $O_1 \asymp_{ISO} O_2$.

Proposition is proved.

Let us consider the class SEC of all sections and the class RET of all retractions of the category **C**.

**Proposition 4.7.** $O_1 \preccurlyeq_{SEC} O_2$ if and only if $O_2 \preccurlyeq_{RET} O_1$.

Proof. Necessity. Let us consider two categorical information operators $O_1$ and $O_2$ such that $O_1 \preccurlyeq_{SEC} O_2$. By Definition 4.6, it means that there is a commutative Diagram (4.1) in which $f$: $B \to C$ is a section. As $f$ is a section (*cf.* Section 2), there is a morphism $g$: $C \to B$, such that $g \circ f = 1_B$. By Definition 2.2, $g$ is a retraction.

Thus, since an arbitrary object $A$ from the category **C**, the relation $O_1 \preccurlyeq_{SEC} O_2$ implies $O_2(A) = f \circ O_1(A)$, which in turn, implies the following sequence of equalities:

$$g \circ O_2(A) = g \circ (f \circ O_1(A)) = (g \circ f) \circ O_1(A) = 1_B \circ O_1(A) = O_1(A)$$

As $A$ is an arbitrary object from the category **C** and $g$ is a retraction, it means that $O_2 \preccurlyeq_{RET} O_1$.

Sufficiency. In a similar way, $O_2 \preccurlyeq_{RET} O_1$ implies $O_1 \preccurlyeq_{SEC} O_2$.

Proposition is proved.

**Definition 4.7.** a) If $A$ and $B$ are objects from the category **C** and O is a categorical information operator, then $A \leq_O B$ (we say, an object $A$ is *less than* or *equal to* an object $B$ *with respect to* a categorical information operator O) if for any object $C$ from **C**, there is a morphism $f$ such that the following diagram is commutative:

$$\begin{tikzcd} A \arrow[d, "f"'] \arrow[dr, "O(A)"] & \\ B \arrow[r, "O(B)"'] & C \end{tikzcd} \tag{4.4}$$

*i.e.*, $O(A) = O(B) \circ f$.

b) an object $A$ is O-*equivalent* to an object $B$ ( $A \approx_O B$ ) if for any object $C$ from **C**, there is an isomorphism $f$ such that the diagram (4.4) is commutative.

Relations $\leq_O$ and $\approx_O$ are called O-*relative order* and O-*relative equivalence*, respectively.

**Lemma 4.4.** For any categorical information operator O, the relation $\leq_O$ is a partial preorder on the class of all objects in the category **C**.

Proof. (a) For any categorical information operator O and any object $A$ from the category **C**, we have $A \leq_O A$ because for any object $A$ from **C**, $O(A) = O(A) \circ 1_A$ (*cf.* Diagram (4.5)).

$$\begin{tikzcd} A \arrow[d, "1_A"'] \arrow[dr, "O(A)"] & \\ A \arrow[r, "O(A)"'] & C \end{tikzcd} \tag{4.5}$$

(b) If $A \leq_O B$ and $B \leq_O D$, then $O(A) = O(B) \circ f$ for some morphism $f$ and $O(B) = O(D) \circ g$ for some morphism $g$ . This gives the following commutative diagram

$$\begin{tikzcd} A \arrow[d, "f"'] \arrow[dr, "\mathrm{O}(A)"] & \\ B \arrow[r, "\mathrm{O}(B)"] \arrow[d, "g"'] & C \\ D \arrow[ur, "\mathrm{O}(D)"'] & \end{tikzcd} \tag{4.6}$$

Consequently, we have

$$\mathrm{O}(A) = \mathrm{O}(B) \circ f = \mathrm{O}(D) \circ g \circ f = \mathrm{O}(D) \circ h$$

where $h = g \circ f$. Consequently, $A \leq_{\mathrm{O}} B$.

Lemma is proved.

**Proposition 4.8.** The relation $\approx_{\mathrm{O}}$ is an equivalence on the class Ob **C** of all objects from **C**, *i.e.*, the relation $\approx_{\mathrm{O}}$ is idempotent, symmetric and transitive.

Proof is similar to the proof of Proposition 4.1.

Definition 4.7 and Proposition 6.13 from [27] imply the following result.

**Proposition 4.9.** If $A \leq_{\mathrm{O}} B$ and $\mathrm{O}(A)$ is an epimorphism, then $\mathrm{O}(B)$ is also an epimorphism.

Definition 4.7 and Proposition 5.11 from [27] imply the following result.

**Proposition 4.10.** If $A \leq_{\mathrm{O}} B$ and $\mathrm{O}(A)$ is a retraction, then $\mathrm{O}(B)$ is also a retraction.

There are different types of categorical information operators.

**Definition 4.8.** A categorical information operator $\mathrm{Op}(I)$ is called:

a) *Monoperator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are monomorphisms.

b) *Epoperator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are epimorphisms.

c) *Bimoperator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are bimorphisms.

d) *Secoperator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are sections.

e) *Retroperator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are retractions.

f) *Isoperator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are isomorphisms.

g) A *constant operator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are constants.

h) A *co-constant operator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are co-constants.

i) A *zero operator* if all morphisms $\mathrm{Op}(I)(A)$ with $A \in \mathrm{Ob}\ \mathbf{C}$ are zeroes.

Informally, we have the following interpretation of the introduced types of information operators:

Information monoperators preserve distinctions between previously accepted information portions.

Information epoperators preserve distinctions between next coming information portions.

Information bimoperators are both information monoperators and information epoperators.

Information retroperators represent such information portions $I$ the impact of which can be erased from system state by another information impact, *i.e.*, there is another information portions $J$ that moves the infological system $\mathrm{IF}(R)$ back to the previous state.

Information secoperators represent such information portions $I$ that act like an eraser of some previously received information.

Information isooperators are both information retroperators and information secoperators.

Information zero operators represent such information portions $I$ that erases all existed information reducing it to minimum.

Information constant operators represent such information portions $I$ that equalize all previously received information.

Information co-constant operators represent such information portions $I$ that equalize all further information.

**Example 4.12.** Taking a category **C** from Example 4.2 where objects are sets of propositions, or more generally, of knowledge items, and morphisms are transformations of these sets, we define the categorical information operator $O$ so that for each set $A$ of propositions, the corresponding morphism $O(A)$ is a deduction in a monotone logic. As we know, deduction in a monotone logic only adds new propositions (knowledge items) to the initial set. Thus, $O$ is a categorical information monoperator.

**Example 4.13.** Let us take a category **C** where objects are collections of books that some library $L$ has at different periods of time and morphisms are transformations of these collections that go from time to time. Assuming that this library never discards books, we see that any categorical information operator $O$ in **C** is a categorical information monoperator.

**Example 4.14.** Let us take a category **C** where objects are collections of software systems that some software depositary $L$ has at different periods of time and morphisms are transformations of these collections that go from time to time. We define the categorical information operator $O$ in the following way. For each set $A$ of software systems, the corresponding morphism $O(A)$ is validation of software systems from $A$, exclusion of invalid systems and elimination of copies of the same system. As software systems are never added by morphisms $O(A)$, we see that any categorical information operator $O$ in **C** is a categorical information epoperator.

**Proposition 4.12.** If $O_2$ is a categorical information monoperator and $O_1 \preccurlyeq O_2$, then $O_1$ is also a categorical information monoperator.

<u>Proof</u>. Let us take an object $A$ from the category **C** and two morphisms $h$: $D \to A$ and $g$: $D \to A$ such that $O_1(A) \circ h = O_1(A) \circ g$ (*cf.* Diagram (4.7)).

$$\begin{tikzcd} D \arrow[r, shift left, "h"] \arrow[r, shift right, "g"'] & A \arrow[d, "O_1(A)"'] \arrow[dr, "O_2(A)"] & \\ & C \arrow[r, "f"'] & B \end{tikzcd} \tag{4.7}$$

Then we have

$$O_2(A) \circ h = f \circ O_1(A) \circ h = f \circ O_1(A) \circ g = O_2(A) \circ g$$

As $O_2$ is a categorical information monoperator, $O_2(A)$ is a monomorphism. Consequently, $h = g$.

Proposition is proved because $A$ is an arbitrary object from the category **C** and $g$ and $h$ are arbitrary morphisms into $A$.

**Corollary 4.4.** If $O_2$ is a categorical information monoperator and $O_1 \subseteq O_2$, then $O_1$ is also a categorical information monoperator.

**Corollary 4.5.** A categorical information operator equivalent to a categorical information monoperator is itself a categorical information monoperator.

Definition 4.6 and Proposition 5.5 from [27] imply the following result.

**Proposition 4.13.** If $O_2$ is a categorical information secoperator and $O_1 \preccurlyeq O_2$, then $O_1$ is also a categorical information secoperator.

**Proposition 4.14.** If $O_1$ is a categorical information epoperator and $O_1 \preccurlyeq_{ISO} O_2$, then $O_2$ is also a categorical information epoperator.

Proof. Let us take an object $A$ from the category **C** and two morphisms $h$: $B \to D$ and $g$: $B \to D$ such that $O_2(A)$: $A \to B$, $h \circ O_2(A) = g \circ O_2(A)$ (*cf.* Diagram (4.8)).

$$\begin{tikzcd} A \arrow[d, "O_1(A)"'] \arrow[dr, "O_2(A)"] & & \\ C \arrow[r, "f"'] & B \arrow[r, shift left, "h"] \arrow[r, shift right, "g"'] & D \end{tikzcd} \tag{4.8}$$

Then we have

$$h \circ O_2(A) = h \circ f \circ O_1(A) = g \circ f \circ O_1(A) = g \circ O_2(A)$$

As $O_1$ is a categorical information epoperator, $O_1(A)$ is an epimorphism. Consequently, $h \circ f = g \circ f$. As $f$ is an isomorphism, there is a morphism $t$: $B \to C$ such that $f \circ r = 1_B$. Consequently,

$$h = h \circ 1_B = h \circ f \circ r = g \circ f \circ r = g \circ 1_B = g$$

As $g$ and $h$ are arbitrary morphisms from $B$, $O_2(A)$ is an epimorphism.

Proposition is proved because $A$ is an arbitrary object from the category **C** and $g$ and $h$ are arbitrary morphisms from $B$.

**Corollary 4.6.** If $O_1$ is a categorical information epoperator, $O_2$ is a categorical information monoperator and $O_1 \preccurlyeq_{ISO} O_2$, then both $O_1$ and $O_2$ are categorical information bimoperators.

**Corollary 4.7.** A categorical information operator ISO-equivalent to a categorical information epoperator is itself a categorical information epoperator.

**Corollary 4.8.** A categorical information operator ISO-equivalent to a categorical information bimoperator is itself a categorical information bimoperator.

**Proposition 4.15.** If $O_1$ is a constant categorical information operator and $O_1 \preccurlyeq O_2$, then $O_2$ is also a constant categorical information operator.

Proof. Let us take an object $A$ from the category **C** and two morphisms $h$: $C \to A$ and $g$: $C \to A$ (*cf.* Diagram (4.9)).

$$\begin{tikzcd} D \arrow[r, shift left, "h"] \arrow[r, shift right, "g"'] & A \arrow[d, "O_1(A)"'] \arrow[dr, "O_2(A)"] & \\ & C \arrow[r, "f"'] & B \end{tikzcd} \tag{4.9}$$

As $O_1$ is a constant categorical information operator, we have

$$O_1(A) \circ h = O_1(A) \circ g$$

Consequently, we have

$$O_2(A) \circ h = f \circ O_1(A) \circ h = f \circ O_1(A) \circ g = O_2(A) \circ g$$

$O_2$ is also a constant categorical information operator because $A$ is an arbitrary object from the category **C** and $g$ and $h$ are arbitrary morphisms into $A$.

Proposition is proved.

**Corollary 4.9.** If $O_1$ is a constant categorical information operator and $O_1 \subseteq O_2$ , then $O_2$ is also a constant categorical information operator.

**Corollary 4.10.** A categorical information operator equivalent to a constant categorical information operator is itself a constant categorical information operator.

**Proposition 4.16.** If $O_1$ is a co-constant categorical information operator and $O_1 \preccurlyeq_{ISO} O_2$, then $O_2$ is also a co-constant categorical information operator.

<u>Proof</u>. Let us take an object $A$ from the category **C** and two morphisms $h$: $B \to D$ and $g$: $B \to D$ such that $O_2(A)$: $A \to B$. As $O_1 \preccurlyeq_{ISO} O_2$, we have $O_2(A) = f \circ O_1(A)$ for an isomorphism $f$ (*cf.* Diagram (4.10)).

$$\begin{tikzcd} A \arrow[d, "O_1(A)"'] \arrow[dr, "O_2(A)"] & & \\ C \arrow[r, "f"'] & B \arrow[r, shift left, "h"] \arrow[r, shift right, "g"'] & D \end{tikzcd} \tag{4.10}$$

As $O_1$ is a co-constant categorical information operator, we have

$$h \circ f \circ O_1(A) = g \circ f \circ O_1(A)$$

Thus, we obtain

$$h \circ O_2(A) = h \circ f \circ O_1(A) = g \circ f \circ O_1(A) = g \circ O_2(A)$$

$O_2$ is a co-constant categorical information operator because $A$ is an arbitrary object from the category **C** and $g$ and $h$ are arbitrary morphisms from $B$.

Proposition is proved.

**Corollary 4.11.** An information suboperator of a zero categorical information operator is itself a zero categorical information operator.

**Corollary 4.12.** An information operator equivalent to a zero categorical information operator is itself a zero categorical information operator.

**Corollary 4.13.** An information operator equivalent to a co-constant categorical information operator is itself a co-constant categorical information operator.

**Corollary 4.14.** If $O_1$ is a constant (co-constant) categorical information operator and $O_1 \subseteq O_2$, then $O_2$ is also a constant (co-constant) categorical information operator.

Information is a dynamic essence and its processing involves different operations with information and its representations and carriers. In the mathematical setting of categories, operations with information are represented by operations with information operators. One of the most important information operations is the (sequential) composition of categorical information operators.

Let us find some properties of the (sequential) composition of categorical information operators.

**Proposition 4.17.** The (sequential) composition of total categorical information operators is a total categorical information operator.

**Proposition 4.18.** If $O_3$ is a categorical information secoperator, then $O_3 \circ O_1 \preccurlyeq O_3 \circ O_2$ if and only if $O_1 \preccurlyeq O_2$.

<u>Proof</u>. 1. Let us assume that $O_1 \preccurlyeq O_2$. Then for an arbitrary object $A$ from the category **C**, we have the commutative Diagram (4.11):

$$\begin{tikzcd}
A \arrow[r] \arrow[dr, "O_1(A)"'] & B \arrow[r, "O_3(B)"] & D \\
& C \arrow[u, "f"'] \arrow[r, "O_3(C)"'] & H
\end{tikzcd} \tag{4.11}$$

As $O_3$ is a categorical information secoperator, the morphism $O_3(C)$ is a section. Consequently, there is a morphism $g$: $H \to C$, such that $g \circ O_3(C) = 1_C$.

Thus, we have

$$O_3(B) \circ O_2(A) = O_3(B) \circ f \circ O_1(A) = O_3(B) \circ f \circ 1_C \circ O_1(A) =$$
$$O_3(B) \circ f \circ g \circ O_3(C) \circ O_1(A) = t \circ O_3(C) \circ O_1(A)$$

where $t = O_3(B) \circ f \circ g$. As $A$ is an arbitrary object from the category **C**, we have

$$O_3 \circ O_1 \preccurlyeq O_3 \circ O_2$$

2. Now let us assume that $O_3 \circ O_1 \preccurlyeq O_3 \circ O_2$. Then we have the commutative Diagram (4.12):

$$\begin{tikzcd}
A \arrow[r] \arrow[dr, "O_1(A)"'] & B \arrow[r, "O_3(B)"] & D \\
& C \arrow[r, "O_3(C)"'] & H \arrow[u, "f"']
\end{tikzcd} \tag{4.12}$$

As $O_3$ is a categorical information secoperator, the morphism $O_3(B)$ is a section. Consequently, there is a morphism $g$: $H \to C$, such that $g \circ O_3(B) = 1_B$ .

Thus, the equality

$$O_3(B) \circ O_2(A) = f \circ O_3(C) \circ O_1(A)$$

implies the equalities

$$g \circ O_3(B) \circ O_2(A) = g \circ f \circ O_3(C) \circ O_1(A) =$$
$$q \circ O_1(A) = 1_B \circ O_2(A) = O_2(A)$$

where $q = g \circ f \circ O_3(C)$. As $A$ is an arbitrary object from the category **C**, we have

$$O_1 \preccurlyeq O_2$$

Proposition is proved.

**Corollary 4.15.** If $O_3$ is a categorical information secoperator, then $O_3 \circ O_1 \asymp O_3 \circ O_2$ if and only if $O_1 \asymp O_2$.

Proof. 1. Let us take categorical information operators $O_1$, $O_2$ and $O_3$ and assume that $O_1 \asymp O_2$ and $O_3$ is a categorical information secoperator. Then by Definition 4.6, we have $O_1 \preccurlyeq O_2$. Thus, by Proposition 4.18, $O_3 \circ O_1 \preccurlyeq O_3 \circ O_2$.

Besides, $O_1 \asymp O_2$ implies $O_2 \preccurlyeq O_1$. Thus, by Proposition 4.18, $O_3 \circ O_2 \preccurlyeq O_3 \circ O_1$. Then by Definition 4.6, we have $O_3 \circ O_1 \asymp O_3 \circ O_2$.

2. Let us take categorical information operators $O_1$, $O_2$ and $O_3$ and assume that $O_3 \circ O_1 \asymp O_3 \circ O_2$ and $O_3$ is a categorical information secoperator. Then by Definition 4.6, $O_3 \circ O_1 \preccurlyeq O_3 \circ O_2$. Thus, by Proposition 4.18, $O_1 \preccurlyeq O_2$.

Besides, $O_3 \circ O_1 \asymp O_3 \circ O_2$ implies $O_3 \circ O_2 \preccurlyeq O_3 \circ O_1$ . Thus, by Proposition 4.18, $O_2 \preccurlyeq O_1$. Then by Definition 4.6, $O_1 \asymp O_2$.

Corollary is proved.

**Proposition 4.19.** If $O_3$ is a categorical information isoperator, then $O_3 \circ O_1 \preccurlyeq_{ISO} O_3 \circ O_2$ if and only if $O_1 \preccurlyeq_{ISO} O_2$.

Proof. 1. Let us take categorical information operators $O_1$, $O_2$ and $O_3$ and assume that $O_1 \preccurlyeq_{ISO} O_2$ and $O_3$ is a categorical information isoperator. Then we have the commutative diagram (4.13) in which $f$ is an isomorphism:

$$\begin{tikzcd}
A \arrow[r, "O_3(B)"] \arrow[dr, "O_1(A)"'] & B \arrow[r] & D \\
& C \arrow[u, "f"'] \arrow[r, "O_3(C)"'] \arrow[ur, "O_2(A)"] & H
\end{tikzcd} \tag{4.13}$$

By Proposition 4.18, it is possible to extend Diagram (4.13) to the commutative Diagram (4.14):

$$\begin{tikzcd} A \arrow[r] \arrow[dr, "O_1(A)"'] & B \arrow[r, "O_3(B)"] & D \\ & C \arrow[u, "O_2(A)", "f"'] \arrow[r, "O_3(C)"'] & H \arrow[u, "t"'] \end{tikzcd} \tag{4.14}$$

where $t = O_3(B) \circ f \circ g$ and $g$ is the inverse to the morphism $O_3(C)$. As $O_3$ is a categorical information isooperator, $O_3(C)$ and $O_3(B)$ are isomorphisms. Consequently, $g$ is an isomorphism [27]. As $f$ is an isomorphism, the morphism $t$ is also an isomorphism as the sequential composition of isomorphisms [27]. It means that $O_3(B) \circ O_2(A) = t \circ O_3(C) \circ O_1(A)$ where $t$ is an isomorphism.

As $A$ is an arbitrary object from the category **C**, we have

$$O_3 \circ O_1 \preccurlyeq_{ISO} O_3 \circ O_2$$

2. Let us take categorical information operators $O_1$, $O_2$ and $O_3$ and assume that $O_3 \circ O_1 \preccurlyeq_{ISO} O_3 \circ O_2$ and $O_3$ is a categorical information isoperator. Then we have the commutative diagram (4.15) in which $f$ is an isomorphism:

$$\begin{tikzcd} A \arrow[r] \arrow[dr, "O_1(A)"'] & B \arrow[r, "O_3(B)"] & D \\ & C \arrow[u, "O_2(A)", "f"'] \arrow[r, "O_3(C)"'] & H \arrow[u] \end{tikzcd} \tag{4.15}$$

By Proposition 4.18, it is possible to extend Diagram (4.15) to the commutative Diagram (4.16):

$$\begin{tikzcd} A \arrow[r] \arrow[dr, "O_1(A)"'] & B \arrow[r, "O_3(B)"] & D \\ & C \arrow[u, "O_2(A)", "q"'] \arrow[r, "O_3(C)"'] & H \arrow[u, "f"'] \end{tikzcd} \tag{4.16}$$

where $q = g \circ f \circ O_3(C)$. and $g$ is the inverse to the morphism $O_3(B)$. As $O_3$ is a categorical information isooperator, $O_3(C)$ and $O_3(B)$ are isomorphisms. Consequently, $g$ is an isomorphism [27]. As $f$ is an isomorphism, the morphism $q$ is also an isomorphism as the sequential composition of isomorphisms [27]. It means that $O_2(A) = q \circ O_1(A)$ where $q$ is an isomorphism.

As $A$ is an arbitrary object from the category **C**, we have

$$O_1 \preccurlyeq_{ISO} O_2$$

Proposition is proved.

Let us take two categorical information spaces **C** and **K**.

**Theorem 4.1.** A covariant functor $F$: **C** → **K** associates categorical information operators over **K** to categorical information operators over **C** and preserves their sequential composition.

Proof. Let us consider a covariant functor $F$: **C** → **K**. To each object $A$ from **C**, the functor $F$ assigns the object $F(A)$ from **K**, and if O is a categorical information operator, then $F$ assigns the morphism $F(\mathrm{O}(A))$ from Mor **K** to each morphism $\mathrm{O}(A)$ from **C**. Assuming axiom of choice for the class Ob **C**, it is possible to choose a unique object $D_A$ from the class $\mathrm{F}_A = \{ C; C \in \mathrm{Ob}\ \mathbf{C}$ and $F(C) = F(A) \}$. Then we define $F\mathrm{O}(F(A))$ as the morphism $F(\mathrm{O}(D_A))$. Note that $F(D_A) = F(A)$. In such a way, we build a categorical information operator $F$O over **K**. By definition (*cf.* Section 2), functors preserve sequential composition. Thus, sequential composition of categorical information operators is also preserved.

Theorem is proved.

Informally, functors are mapping of categorical information spaces that preserve the structure of information transformations. In particular, Theorem 4.1 shows that functors are mappings between categories of states of different infological systems that are compatible with actions of information. Functors allow one to study how the same information operates in different infological systems.

As we can see from the proof of Theorem 4.1, in a general case, it is possible to associate different categorical information operators over **K** to the same categorical information operator over **C**, *i.e.*, correspondence between categorical information operators over **K** and categorical information operators over **C** is not a function. To make it a function, we need additional conditions on the functor $F$.

**Theorem 4.2.** If $F$ is an embedding covariant functor, then a unique categorical information operator $F$O over **K** is associated to each categorical information operator O over **C** and if O is total, then $F$O is total.

Proof. Let us consider an embedding covariant functor $F$: **C** → **K**. To each object $A$ from **C**, the functor $F$ assigns the object $F(A)$ from **K**, and if O is a categorical information operator, then $F$ assigns the morphism $F(\mathrm{O}(A))$ from Mor **K** to each morphism $\mathrm{O}(A)$ from **C**. Because $F$ is an embedding covariant functor, it defines a one-to-one correspondence between objects from **C** and objects from **K** [27; Section 12]. Thus, there is only one object $X$ in the class Ob **C** such that $F(X) = F(A)$ and we define $F\mathrm{O}(F(A))$ as the morphism $F(\mathrm{O}(A))$. In such a way, we build a unique categorical information operator $F$O over **K**. As O is total and $F_{\mathrm{Ob}\mathbf{C}}$: Ob **C** → Ob **K** is a one-to-one correspondence, the operator $F$O is also total.

Theorem is proved.

**Theorem 4.3.** A full dense covariant functor $F$: **C** → **K** maps categorical information operators over **K** into categorical information operators over **C** and preserves totality of categorical information operators.

Proof. Let us consider a full dense covariant functor $F$: **C** → **K** and a categorical information operator O over **K**. To each object $D$ from **K**, the operator O assigns the morphism $\mathrm{O}(D)$ from **K**. Besides, to each object $A$ from **C**, the functor $F$ assigns the object $F(A)$ from **K**. If $C = \mathrm{Im}\ \mathrm{O}F(A)$, then there is an object $B$ from **C**, such that $F(B)$ is isomorphic to $C$ because $F$ is a dense functor, *i.e.*, there is an isomorphism $f$: $C \to F(B)$. This gives us the following diagram

$$\begin{tikzcd} A \arrow[d, "F"] & B \arrow[d, "F"'] \\ F(A) \arrow[r, "f \circ O(F(A))"'] & F(B) \end{tikzcd} \tag{4.17}$$

Because $F$ is a full functor, there is a morphism $g$: $A \to B$ such that $F(g) = f \circ O(F(A)))$. This gives us the following diagram

$$\begin{tikzcd} A \arrow[r, "g"] \arrow[d, "F"] & B \arrow[d, "F"'] \\ F(A) \arrow[r, "f \circ O(F(A))"'] & F(B) \end{tikzcd} \tag{4.18}$$

Thus, we define $U(A) = g$. This is done for each object $A$ in the class Ob **C**, and such a way we obtain a categorical information operator $U$ over **C**. As O is total and $F_{\mathrm{ObC}}$: Ob **C** $\to$ Ob **K** is a one-to-one correspondence, the operator $U$ is also total. We denote the categorical information operator $U$ by $F^{-1}$O.

Theorem is proved.

Let us take a property $P$ of categorical information operators.

**Definition 4.9.** A functor $F$: **C** $\to$ **K**:

a) *preserves* the property $P$ if for any categorical information operator $A$ over **C** with the property $P$, its image $F(A)$ is a categorical information operator over **K** with the property $P$.

a) *reflects* the property $P$ if for any categorical information operator $A$ over **C**, if its image $F(A)$ has the property $P$, then $A$ also has the property $P$.

For instance, by definition, a covariant functor preserves the property of a morphism to be a composition of other morphisms.

**Proposition 4.20.** Every covariant functor $F$: **C** $\to$ **K** preserves the relation $\preccurlyeq$ between categorical information operators.

Indeed, if $\mathrm{O}_1 \preccurlyeq \mathrm{O}_2$ for categorical information operators $\mathrm{O}_1$ and $\mathrm{O}_2$, then by Definition 4.6, for any object $A$ from **C**, we have

$$\mathrm{O}_2(A) = f \circ \mathrm{O}_1(A) \text{ for some morphism } f$$

Because $F$ is a functor, we also have

$$F\mathrm{O}_2(F(A)) = F(f) \circ F\mathrm{O}_1(F(A))$$

As $F(A)$ is an arbitrary object where categorical information operators $F\mathrm{O}_1$ and $F\mathrm{O}_2$ are defined, we have $F\mathrm{O}_1 \preccurlyeq F\mathrm{O}_2$.

Proposition 4.20 and Proposition 12.2 from [27] imply the following result.

**Proposition 4.21.** Every covariant functor $F$: **C** $\to$ **K** preserves relations $\preccurlyeq_{\mathrm{RET}}$, $\preccurlyeq_{\mathrm{SEC}}$, and $\preccurlyeq_{\mathrm{ISO}}$ between categorical information operators.

**Proposition 4.22.** If $A \leq_{\mathrm{O}} B$, then $FA \leq_{F\mathrm{O}} FB$.

Indeed, the commutative Diagram 4.4 in the category **C** implies the commutative Diagram 4.19 in the category **K**.

$$\begin{tikzcd} F(A) \arrow[d, "F(f)"'] \arrow[dr, "F\mathrm{O}(A)"] & \\ F(B) \arrow[r, "\mathrm{O}(B)"'] & F(C) \end{tikzcd} \tag{4.19}$$

**Proposition 4.23.** For any categorical information operators $\mathrm{O}_1$ and $\mathrm{O}_2$ over **C**, if $A \leq_{\mathrm{O}1} B$, then $A \leq_{\mathrm{O}2 \circ \mathrm{O}1} B$.

Indeed, the commutative Diagram 4.4 implies the commutative Diagram 4.20.

$$\begin{tikzcd} A \arrow[d, "f"'] \arrow[dr, "\mathrm{O}_1(A)"] & & \\ B \arrow[r, "\mathrm{O}_1(B)"'] & C \arrow[r, "\mathrm{O}_2(B)"'] & D \end{tikzcd} \tag{4.20}$$

Proposition 12.2 from [27] implies the following result.

**Theorem 4.4.** Every covariant functor $F$: **C** $\rightarrow$ **K** preserves categorical information secoperators, categorical information retroperators, and categorical information isoperators over the category **C**.

Theorem 12.10 from [27] implies the following result.

**Theorem 4.5.** Every dense full and faithful covariant functor $F$: **C** $\rightarrow$ **K** preserves categorical information monooperators, categorical information epoperators, categorical information bimoperators over the category **C**.

Propositions 12.8 and 12.9 from [27] imply the following result.

**Theorem 4.6.** Every full faithful covariant functor $F$: **C** $\rightarrow$ **K** reflects categorical information secoperators, categorical information retroperators, categorical information monoperators, categorical information epoperators, categorical information bimoperators, constant categorical information operators, co-constant categorical information operators, zero categorical information operators, and categorical information isoperators over the category **C**.

In addition to sequential composition, it is possible to define concurrent and parallel compositions of categorical information operators.

Let us consider a category **C** with pushouts [27].

**Definition 4.10.** A categorical information operator $O$ over **C** is called the *free sum* of categorical information operators $O_1$ and $O_2$ over **C** if for any $A \in$ Ob **C**, there is the pushout (4.21) where $q = g \circ O_1(A) = f \circ O_2(A)$ and $\mathrm{O}(A) = q$.

$$\begin{tikzcd} B \arrow[r, "g"] & D \\ A \arrow[u, "O_1(A)"] \arrow[ur, "q"] \arrow[r, "O_2(A)"'] & C \arrow[u, "f"'] \end{tikzcd} \tag{4.21}$$

The free sum of categorical information operators $O_1$ and $O_2$ over **C** is denoted by $O_1 \circledast O_2$.

By properties of pushouts [27] the free sum of categorical information operators is defined uniquely up to an isomorphism.

On the level of information processes, the free sum of categorical information operators represents consistent integration of information. Now when people and databases receive information from diverse sources, information integration has become an extremely important cognitive operation. Information integration plays a mission critical role in a diversity of applications from life sciences to E-Commerce to ecology to disaster management. These applications rely on the ability to integrate information from multiple heterogeneous sources.

Operationally, Definition 4.3 means that the free sum of two categorical information operators $O_1$ and $O_2$ is constructed by taking pushouts for couples of actions of $O_1$ and $O_2$ on each object in **C**. It models concurrent information processing (e.g., integration and composition).

**Theorem 4.7.** The free sum of any two categorical information operators is a categorical information epoperator.

Proof. Let us take the free sum $O$ of two categorical information operators $O_1$ and $O_2$ over **C**. Then for any object $A \in \mathrm{Ob}\ \mathbf{C}$, there is the pushout (4.21). To prove that $O$ is a categorical information epoperator, we need to show that $q$ always is an epimorphism.

Let us consider two morphisms $k, h \in \mathrm{Hom}_{\mathbf{C}}(D, Q)$ such that $k \circ q = h \circ q$. Then we have

$$k \circ q = k \circ f \circ O_2(A) = k \circ g \circ O_1(A) = h \circ q = h \circ f \circ O_2(A) = h \circ g \circ O_1(A)$$

It gives us the commutative diagram (4.22).

$$\begin{tikzcd} B \arrow[r, "k \circ g"] & Q \\ A \arrow[u, "O_1(A)"] \arrow[ur, "k \circ q = h \circ q"] \arrow[r, "O_2(A)"'] & C \arrow[u, "h \circ f"'] \end{tikzcd} \tag{4.22}$$

By the definition of a pushout (*cf.* for example, [27]), we have a morphism $p$: $D \to Q$, which is defined in a unique way and for which

$$p \circ f \circ O_2(A) = p \circ g \circ O_1(A) \tag{4.23}$$

As the equality (4.23) is also true for morphisms $k$ and $h$ and morphism $p$ is unique, we have $k = h$.

Theorem is proved because the object $A \in \mathrm{Ob}\ \mathbf{C}$ and morphisms $k$ and $h$ were chosen in an arbitrary way.

The free sum preserves relations between categorical information operators.

**Theorem 4.8.** If $O_1 \preccurlyeq O_3$ and $O_2 \preccurlyeq O_4$, then $O_1 \circledast O_2 \preccurlyeq O_3 \circledast O_4$.

Proof. Let us take categorical information operators $O_1$, $O_2$, $O_3$ and $O_4$ and assume that $O_1 \preccurlyeq O_3$ and $O_2 \preccurlyeq O_4$. For each pair of them, we build their free sums $O_1 \circledast O_2$ and $O_3 \circledast O_4$, presented for an object $A$ in diagrams (4.24) and (4.25).

$$\begin{tikzcd} B \arrow[r, "g"] & D \\ A \arrow[u, "O_1(A)"] \arrow[ur, "O_1 \circledast O_2(A)"'] \arrow[r, "O_2(A)"'] & C \arrow[u, "f"'] \end{tikzcd} \tag{4.24}$$

$$\begin{tikzcd} H \arrow[r, "k"] & U \\ A \arrow[u, "O_3(A)"] \arrow[ur, "O_3 \circledast O_4(A)"'] \arrow[r, "O_4(A)"'] & K \arrow[u, "h"'] \end{tikzcd} \tag{4.25}$$

As $O_1 \preccurlyeq O_3$ and $O_2 \preccurlyeq O_4$, by Definition 4.6, we have:

$$O_3(A) = p \circ O_1(A)$$

and

$$O_4(A) = q \circ O_2(A)$$

for some morphisms $f$ and $g$.

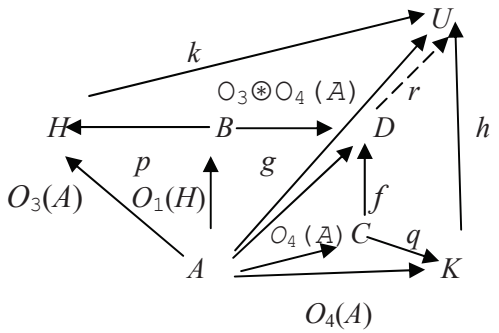
This gives us the following commutative diagram

$$\begin{tikzcd} & & U & \\ H & B \arrow[l, "p"'] \arrow[r, "g"] & D \arrow[ur, "O_3 \circledast O_4(A)"] & \\ & A \arrow[ul, "O_3(A)"] \arrow[u, "O_1(H)"] \arrow[uur, "k"] \arrow[uurr, "O_3 \circledast O_4(A)"] \arrow[r, "O_4(A)"] \arrow[urr, "O_4(A)"] \arrow[rr, bend right, "O_4(A)"'] & C \arrow[u, "f"'] \arrow[r, "q"] & K \arrow[uu, "h"'] \end{tikzcd} \tag{4.26}$$

From Diagram (4.24), we obtain the following commutative square

$$\begin{tikzcd} B \arrow[r, "k \circ p"] & U \\ A \arrow[u, "O_1(A)"] \arrow[r, "O_2(A)"'] & C \arrow[u, "h \circ q"'] \end{tikzcd} \tag{4.27}$$

As the commutative square in Diagram (4.24) is a pushout, there is a morphism $r$: $D \to U$, such that $k \circ p = r \circ g$ and $h \circ q = r \circ f$. This gives us the following commutative diagram




(4.28)

From Diagram (4.28), we have

$$O_3(A) \circledast O_4(A) = r \circ O_1(A) \circledast O_2(A)$$

Thus, $O_1 \circledast O_2 \preccurlyeq O_3 \circledast O_4$.

Theorem is proved.

Definitions imply the following result.

**Theorem 4.9.** For any categorical information operators $O_1$ and $O_2$, we have $O_1 \preccurlyeq O_1 \circledast O_2$ and $O_2 \preccurlyeq O_1 \circledast O_2$.

Indeed, $O_1 \circledast O_2(A) = g \circ O_1(A) = f \circ O_2(A)$ for any object $A$ from the category **C**.

## 5. Conclusions

Using the operator model of information, we studied decompositions of portions of information and relation of complementarity between portions of information, which are represented by relations between categorical information operators. These relations are preserved by covariant functors, sequential compositions and free sums of categorical information operators.

Categorical representation of information processes can solve different problems and clarify general misconceptions. One of them is related to computers.

As we well understand, computers process information. However, what we see is only data processing. Indeed, computers get data as their input and give data as their output. As the rule, these data have the symbolic form. This is the observed picture.

It is similar to dynamics of material bodies. People see movement of bodies but to find energy or acceleration, it is necessary to perform measurements and calculations, representing observable quantities by relevant structures in Newtonian dynamics.

In a similar way, we represent information by information operators (in this paper, by categorical information operators) and information transformations by information metaoperators (in this paper, by functors). This model makes it possible to describe information processing by compositions of information metaoperators (functors), *i.e.*, by the sequential composition of information metaoperators (functors). In such a way, categorical model allows researchers to describe in a theoretical form information processing and not only data processing.

There are operations with categorical information operators and relations between these operators that are not studied in this paper but are important for information theory. This brings us to the following problems.

**Problem 1.** Find necessary and/or sufficient conditions when an operation with categorical information operators preserves given relations between these operators.

**Problem 2.** Find conditions when an operation with categorical information operators implies prescribed relations between these operators.

**Problem 3.** Find conditions when categorical information operators form a lattice with respect to the relation $\preccurlyeq$.

**Problem 4.** Characterize classes K of morphisms in a category **C** such that categorical information operators in **C** form a lattice with respect to the relation $\preccurlyeq_K$.

In this paper, relations between categorical information operators and covariant functors are studied.

**Problem 5.** Study relations between categorical information operators and contravariant functors.

## References and Notes


1. Agre, P. Institutional Circuitry: Thinking about the Forms and Uses of Information. *Inform. Technol. Libr.* **1995**, *December*, 225-230.
2. Capurro, R.; Fleissner, P.; Hofkirchner, W. Is a Unified Theory of Information Feasible? In *The Quest for a UnifiedTtheory of Information*, Proceedings of the 2nd International Conference on the Foundations of Information Science; Gordon and Breach Publ.: Amsterdam, The Netherlands, 1999; pp. 9-30.
3. Flükiger, D.F. *Contributions towards a Unified Concept of Information*, (Doctoral Thesis); University of Berne: Bern, Switzerland 1995.
4. Shannon, C.E. Collected Papers; Sloane, N.J.A., Wyner, A.D., Eds.; IEEE Press: New York, NY, USA, 1993.
5. Gofman, W. Information Science: Discipline or Disappearence? *Aslib Proc*. **1970**, *22*, 589-585.
6. Gilligan, J. Patterns on glass: The language games of information. *Symposium on the Philosophical and Logical Aspects of Information Systems*; University of the West of England: Bristol, UK, 1995; pp. 20-22.
7. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: New York, NY, USA, 2010.
8. Baez, J.; Stay, M. Physics, Topology, Logic and Computation: A Rosetta Stone. *Quantum Phys.* (Preprint) **2009**, arXiv:quant-ph/0903.0340.
9. Ehresmann, A.C.; Vanbremeersch, J.-P. Hierarchical Evolutive Systems: A mathematical model for complex systems. *Bull. Math. Biology*. **1987**, *49*, 13-50.
10. Ehresmann, A.C.; Vanbremeersch J.P. *Memory Evolutive Systems*: *Hierarchy, Emergence, Cognition*, Elsevier Science, Amsterdam, The Netherlands, 2007.
11. Levich, A.P.; Solov'yov, A.V. Category-Functor Modeling of Natural Systems. *Cybern Syst*, **1999**, *30*, 571-585.
12. Anderson, B.D.O. A control theorist looks at abstract nonsense. In *Proceedings of the First International Symposium*, *Lecture Notes in Computer Science*; Springer: New York, NY, USA, 1975; pp. 35-50.

13. Arbib, M.A.; Manes, E.G. A categorist's view of automata and systems, in *Proceedings of the First International Symposium*, *Lecture Notes in Computer Science,* Springer: New York, USA, 1975; Volume 25, pp. 51-64.
14. Ehrig, H.; Kreowski, H.-J. Power and initial automata in pseudoclosed categories, in *Proceedings of the First International Symposium*, *Lecture Notes in Computer Science*, Springer: New York, USA, 1975; Volume 25, pp. 144-150.
15. Manin, Y.I. Classical computing, quantum computing, and Shor's factoring algorithm. *Quantum Phys.* (Preprint) **1999**, arXiv:quant-ph/9903008.
16. Manin, Yu. I. Renormalization and computation I: motivation and background. *Quantum Algebr.* (Preprint) **2009**, arXiv: math.QA/0904.4921.
17. Moggi, E. Notions of computation and monads, *Information and Computation*, **1991**, *93*, 55-92.
18. Power, J.; Robinson, E. Premonoidal categories and notions of computation. *Math. Structures Comput. Sci*. **1997**, *7*, 453-468.
19. Scott, P. Some aspects of categories in computer science, in *Handbook of Algebra*; Elsevier: Amsterdam, The Netherlands, **2000**; Volume 2.
20. Abramsky, S.; Duncan, R. A categorical quantum logic. *Math. Structures Comput. Sci*. **2005**, *16*, 469-489.
21. Barr, M.; Wells, C. *Toposes, Triples and Theories*, Springer: Berlin, Germany, 1983.
22. Johnstone, P.T. *Topos Theory*, Academic Press: New York, USA, 1977.
23. Lambek, J.; Scott, P.J. *Introduction to Higher-order Categorical Logic*, Cambridge U. Press: Cambridge, UK, 1986.
24. Goldblatt, R. *Topoi*: *The Categorical analysis of Logic*, North-Holland P.C., Amsterdam, 1979.
25. Burgin, M. Information and Transformation. *Transformation* **1998/1999**, *1*, 48-53 (in Polish).
26. Burgin, M. Categories with involution and relations in γ-categories. *Transactions of the Moscow Mathematical Society* **1970**, *22*, 161-228 (translated from Russian).
27. Herrlich, H.; Strecker, G.E. *Category Theory*, Allyn and Bacon Inc.: Boston, USA, 1973.
28. Minsky, M. *The Society of Mind*, Simon and Schuster, New York, 1986.
29. Melik-Gaikazyan, I. V. *Information processes and reality*, Nauka, Moscow, 1997 (in Russian, English summary).
30. Burgin, M. Information: Problem, Paradoxes, and Solutions. *TripleC* **2003**, *1*, 53-70.
31. Mizzaro, S. Towards a theory of epistemic information, in *Information Modelling and Knowledge Bases*, IOS Press: Amsterdam, The Netherlands, 2001; Volume 12, pp. 1-20.
32. Shreider, Y.A. On Semantic Aspects of Information Theory. In *Information and Cybernetics*. Radio; Moscow, USSR , 1967; pp. 15-47 (in Russian).
33. Bar-Hillel, Y.; Carnap, R. Semantic information. *Brit. J. Phil. Sci.* **1958**, *4*, 147-157.
34. Halpern, J.Y.; Moses, Y. Towards a theory of knowledge and ignorance. in *Logics and Models of Concurrent Systems*; Springer-Verlag: New York, NY, USA, 1985; pp. 459-476.
35. Burgin, M. *Super-recursive Algorithms*; Springer: New York, NY, USA, 2005.
36. Cooper, W.S. *Foundations of Logico-Linguistics*, D. Reidel P.C.: Dordrecht, The Netherlands, 1978.
37. Burgin, M. Multiple computations and Kolmogorov complexity for such processes, *Notices of the Academy of Sciences of the USSR* **1983**, *27*, 793-797.

38. Barwise, J.; Seligman, J. *Information Flow*: *The Logic of Distributed Systems*, Cambridge Tracts in Theoretical Computer Science 44, Cambridge University Press: Cambridge, UK, 1997.
39. Rocchi, P.; Panella, O. Acquisition of Information is Achieved by the Measurement Process in Classical and Quantum Physics, in *AIP Conf. Proceedings*, Växjö, Sweden, 2007; Volume 962, pp. 206-214.
40. Bateson, G.; Ruesch, J. *Communication*: *The Social Matrix of Psychiatry*, W.W. Norton and Co.: New York, NY, USA, 1951.
41. Burgin, M. Is Information Some Kind of Data? Proceedings of the Third Conference on the Foundations of Information Science (FIS 2005), Paris, France, July 2005; pp. 1-31; available online: http://www.mdpi.net/fis2005/proceedings.html (last accessed on 17 November 2010).

© 2010 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).

*Information* **2011**, *2*, 697-726; doi:10.3390/info2040697

OPEN ACCESS

*information*

**ISSN 2078-2489**

www.mdpi.com/journal/information

*Article*

# Epistemic Information in Stratified M-Spaces

**Mark Burgin**

Department of Mathematics, University of California, Los Angeles, 405 Hilgard Ave. Los Angeles, CA 90095, USA; E-Mail: markburg@cs.ucla.edu

*Received: 15 September 2011; in revised form: 24 November 2011 / Accepted: 1 December 2011 / Published: 16 December 2011*

---

**Abstract:** Information is usually related to knowledge. However, the recent development of information theory demonstrated that information is a much broader concept, being actually present in and virtually related to everything. As a result, many unknown types and kinds of information have been discovered. Nevertheless, information that acts on knowledge, bringing new and updating existing knowledge, is of primary importance to people. It is called epistemic information, which is studied in this paper based on the general theory of information and further developing its mathematical stratum. As a synthetic approach, which reveals the essence of information, organizing and encompassing all main directions in information theory, the general theory of information provides efficient means for such a study. Different types of information dynamics representation use tools of mathematical disciplines such as the theory of categories, functional analysis, mathematical logic and algebra. Here we employ algebraic structures for exploration of information and knowledge dynamics. In Introduction (Section 1), we discuss previous studies of epistemic information. Section 2 gives a compressed description of the parametric phenomenological definition of information in the general theory of information. In Section 3, anthropic information, which is received, exchanged, processed and used by people is singled out and studied based on the Componential Triune Brain model. One of the basic forms of anthropic information called epistemic information, which is related to knowledge, is analyzed in Section 4. Mathematical models of epistemic information are studied in Section 5. In Conclusion, some open problems related to epistemic information are given.

**Keywords:** information; knowledge; information operator; stratification; epistemic information; composition; cognition; algebra; equivalence

---

## 1. Introduction

Explication and clarification of the concept of information requires not only elaboration and exploration of a general unified definition of information but also needs separation and examination of basic special types of information. The general theory of information gives the most general definition of information, organizing and encompassing all main directions in information theory [1]. According to the main principle of general theory of information, which impeccably correlates with people's practice and observations, when information comes to a system and is accepted, it causes changes in the system. This feature of information makes it reasonable to model information by operators and information dynamics by actions of these operators. This approach is adopted in the general theory of information.

In addition, the general theory of information provides efficient means for information classification and study of the basic information types. One of such types is information that is received, exchanged, processed and used by people. It is called anthropic information. We explain the methodological principles of the anthropic information explication and classification, which bring us to cognitive information and its important subclass called epistemic information.

Portions of epistemic information are modeled/represented by epistemic information operators acting in spaces of knowledge, which are represented by a formal construction called a Mizzaro space. These spaces consist of knowledge items often unified by structural relations.

In a general setting, epistemic information has been studied by different authors. Bar-Hillel and Carnap [2], Hintikka [3-5] and Israel and Perry [6] explored information in knowledge represented by means of mathematical logic. Shreider [7], Mackay [8], Brookes [9], Mizzaro [10,11] and Gackowski [12] base their theories on the following assumption:

*Information is a change in a knowledge system*

Later this principle has been made more exact [13] and formulated as

*Epistemic information is a change in a knowledge system*

The general theory of information [1] makes the next step to the better understanding of epistemic information. Namely, it is explained that

*Epistemic information is a capacity to cause changes*
*in a knowledge system and it is possible to measure*
*this capacity by changes in the knowledge system*
*impacted by information*

Such changes in the knowledge system of a system $R$ reflect accepted information. Note that information can be accepted not only as true information but also as false information. In this case, changes in the knowledge system can result in exclusion of some knowledge or in labeling this knowledge as false, e.g., treating it as a misconception or blunder.

The approach of Mizzaro [10,11,13] and Burgin [1] to epistemic information does not consider knowledge in general, but makes use of the term a *knowledge state* (KS) of a dynamic knowledge system or a cognitive agent. It is postulated that knowledge of a knowledge system/agent consists of atomic components called *knowledge items* (KI) and the number of these items in a portion of knowledge gives an adequate measure of this knowledge. Such knowledge states are set-theoretical or unstructured Mizzaro spaces defined in [1].

This is the first approximation to modeling information processes related to knowledge transformations. In it, the substantial dimensions of knowledge are reflected, while the relational dimensions, which describe relations between knowledge items, are studied in higher-level models of operational information theory, which goes beyond the scope of this paper. Taking into account such relations allows one to achieve higher precision in measuring information and knowledge. However, there are many situations where precision provided by quantities of knowledge items is sufficient. For instance, exactly this level of precision is successfully used in software engineering and technology where program instructions, which belong to the procedural type of knowledge, are used as knowledge items that determine measures of information and knowledge transformations [14].

Taking knowledge states as the base, Mizzaro [13] considers and utilizes natural set-theoretical operations, such as union, difference, complement, and intersection, as well as set-theoretical relations, such as inclusion, emptiness, and membership, on the set of all knowledge states. These and more sophisticated operations and relations are advanced to the level of local information operators and studied in [1]. Namely, in the context of the general theory of information, transformations in knowledge states caused by receiving data (messages) are represented by local information operators, which are projections of global information operators also described in [1]. Consequently, the theory of Mizzaro spaces and local information operators in these spaces is a localization of the mathematical (formalized) stratum of the general theory of information.

In this paper, we continue exploration of epistemic information, extending Mizzaro spaces to M-spaces, which represent and model information dynamics by algebraic structures. In addition to uniform Mizzaro spaces studied in [1,11,13], here we consider Mizzaro spaces and M-spaces with stratification and extend the scope of epistemic information operators. In [1,11,13], only two operators, addition and deletion are studied. Here we consider five basic epistemic information operators: addition, deletion, moving, replication, and transformation, as well as their compositions. Section 2 gives a compressed description of the parametric phenomenological definition of information in the general theory of information. In Section 3, anthropic information that is received, exchanged, processed and used by people is separated and studied based on the Componential Triune Brain model introduced in [15] and further developed in [1]. This model allows overcoming the pitfalls of the simple linear hierarchy suggested by MacLean [16,17] by considering not anatomically localized but distributed in the brain basic systems and including such an important psychological construct as the Will [18,19] in the brain structure. Based on this model, one of the basic forms of anthropic information called epistemic information, which is related to knowledge, is analyzed in Section 4. Mathematical models of epistemic information are studied in Section 5. In Conclusion, we give some open problems related to epistemic information.

## 2. The Concept of Information

Our study of epistemic information is based on the system of ontological principles from the general theory of information. All these principles serve as an extended definition of information. A concise definition is given in the second ontological principle, which has several forms.

**Ontological Principle O2 (the *General Transformation Principle*).** In a broad sense, *information* for a system $R$ is a capacity to cause changes in the system $R$.

Thus, we may understand information in a broad sense as a capacity (ability or potency) of things, both material and abstract, to change other things. Information exists in the form of *portions of information*. Informally, a portion of information is such information that can be separated from other information. Information is, as a rule, about something. What information is about is called an *object* of this information.

The Ontological Principle O2 has several consequences. First, it demonstrates that information is closely connected to transformation. Namely, it means that information and transformation are functionally similar because they both point to changes in a system [20]. At the same time, they are different because information is potency for (or in some sense, cause of) change, while transformation is the change itself, or in other words, transformation is an operation, while information is what induces this operation.

Second, the Ontological Principle O2 explains *why* information influences society and individuals all the time, as well as why this influence grows with the development of society. Namely, reception of information by individuals and social groups induces transformation. In this sense, information is similar to energy. Moreover, according to the Ontological Principle O2, energy is a kind of information in a broad sense. This well correlates with the Carl Friedrich von Weizsäcker's idea [21,22] that *energy might in the end turn out to be information*.

Third, the Ontological Principle O2 makes it possible to separate different kinds of information. For instance, people, as well as any computer, have many kinds of memory. It is even supposed that each part of the brain has several types of memory agencies that work in somewhat different ways, to suit particular purposes [23]. It is possible to consider each of these memory agencies as a separate system and to study differences between information that changes each type of memory. This might help to understand the interplay between stability and flexibility of mind, in general, and memory, in particular.

In essence, we can see that all kinds and types of information are encompassed by the Ontological Principle O2.

However, the common usage of the word *information* does not imply such wide generalizations as the Ontological Principle O2 implies. Thus, we need a more restricted theoretical meaning because an adequate theory, whether of the information or of anything else, must be in significant accord with our common ways of thinking and talking about what the theory is about, else there is the danger that theory is not about what it purports to be about. To achieve this goal, we use the concept of an *infological system* IF($R$) of the system $R$ for the information definition. It is done in two steps. At first, we make the concept of information relative and then we choose a specific class of infological systems to specify information in the strict sense. That is why it is impossible and, as well as, counterproductive to give an exact and thus, too rigid and restricted definition of an infological system. Indeed, information is a very rich and widespread phenomenon to be reflected by a restricted rigid definition (*cf.*, for example, [24,25]).

The concept of infological system plays the role of a free parameter in the general theory of information, providing for representation of different kinds and types of information in this theory. That is why the concept of *infological system*, in general, should not be limited by boundaries of exact definitions. A free parameter must really be free. Identifying an infological system IF($R$) of a system $R$,

we can define information relative to this system. This definition is expressed in the following principle.

**Ontological Principle O2g** (**the *Relativized Transformation Principle***). *Information* for a system *R relative to the infological system* IF(*R*) is a capacity to cause changes in the system IF(*R*).

As a model example of an infological system IF(*R*) of an intelligent system *R*, we take the system of knowledge of *R*. In cybernetics, it is called the *thesaurus* Th(*R*) of the system *R*. Another example of an infological system is the memory of a computer. Such a memory is a place in which data and programs are stored and is a complex system of diverse components and processes.

Elements from IF(*R*) are called *infological elements*.

There is no exact definition of infological elements although there are various entities that are naturally considered as infological elements as they allow one to build theories of information that inherit conventional meanings of the word *information*. For instance, knowledge, data, images, algorithms, procedures, scenarios, ideas, values, goals, ideals, fantasies, abstractions, beliefs, and similar objects are standard examples of infological elements.

When we take a physical system *D* as the infological system and allow only for physical changes, we see that information with respect to *D* coincides with (physical) energy.

Taking a mental system *B* as the infological system and considering only mental changes, information with respect to *B* coincides with mental energy.

These ideas are crystallized in the following principle.

**Ontological Principle O2a** (**the *Special Transformation Principle***). *Information in the strict sense* or *proper information* or, simply, *information* for a system *R*, is a capacity to change structural infological elements from an infological system IF(*R*) of the system *R*.

As the concept of mental energy is much less understood than the concept of physical energy, we give some explanations based on the origin, development and contemporary understanding of mental energy.

Considering mental energy on the level of individual mentality, it is possible, as the first approximation, to equate it with psychic energy. The concept of psychic energy first entered physiology and to some extent psychology through the discussions of Ernst Brücke, Herman Helmholtz, and Emil Du Bois-Reymond, who during the years 1838-1842, worked in the laboratory of the German physiologist Johannes Muller. At the same time, according to Jung [26], Nicolas von Grot was the first to explicitly define the concept of psychic. He wrote [27]:

"The concept of psychic energy is as much justified in science as that of physical energy, and psychic energy has just as many quantitative measurements and different forms as has physical energy."

Later this concept was further developed by Sigmund Freud [28,29], Brücke's student, and then by Carl Jung, Freud's student. Jung [26] regarded psychic energy as a basic life-force, manifesting itself through actions, such as eating, moving, thinking, sex, remembering, etc.

Contemporary understanding determines mental energy as the ability to perform mental tasks, the intensity of feelings of energy/fatigue, and the motivation to accomplish mental and physical tasks [30,31]. This shows that basic aspects of mental energy manifestation include: (a) cognition (knowledge that is gained through perception, reasoning or intuition), (b) changing of moods or feelings (states of mind), and (c) motivation (an incentive for action). Factors that can influence mental

energy include, among others, psychological issues such as interest, passion, desire, concern, and biological issues such as genetics, nutrition, pain, and sleep [31].

Often psychic/mental energy is confused with psychic/mental force. For instance, some psychoanalysts do not distinguish energy and force, particularly, when they follow Freud's occasional practice of calling libido itself a force. Actually, as in physics, energy and force are different phenomena. Energy is a potential to make changes, while force is what is making changes. This allows one to take the Will as a representative of force in the individual mentality and above. This is expressed in the remark attributed to Loewenstein that "force is energy in action" (*cf.* [32]).

## 3. The phenomenological stratum of the general theory of information

In this paper, we are primarily interested in information received, exchanged, processed and used by people. We call it *anthropic information*. A relevant infological system IF(*R*) for anthropic information is the human brain. Thus, to further classify and study anthropic information, we need to utilize our knowledge about the structure and functions of the brain.

In our case, the most relevant is the Triune Brain model introduced and studied by Paul MacLean [33,34]. The main conception of his approach is existence of three levels of perception and action that are controlled by three corresponding centers of perception in the human brain. These three centers together form the Triune Brain and have the structure of a triad. It is natural to call the initial MacLean's structure by the name Anatomic Triune Brain model because it is based on the anatomy of the brain where three indispensable parts are distinguished: the neocortex, limbic system and R-complex.

According to the theory of MacLean, the neural basis, or framework, of the brain consists of three parts: the spinal cord, hindbrain, and midbrain. In addition to it, centuries of evolution have endowed people with three distinct cerebral systems. The oldest of these is called the *reptilian brain* or R-*complex*. It programs behavior that is primarily related to instinctual actions based on ancestral learning and memories, satisfying basic needs such as self-defense, reproduction and digestion. The reptilian brain is fundamental in acts such as primary motor functions, primitive sensations, dominance, establishing territory, hunting, breeding, and mating.

Through evolution, people have developed a second cerebral system, the *limbic system*, which MacLean refers to as the *paleomammalian brain* and which contains hippocampus, amygdala, hypothalamus, pituitary gland, and thalamus. This system is situated around the R-complex, is shared by humans with other mammals, and plays an important role in human emotional behavior.

The most recent addition to the cerebral hierarchy is called the *neomammalian brain*, or the *neocortex*. It constitutes 85% of the whole human brain mass and receives its information from the external environment through the eyes, ears, and other organs of senses. This brain component (neocortex) contains cerebrum, corpus callosum, and cerebral cortex. The cerebrum and cerebral cortex are divided into two hemispheres, while the corpus callosum connects these hemispheres. The neocortex deals with information in a logical and algorithmic way. It governs people creative and intellectual functions like social interaction and advance planning. The left hemisphere works with symbolic information, applying step-by-step reasoning, while the right hemisphere handles images processed by massively parallel (gestalt) algorithms.

Even psychologists who have objections to the Anatomic Triune Brain model admit that it is a useful, although oversimplified, metaphor, as the structure presented as the triune brain is based on a sound idea of three functional subsystems of the brain [16,35]. In the development of neurophysiology and neuropsychology, MacLean's theory was used as a base for the Whole Brain model, developed by Herrmann [36]. The main idea of this development is a synthesis of the Anatomic Triune Brain model with the two-hemisphere approach to the brain functioning.

The theory of the triune brain (reptilian, old mammalian and new mammalian) is used as a metaphor and a model of the interplay between instinct, emotion, and rationality in humans. Cory [37] applied this model to economic and political structures. In Cory's schema, the reptilian brain mediates the claims of self-interest, whereas the old mammalian brain mediates the claims of empathy. If selfish interests of an individual are denied for too long, there is discontent due to a feeling of being unjustly treated. If empathic interests are denied for too long, there is discontent due to guilt. In either case, the center of intelligence at the prefrontal cortex plays the role of a mediator. Its executive function is required to restore balance, generating the reciprocity required for effective social and economic structures.

The Triune Brain model is used to explain hyperactivity of youngsters studied by Zametkin [38] and other researchers. Peter Levine bases his approach to trauma treatment on the Triune Brain model [39]. According to Levine, there are tree types of the uniform stress and relaxation responses to a threatening situation that are active in all animal species through the autonomic nervous system. In the everyday language, these responses are metaphorically called *fight*, *flight* or *freeze*. The first two of them are well-known, while the third one was introduced by Levine. The freezing or immobility response has evolved over millions of years and it has served an adaptive purpose well for all species—except humans. In an individual, it can lead to trauma. Many physical ailments are actually residues of thwarted trauma reactions incurred during stressful events. What usually happens to non-human species is that after the threatening situation resolves itself, the animal forgets the stress and goes on its way without being traumatized.

In contrast to this, people can get stuck in the freezing response, while the reasoning mind resists or blocks the natural bodily sensations and fine motor movements needed to come out of the freeze response. The contemporary rationalistic culture is not helpful in supporting people in such a traumatizing situation. The feelings that people go through after experiencing a traumatic event are outside of their voluntary control, often being frightening and even potentially re-traumatizing.

Levine [39] postulates that trauma exists not in the event or in the story of the event, but is stored within the nervous system. The main principle of the Levine's treatment approach is that the body has a natural, innate, and miraculous capacity to heal once these reactions are understood and guided.

Although the Triune Brain has become a well-known model in contemporary psychology, it caused several objections on the ground of the development and structure of the triune brain system [16,40]. First, there is evidence that the, so-called, paleomammalian and neomammalian brains appeared, although in an undeveloped form, on much earlier stages of evolution than it is assumed by MacLean. Second, there are experimental data that in the neocortex, regions that are homological to the, so-called, paleomammalian and reptilian brains exist. For instance, neuropsyhological data give evidence that amygdala, which is a part of a limbic system, performs the low-level emotion processing, while the ventromedial cortex performs the high-level emotion processing. This shows that emotions

exist, at least, on three levels: on the subconscious level of limbic system, on the conscious intuitive level, and on the conscious rational level in the cortex. The first level utilizes direct affective information, while the second and, to some extent, the third levels make use of cognitive emotional information [1].

At the same time, the development of the system of Will demands inclusion of some regions that are not included into the R-complex (the reptilian brain in MacLean's theory) into this system. It means that the centers of rational intelligence, emotion and will are not concentrated in three separate regions of the brain but are highly distributed among several components of the brain. Thus, it is better to call them not centers but systems of intelligence, emotion and will. This extension of functional characteristics results in the Componential Triune Brain model described in [1], making this model more adequate to experimental data and overcoming the pitfalls of the simple linear hierarchy suggested by MacLean [16,40].

Thus, the Componential Triune Brain model consists of three basic systems of the brain:

- the *System of Rational Intelligence* (also called *System of Reasoning*) (SRI);
- the *System of Emotions* (or more generally, of *Affective States*) (SAS);
- the *System of Will and Instinct* (SWI).

All three systems of the brain are schemes in the sense of the schema theory, which is developed as a specific direction of the brain theory [41-44]. According to this theory, brain schemes are anatomically distributed in the brain and interact in a way of concurrent competition and coordination. All these interactions are based on physical processes but have an inherent informational essence related to a specific type of information. Information processes in the brain are more exactly reflected by the theory of the *triadic mental information* than by the conventional information theory that deals only with cognitive information.

In standard structuring of the brain, we also find these three systems. In the conventional setting (*cf.*, for example, [40,45-47] , the brain includes three components: the forebrain, midbrain, and hindbrain.

The *forebrain* is the largest division of the brain involved in a wide range of activities that make people human. The forebrain has a developed inner structure. It includes the *cerebrum*, which consists of two *cerebral hemispheres*. The cerebrum is the nucleus of the System (Center) of Rational Intelligence.

Under the cerebrum, is the *diencephalon*, which contains the *thalamus* and *hypothalamus*. The thalamus is the main relay center between the *medulla* and the *cerebrum*. The hypothalamus is an important control center for sex drive, pleasure, pain, hunger, thirst, blood pressure, body temperature, and other visceral functions. The forebrain also contains the limbic system, which is directly linked to the experience of emotion. The *limbic system* is the nucleus of the System (Center) of Emotions (or more generally, of Affective States).

The *midbrain* is the smallest division and it makes connections with the other two divisions—forebrain and hindbrain—and alerts the forebrain to incoming sensations.

The *hindbrain* is involved in sleeping, waking, body movements and the control of vital reflexes such as heart rate, blood pressure. The structures of the hindbrain include the pons, medulla and cerebellum. The hindbrain is the nucleus of the System (Center) of Will and Instinct.

The System of Rational Intelligence realizes rational thinking. It includes both symbol and image processing, which go on in different hemispheres of the brain [48]. The System of Emotions governs sensibility and the emotional sphere of personality. The System of Will and Instincts directs behavior and thinking. Two other systems influence behavior only through the will. For instance, a person can know that it is necessary to help others, especially, those who are in need and deserve helping. However, in many cases, this person does nothing without a will to help. In a similar way, we know situations when an individual loves somebody but neither tells this nor explicitly shows this due to an absence of a sufficient will.

It is necessary to remark that discussing the will of an individual we distinguish *conscious will*, *unconscious will*, and *instinct*. All of them are controlled by the Center of Will and Instincts (SWI). In addition, it is necessary to make distinctions between thoughts about intentions to do something and the actual will to do this. Thoughts are generated in the Center of Rational Intelligence, while the will dwells in the Center of Will and Instincts. In other words, thoughts and words about wills, wishes, and intentions may be deceptive if they are not based on a Will.

Will is a direct internal injunction, as well as any kind of motivation [19]. That is, the forces that act on or within an organism to initiate and direct behavior, has to be transformed into a will in order to cause the corresponding action. Will is one of the basic components of different models of personality. For instance, in psychosynthesis, which is a holistic transpersonal psychology and philosophy of life developed by Roberto Assagioli, a contemporary of both Freud and Jung, the Will is the basic component of the self [18].

Assagioli explicated essential properties of the Will—it can be assertive, aggressive, and controlling. In addition, there are three categories of Will: the *accepting Will*, *yielding Will*, and *dedicated Will*.

Thus, we can see that the Componential Triune Brain model corresponds to the three basic functions of the brain:

- Reasoning as symbolic information processing and information processing of images.
- Emotions and feelings as affective states of the brain.
- Will and instinct as forces of the psyche.

The Anatomic Triune Brain model of MacLean also corresponds to the basic functions of the brain but misses one of them, namely, the Will.

Many other psychological theories and psychiatric tecniques consider the Will as a primary factor of human behavior and dispositions [19,49].

Often the Will is considered as a process that deliberates on what is to be done [50].

The Componential Triune Brain model is not only a necessary extension of the Triune Brain model but it also continues a long standing approach to the brain stratification. As Smith [35] demonstrates, triadic models of the brain and psyche have featured through two and half millennia of Western thought, starting with works of Pythagoras, Plato and Aristotle and receiving a modern airing in Paul MacLean's the Triune Brain model. A generation later after Pythagoras, Plato and Aristotle, Herophilus and Erasistratus from Alexandrian put together a more anatomically informed triadic theory, which was modified by Galen in the 2nd century and remained the prevailing paradigm for nearly fifteen hundred years until it was overturned by the great thinkers of the Renaissance. Nonetheless, the notion that the human neuropsychological system is somehow best thought of as

having a triadic (tripartite) structure has remained remarkably resilient and has reappeared time and again in modern and early modern times. For instance, the Triune Brain model well correlates with the Freud's model of personality, which has the structure of the triad (*Id*, *Ego*, *Super-ego*). In the correspondence between the Triune Brain model and the Freud's model, the reptilian complex corresponds to Id, the limbic system corresponds to Ego, and the neocortical complex corresponds to Super-ego. In the context of triadic models, it is also possible to consider the *triarchic theory of intelligence* developed by Robert Sternberg [51].

Taking each of these the centers, SRI, SAS, and SWI, as a specific infological system, we find three types of information. One is the conventional information that acts on the center of reasoning and of other higher brain functions (SRI), which is situated in the neocortex. This information gives knowledge, changes beliefs and generates ideas. Thus, it is natural to call it *cognitive* information. Information of the second type acts on the system of emotions (SAS), which includes the paleomammalian brain. It is natural to call this information by the name *direct emotional information*, or *direct affective information* or *emotive information*. Information of the third type acts on the System of Will and Instinct (SWI), which contains the reptilian brain. It is natural to call this information by the name *direct regulative* or *direct effective information*.

Thus, anthropic information has three dimensions:

*Cognitive information* changes the content of the SRI, which includes the knowledge system (thesaurus) and neocortex (neomammalian brain) as its carrier.

*Direct emotional/affective information* changes the content of the SAS, which includes the paleomammalian brain (limbic system).

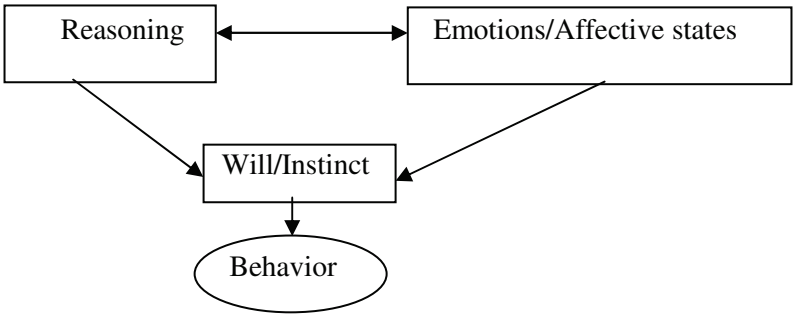
*Direct regulative/effective information* changes the content of the SWI, which includes the reptilian brain or R-complex.

However, in general, emotions constitute only one part of affective states, which also include moods, feelings, etc. That is why in general, direct affective information is more general than direct emotional information. However, as there is no consensus on the differences between emotions and affective states, these two types of information are used without differentiation.

Interactions between the basic brain systems imply dependencies between thinking, emotions, and actions of people. Emphasizing some of these relations, psychologists build their theories and psychotherapists develop their therapeutic approaches. Giving priority to the System of Rational Intelligence (SRI), the so-called "cognitive revolution" has taken hold around the world. It influenced both psychology, resulting in the emergence of cognitive psychology [52], and psychotherapy, inspiring creation of cognitive therapy [53]. In psychology, the word *cognitive* often means thinking in many contexts of contemporary life [54]. The cognitive therapeutic approach begins by using the extremely powerful reasoning abilities of the human brain. This is important because our emotions and our actions are not separate from our thoughts. They are all interrelated. Thinking (SRI) is the gateway to our emotions (SAS)—and our emotions are the gateway to our actions through motivation and will (SWI). This is only another way of saying that information from the System of Rational Intelligence (SRI) goes to the System of Emotions (SAS)—and from it to the System of Will and Instincts (SWI) that controls our actions. Consequently, the cognitive psychotherapeutic approach, which has been successfully utilized for treating many mental disorders, gives additional supportive evidence for the theory of the triune brain and behavior, as well as for the theory of the triadic mental information. The

latter explains that while going from the System of Rational Intelligence to the System of Emotions and to the System of Will and Instincts, information is transformed from cognitive information, to direct emotional/affective information to direct effective/regulative information. As a result, we have Interaction of the personality components presented in Figure 1.

**Figure 1.** Interaction between components of personality.




We can see that cognitive information is one of the three basic types of anthropic information with the corresponding infological system CIF(*R*), which contains (stores and processes) cognitive elements or constituents, such as knowledge, data, ideas, beliefs, images, algorithms, procedures, scenarios, ideas, values, goals, ideals, fantasies, abstractions, etc. Cognitive infological systems are very important, especially, for intelligent systems as the majority of researchers believe that information is intrinsically connected to cognition. This peculiarity is reflected in the following Cognitive Transformation Principle.

**Ontological Principle O2c** (**the *Cognitive Transformation Principle***). *Cognitive information* for a system *R*, is a capacity to cause changes in the cognitive infological system CIF(*R*) of the system *R*.

In the case of a cognitive infological system CIF(*R*), it looks like it is possible to give an exact definition of cognitive information. However, now cognitive sciences do not know all structural elements involved in cognition. A straightforward definition specifies cognition as activity (process) that gives knowledge. At the same time, we know that knowledge, as a rule, comes through data and with data. So, data are also involved in cognition and thus, have to be included in cognitive infological systems. In addition, cognitive processes employ such structures as ideas, images, texts, beliefs, values, measures, problems, schemas, procedures, tasks, goals, etc. Thus, to comprehensively represent cognitive information, it is imperative to include all such objects in cognitive infological systems.

As the cognitive infological system contains knowledge of the system it belongs, cognitive information is the source of knowledge changes.

There are also different types of cognitive information. One approach to classification is based on the structures in the brain. Researchers found that a longitudinal fissure separates the human brain into two distinct cerebral hemispheres, connected by the corpus callosum. The sides resemble each other and each hemisphere's structure is generally mirrored by the other side. Yet despite the strong similarities, the functions of each cortical hemisphere are different. As it is known (*cf.*, for example, [55]), the left hemisphere operates with symbolic information representation, performing logical and analytical functions, such as linear reasoning, numeric manipulation, language processing, and mental arithmetic. It is also supposed that the left hemisphere works in the sequential mode on the level of

consciousness. The right hemisphere processes images and realizes intuitive, creative and synthesizing functions of the brain, such as processing of visual and audiological stimuli, spatial manipulation, facial perception, and artistic performance. It is also supposed that the right hemisphere works in the parallel (concurrent) mode on the level of consciousness. In a similar way, Herrmann [36] differentiates functioning of the left parts and right parts of both the cerebral and limbic regions of the brain.

Taking each of these hemispheres as an infological system, we come to two types of cognitive information.

*Symbolic* (*discrete*) cognitive information transforms (or can transform) the symbolic content of the left hemisphere.

*Holistic* (*continuous*) cognitive information transforms (or can transform) the integral (gestalt) content of the right hemisphere.

This classification of cognitive information is complementary to another classification, in which one of the basic types of cognitive information is epistemic information, which can be both symbolic and holistic, belonging to both, the right hemisphere and left hemisphere. It is studied in Section 5.

## 4. Stratification of Knowledge Systems

There are different kinds of stratification.

In *physical stratification*, each stratum is a separate physical system. Any distributed database is physically stratified.

In *analytical stratification*, each stratum is determined by a specific name (label) and all elements from this stratum have this label (name). Knowledge base stratification used for handling inconsistent knowledge bases [56,57], for constructing models of a knowledge base [58] and for merging multiple knowledge bases [59,60] is analytical. The same is logic stratification used for formalization of commonsense reasoning [61]. In this section, we are mostly interested in analytical stratification based on knowledge classification. Different classes of knowledge form corresponding strata of knowledge systems.

There are different principles of knowledge classification, which allow us to build several types of knowledge system stratifications.

Time is an important characteristic of knowledge, giving different stratifications.

The *temporal stratification*.

1. The *past stratum* of knowledge consists of knowledge obtained/accepted in the past.
2. The *current stratum* of knowledge consists of the actual (used now) knowledge.
3. The *future stratum* of knowledge.

For instance, the knowledge “the Earth is flat” is past knowledge, while the knowledge “the Earth is round” is current knowledge.

The past stratum of knowledge consists of three substrata: the *forgotten past knowledge*, *outdated but preserved past knowledge* and *still actual past knowledge*.

The current stratum of knowledge consists of three substrata: the *disappearing current knowledge*, *consolidated current knowledge*, and *emergent current knowledge*.

The future stratum of knowledge consists of three substrata: the *tentative/potential future knowledge*, *realizable future knowledge* and *emergent future knowledge*.

More precise temporal stratifications are used in temporal the knowledge and databases. A temporal knowledge/database is a database with built-in time aspects. In particular, it supports a temporal knowledge/data model and has a temporal version of the query language [62,63]. Temporal knowledge/data stored in a temporal knowledge/database are different from the knowledge/data stored in non-temporal knowledge/database in that a time coordinate is attached to the knowledge/data. This is different from the conventional knowledge/data, which are usually considered to be valid now. Past and future knowledge/data are not stored. Usually past knowledge/data are modified, overwritten with new (updated) knowledge/data or deleted to achieve their temporal relevancy. Future knowledge/data are not considered because it is assumed that we do not receive information about the future.

There are many complexity measures of algorithms, methods and procedures. Taking a complexity measure *C*, it is possible to partition all algorithms (methods or/and procedures) into separate classes that have different complexity measures. Each such a partition induces a corresponding stratification of knowledge with respect to such knowledge characteristics as accessibility, inference, and generation, which are specific forms of knowledge acquisition. Here are some examples of such stratifications.

The *accessibility stratification*.

1. *Directly accessible* knowledge.
2. *n-step accessible* knowledge.

Another stratification is based on complexity of knowledge inference.

The *inference stratification*.

1. *Directly implied* knowledge.
2. *n-step inferable* knowledge.

One more stratification is based on complexity of knowledge generation.

The *generation stratification*.

1. *Directly generable/computable* knowledge.
2. *n-step generable/computable* knowledge.

Steps in generation, inference and access may be determined by:

- *Time slicing* when each step is assigned some period of time for realization.
- *Elementary operations*.

For instance, it is possible to assume that knowledge acquisition is direct if it demands less than 3 seconds. The first step of knowledge acquisition can be estimated as an interval from 3 seconds to 30 seconds. The second step of knowledge acquisition can be estimated as an interval from 30 seconds to 1 minute. The third step of knowledge acquisition can be estimated as an interval from 1 minute to 3 minutes and so on.

It is also possible to measure complexity, e.g., effort in generation, by the power of algorithms [64]. In this case, we have an *algorithmic ladder*, which consists of classes of algorithms with increasing computing power.

The traditional algorithmic ladders have one of the following forms:

(1) Finite automata, deterministic pushdown automata, pushdown automata, and Turing machines.

(2) Regular, or linear grammars, context-free grammars, context-sensitive grammars, and unrestricted, or phrase-structure grammars.

New achievements of the theory of algorithms and computation extend these ladders:

(1') Finite automata, deterministic pushdown automata, pushdown automata, Turing machines, inductive Turing machines [65], and infinite-time Turing machines [66].

(2') Regular, or linear grammars, context-free grammars, context-sensitive grammars, unrestricted, or phrase-structure grammars, grammars with prohibition [67], and Boolean grammars [68].

Inductive Turing machines give an example of an algorithmic ladder. Namely, the $n$-th strata of the inductive algorithmic ladder consists of inductive Turing machines with the structured memory that have order $n$ but do not have order $n + 1$ [65]. It is also possible to build an algorithmic ladder using inductive or limit) Turing machine have structured program (rules for computation) or structured (heads) operating devices [65].

## 5. Stratified M-spaces and Information Operators

Let us consider a universal set or multiset $\boldsymbol{W}$ of knowledge items (units). It is possible to take the set $\boldsymbol{W}_C$ of elementary knowledge units mathematically modeled in [1,69] as a universal set (multiset) $\boldsymbol{W}$, obtaining a reasonable formalization of the concept of a knowledge state. Another possibility for $\boldsymbol{W}$ is realized by the set (multiset) $\boldsymbol{W}_L$ of propositions and/or predicates from some logical language $L$. This logical approach was adopted in works of Bar-Hillel and Carnap [2], Hintikka [3-5] and some other authors. Shreider [7] interpreted knowledge items as texts in a thesaurus. Many researchers employ schemas as knowledge items in the brain (*cf.*, for example, [41-44]. One more possibility for $\boldsymbol{W}$ is the set, or more exactly, a multiset, $\boldsymbol{W}_S$ of situations possible in a world $U$, which are taken as knowledge items or knowledge units (*cf.*, for example, [70,71].

The set $\boldsymbol{W}$ is called *universal* because we assume that the following axiom is true.

**MA1 (the Internal Representation Axiom).** For any cognitive system (agent) $A$, knowledge states $K_{Ai}$ of $A$ are subsets (submultisets) of the set (multiset) $\boldsymbol{W}$.

It is possible to interpret $\boldsymbol{W}$ as the base of all knowledge that agents are able to have about their environment.

Another aspect of universality of the set (multiset) $\boldsymbol{W}$ may be in the possibility to describe all possible (existing) worlds utilizing knowledge only from $\boldsymbol{W}$. For instance, when $\boldsymbol{W}$ is the set (multiset) $\boldsymbol{W}_L$ of propositions and/or predicates from some logical language $L$, then it is possible to build all descriptions of all possible worlds by combining elements from $\boldsymbol{W}_L$. This possibility is reflected in the following axiom.

**MA2 (the External Representation Axiom).** For any environment (situation) $D$, there is a subset (submultiset) $\boldsymbol{W}_D$ of the set (multiset) $\boldsymbol{W}$ that contains all accessible knowledge about $D$.

Taking these two axioms as the foundation, we develop a theory of cognitive systems/agents called the *theory of M-spaces*.

**Definition 5.1** [1]**.** a) Subsets of $\boldsymbol{W}$ are called *Mizzaro spaces*.

b) Submultisets of $\boldsymbol{W}$ are called *Mizzaro multispaces*.

In some cases, only specific subsets (submultisets) of $\boldsymbol{W}$ are used in the theory. For instance, if elements of $\boldsymbol{W}$ are propositions and the model satisfies conditions of classical logical calculi, then only consistent subsets of propositions are acceptable as Mizzaro spaces. When, in addition, all deducible propositions are also included in such a logical Mizzaro space, then Mizzaro spaces are components of a logical variety [72].

Taking a knowledge system $\boldsymbol{K}$, we model the states of $\boldsymbol{K}$ by Mizzaro spaces (Mizzaro multispaces), *i.e.*, each knowledge state is represented by a Mizzaro space (Mizzaro multispace) $K_{\boldsymbol{K}i}$ . The whole knowledge system $\boldsymbol{K}$ is modeled by an M-space.

In modeling knowledge systems and information processes, we consider two structures – sets and multisets – because using the classical background it is possible to consider only sets, which make the model simpler. However, many real cognitive systems contain several copies of the same element. For instance, the same element of knowledge can be stored in different parts of a computer memory or of the brain. This makes utilization of multisets necessary.

**Definition 5.2.** A *type of structures* is a system of conditions (axioms) that all these structures, *i.e.*, sets with relations, satisfy.

To define an M-space, we consider a type θ of structures in Mizzaro spaces (Mizzaro multispaces).

**Definition 5.3.** An M-*space* (M-*multispace*) $\boldsymbol{M}$ is a structure of the form

$$\boldsymbol{M} = \{\mathrm{KS}_{\boldsymbol{M}} ; \mathrm{OS}_{\boldsymbol{M}} \}$$

where KS consists of Mizzaro spaces (Mizzaro multispaces) $K$ with the structure of the type θ, and $\mathrm{KS}_{\boldsymbol{M}}$ is a system of information operators in $\mathrm{OS}_{\boldsymbol{M}}$ acting on Mizzaro spaces (Mizzaro multispaces).

Thus, $\mathrm{KS}_{\boldsymbol{M}} = \{K_i \mid i \in I\}$ and $\mathrm{OS}_{\boldsymbol{M}} = \{A_t \mid t \in T\}$.

**Example 5.1.** It is possible to represent a logical variety or a prevariety $\boldsymbol{V}$ [72] as an M-space where $\mathrm{KS}_{\boldsymbol{M}}$ consists of the components of $\boldsymbol{V}$ and operators from $\mathrm{OS}_{\boldsymbol{M}}$ apply mappings $f_i$: $A_i \to L$ and $g_i : T_i \to L$ ($i \in I$), which form connections between components of the variety (prevariety).

**Definition 5.4.** a) The set $\mathrm{KS}_{\boldsymbol{M}}$ is called the *state space* of the M-space $\boldsymbol{M}$.

b) The set $U_{\boldsymbol{M}} = \bigcup_{\substack{i \in I \\ K_i \in KS_{\boldsymbol{M}}}} K_i$ is called the *universe* of the M-space $\boldsymbol{M}$.

c) The system $\mathrm{OS}_{\boldsymbol{M}}$ is called the *operating system* of the M-space $\boldsymbol{M}$.

The simplest structure of Mizzaro spaces $K_i$ of the type θ is the structure of a set and the simplest structure of Mizzaro multispaces $Ki$ of the type θ is the structure of a multiset. However, Mizzaro spaces $K_i$ can be logical calculi, linear spaces or groups.

In the algebraic context, each M-space $\boldsymbol{M}$ is a universal algebra [73] with the support $\mathrm{KS}_{\boldsymbol{M}}$ and system of operations $\mathrm{OS}_{\boldsymbol{M}}$. In this paper, we consider only *unary Mizzaro spaces* (*unary Mizzaro multispaces*) in which each information operator maps one Mizzaro space (Mizzaro multispace) $K_i$ into another (may be the same) Mizzaro space (Mizzaro multispace) $K_j$ .

Information operators from $\mathrm{OS}_{\boldsymbol{M}}$ are global epistemic information operators in $\mathrm{KS}_{\boldsymbol{M}}$ . When an operator acts only on one Mizzaro space (Mizzaro multispace), then it is a local epistemic information operator. Local epistemic information operators in non-structured Mizzaro spaces are studied in [1,10,11,13].

Note that it is possible to consider any system that contains a knowledge system as a knowledge system itself. Thus, any intelligent agent is a knowledge system.

There are two basic types of epistemic information operators: content, bond and mixed operators.

**Definition 5.5.** A *content epistemic information operator* acts on knowledge items in a knowledge state.

For instance, all information operators studied in [1,10,11,13] are content epistemic information operators.

**Definition 5.6.** A *bond epistemic information operator* acts on connections (bonds or relations) between knowledge items in a knowledge state.

Such operators as interpretation and reinterpretation of information/knowledge items are bond epistemic information operators.

**Definition 5.7.** A *mixed epistemic information operator* acts both on knowledge items and on connections (bonds or relations) between knowledge items in a knowledge state.

Operators of logical inference, such as rules of deduction, are mixed epistemic information operators act because they add new knowledge items in the form of propositions or/and predicates and establish relations of inferrability/deducibility between propositions or/and predicates.

Here we are mostly interested in content epistemic information operators, which we simply call epistemic information operators.

To correctly model stratified knowledge system, the modeling M-space also has to be stratified.

**Definition 5.8.** a) An M-space $\boldsymbol{M} = \{\mathrm{KS}_{\boldsymbol{M}}; \mathrm{OS}_{\boldsymbol{M}}\}$ is *stratified* if there is a set $J$ and each Mizzaro space (Mizzaro multispace) $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$ has the form

$$K_i = \bigcup_{j \in J} K_{ij}$$

b) A stratification of an M-space $M = \{ \mathrm{KS}_{\boldsymbol{M}}; \mathrm{OS}_{\boldsymbol{M}} \}$ is *strict* if for each Mizzaro space (Mizzaro multispace) $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$, $K_{ij} \cap K_{ik} = \varnothing$ when $j \neq k \in J$.

c) An M-space $\boldsymbol{M} = \{\mathrm{KS}_{\boldsymbol{M}}; \mathrm{OS}_{\boldsymbol{M}}\}$ is *linearly stratified* if each Mizzaro space (Mizzaro multispace) $K_i$ from MK has the form

$$K_i = \bigcup_{n=1}^{\infty} K_{in}$$

in the case when the stratification is infinite and the form

$$K_i = \bigcup_{n=1}^{m} K_{in}$$

in the case when the stratification is finite (*cf.* Figure 2).

Linear stratification means that the set of stratum indices $J$ is finite or countable and linearly ordered.

**Figure 2.** A finite M-space stratification with the linear topology.

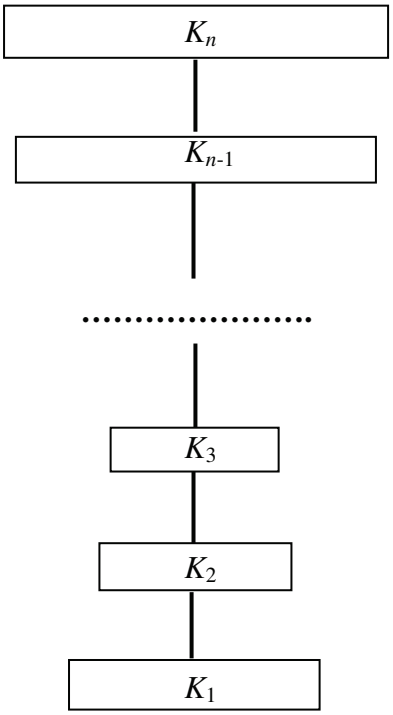
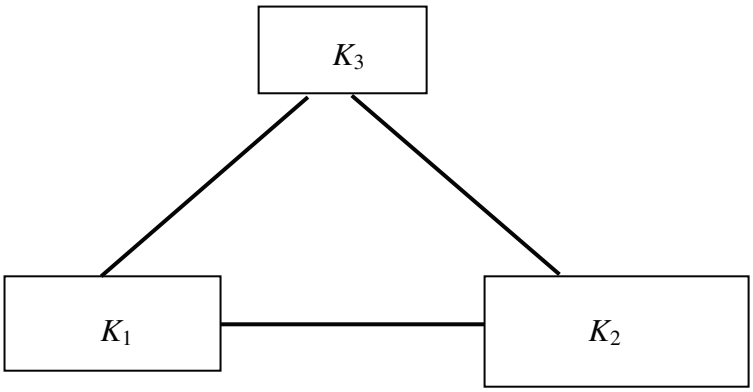



**Figure 3.** A finite M-space stratification with the cyclic topology.




There are M-space stratifications with a non-linear topology. For instance, the stratification in Figure 3 has a cyclic topology. Important cases of M-space stratifications have structures of a tree or a forest.

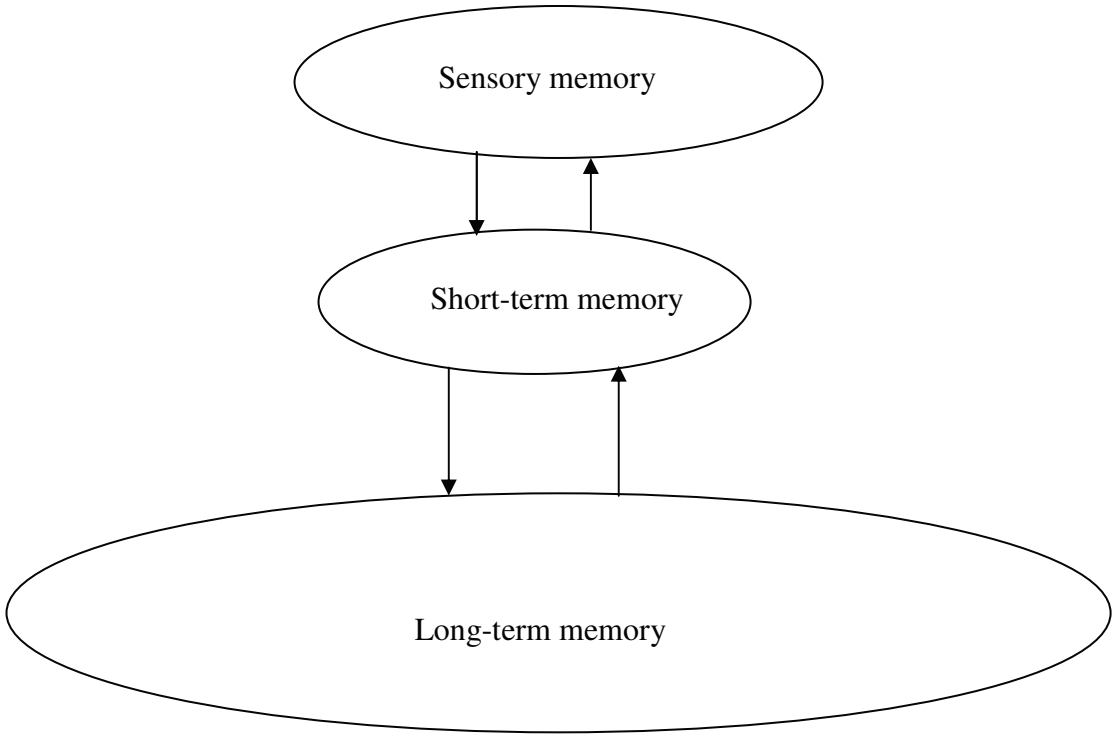
**Example 5.2.** People, as well as computers, have many kinds of memory. It is even supposed that each part of the brain has several types of memory agencies that work in somewhat different ways, to suit particular purposes [23]. It is possible to consider each of these memory agencies as a separate system and to study differences between information that changes each type of memory. This might

help to understand the interplay between stability and flexibility of the mind, in general, and memory, in particular.

Psychologists differentiate three types of human memory: *sensory memory*, *short-term memory*, and *long-term memory*. It is the most important and best documented by scientific research memory stratification. However, memory researchers do not employ uniform terminology. Sensory memory is also known as *sensory register*, *sensory store*, *sensory information storage*, *eidetic memory* and *echoic memory*. Short- and long-term memories are also referred to as *primary memory* and *secondary memory*, correspondingly. Each component of memory differs with respect to its function, the form of information held, the length of time information is retained, and the amount of information-handling capacity.

Thus, human memory has three basic strata. As a result, all knowledge in the memory of an individual is also stratified into three components: knowledge/data in the sensory memory, knowledge in the short-term memory, and knowledge in the long-term memory of this individual (*cf.* Figure 4). Additional stratification of human memory as a knowledge space induces additional stratification of knowledge.

**Figure 4.** The human memory hierarchy.




Sensory memory acts as a buffer for stimuli received through the senses, which are then filtered and passed from sensory memory into short-term memory by attention.

In tern, sensory memory is also stratified by different sensory channels. There are *iconic memory* for visual stimuli, *echoic memory* for aural stimuli and *haptic memory* for touch.

Long-term memory is naturally stratified. The most popular stratification divides it into two parts: *episodic memory* and *semantic memory*. Episodic memory stores knowledge of events and experiences in a serial form. In contrast to this, semantic memory is a structured record of facts, concepts and skills

that people have acquired. The information in semantic memory is derived from that in the episodic memory of the same individual.

Neuroscientists distinguish three main activities related to long term memory: *storage*, *deletion* and *retrieval*. These operations are modeled by epistemic information operators. Storage is modeled by the epistemic information operator REPL. In information storage, information from sensory memory, at first goes to short-term memory and then is stored in long-term memory, usually by the process called *rehearsal*. Rehearsal is the repeated exposure to a stimulus of a knowledge/data portion, which transfers it into long-term memory. Deletion of a knowledge/data portion is modeled by the epistemic information operator DEL and is mainly caused by *decay* and *interference*. Emotional factors essentially affect the long-term memory functioning. According to contemporary there are two types of information retrieval: recall and recognition. In knowledge/data *recall*, the information is reproduced from memory. Recall is modeled by the epistemic information operator COPY. Knowledge/data *recognition* is based on information that this knowledge/data portion has been seen before and is assisted by the provision of retrieval cues to enable better access in the long-term memory. Recognition is modeled by the epistemic information operator GEN.

Scientists also use another stratification of the human memory: *personal memory*, *semantic memory*, *perceptual memory*, and *skill memory*, which includes, *motor skill memory*, *cognitive* skill memory, and *rote linguistic skill memory*.

**Example 5.3.** The computer memory is also a complex system of diverse components and processes. Memory of a computer includes such three basic components as the random access memory (RAM), read-only memory (ROM), and secondary storage. While RAM forgets everything whenever the computer is turned off and ROM cannot learn anything new, secondary storage devices allow the computer to record information for as long period of time as we want and change it whenever we want. Now the following devices are utilized for log-term computer memory: magnetic tapes and corresponding drives, magnetic disks and corresponding drives, flash memory storage devices and corresponding drives, and optical disks and corresponding drives.

Computer memory is intrinsically stratified by the hierarchy in which levels are distinguished by the response time, complexity, and capacity. The overall goal of using a memory stratification is to obtain the higher possible average access performance, while minimizing the cost of the entire memory system.

Often four major memory levels are separated:

1. *Internal memory* – Processor registers and cache.
2. *Main memory* – the system RAM and controller cards.
3. *On-line mass storage* – Secondary storage.
4. *Off-line bulk storage* – Tertiary and Off-line storage.

At the same time, other experts use another stratification of computer memory:

1. Processor registers have the fastest possible access (in just a few cycles) and usually stores tens of kilobytes
2. Level 2 (L2) cache usually stores 512 KiB or more
3. Level 3 (L3) cache usually stores 2048 KiB or more
4. Main memory, access to which may take hundreds of cycles, but which usually stores multiple gigabytes.

5. Disk storage, access to which may take millions of cycles latency if not cached, but which usually is very large
6. Tertiary storage, access to which may take several seconds latency and which can be huge

Such memory stratifications are linear, reflecting the access time with the fast CPU registers at the top and the slow hard drive at the bottom.

Another, although related, stratification is induced by different electronic devices, which include CPU registers, on-die SRAM caches, external caches, DRAM, paging systems, and virtual memory or swap space on a hard drive. These devices are called RAM (random access memory) by many developers, even though the various subsystems can have very different access times, violating the original concept behind the *random access* term in RAM. RAM consists of two strata: dynamic random access memory, which requires the stored information to be periodically re-read and re-written, or refreshed in order not to lose it, and static memory, never needs to be refreshed as long as power is applied, although it can lose its content if power is removed.

Usually each stratum in RAM is also stratified. For instance, in DRAM, strata are defined by the row, column, bank, rank, and channel.

In addition, computer memory is also stratified by data storage technologies, such as semiconductor, magnetic, and optical technologies. In modern computers, primary storage almost exclusively consists of dynamic volatile *semiconductor memory*, which uses semiconductor-based integrated circuits to store information. Since the turn of the century, a type of non-volatile semiconductor memory known as flash memory has steadily gained share as off-line storage in various advanced electronic devices and computers.

*Magnetic storage*, which is non-volatile, uses different types of magnetization on a magnetically coated surface to store information. The information is accessed using one or more read/write heads which may contain one or more recording transducers. A read/write head only covers a part of the surface so that the head or medium or both must be moved relative to another in order to access data. In modern computers, there are following kinds of magnetic storage devices:

- Magnetic disk, such as sloppy disks, used for off-line storage, and the hard disk drive, used for secondary storage
- Magnetic tape data storage, used for tertiary and off-line storage

At the beginning of computer era, magnetic storage was also used for primary storage in a form of a magnetic drum, or core memory, core rope memory, thin-film memory, twistor memory or bubble memory, while magnetic tapes were often used for secondary storage.

Another popular type of storage is *optical discs*, which stores information in deformities on the surface of a circular disc, reading this information by illuminating the surface with a laser diode and observing the reflection. In modern computers, there are following kinds of optical storage devices:

- CD, CD-ROM, DVD, BD-ROM: Read only storage, used for mass distribution of digital information (music, video, computer programs)
- CD-R, DVD-R, DVD+R, BD-R: Write once storage, used for tertiary and off-line storage
- CD-RW, DVD-RW, DVD+RW, DVD-RAM, BD-RE: Slow write, fast read storage, used for tertiary and off-line storage
- Ultra Density Optical or UDO is similar in capacity to BD-R or BD-RE and is slow write, fast read storage used for tertiary and off-line storage.

In magneto-optical disc storage, the information is read optically and written into the magnetic state on a ferromagnetic surface by combining magnetic and optical methods. It is usually used for tertiary and off-line storage.

*Paper data storage*, typically in the form of paper tape or punched cards, has long been used to store information for automatic processing, particularly before electronic computers were invented.

There are also such memory devices as the vacuum tube memory, electro-acoustic memory, optical tapes, phase-change memory, holographic data storage, and molecular memory, which stores information optically inside crystals or photopolymers.

All these memory devices and components determine the corresponding stratification of knowledge stored in computers, e.g., of knowledge bases.

**Example 5.4.** Stratification is a popular technique in knowledge base theory and practice. For instance, Hunter and Liu [59] introduce knowledge base stratification to solve the problem of merging multiple knowledge bases. Benferhat and Baida [56] use stratified first order logic for access control in knowledge bases. Benferhat and Garcia [57] employ stratification for handling inconsistent knowledge bases. Lassez, *et al*, [58] show how stratification can be used as a tool in the interactive model-building process. Namely it is possible to reduce the computational complexity of the process by the use of stratification, which limits consistency checking to minimal strata.

**Definition 5.9.** a) An M-space $\boldsymbol{M}$ is a *subspace* of an M-space $\boldsymbol{H}$ if the state space $\mathrm{KS}_{\boldsymbol{M}}$ is a substructure of the state space $\mathrm{KS}_{\boldsymbol{H}}$ and the operational system $\mathrm{OS}_{\boldsymbol{M}}$ is a substructure of the operational system $\mathrm{OS}_{\boldsymbol{H}}$.

b) If an M-space $\boldsymbol{M}$ is a subspace of an M-space $\boldsymbol{H}$, then $\boldsymbol{H}$ is called a *superspace* of $\boldsymbol{M}$.

In particular, the stratification of the knowledge space $KS_{\mathbf{M}}$ is induced by the stratification of the knowledge space $KS_{\mathbf{H}}$.

**Example 5.5.** If a structured M-space **H** models the group memory of a group $G$ of several people, then a structured M-space **M** that models the memory of one individual from this group $G$ is a subspace of **H**.

Subspaces of M-spaces and M-multispaces represent subsystems of knowledge systems. For instance, in large knowledge systems, such as a scientific theory, it is possible to separate the subsystem of denotational knowledge and the subsystem of operational knowledge.

**Definition 5.10.** If $X$ is a structure, *i.e.*, a set/multiset with relations, then $\underline{X}$ is the set of all elements from $X$ and $\underline{X}$ is the multiset of all elements from $X$, while Rel$X$ is the set of all relations from $X$.

In such a way, ignoring the M-space stratification, it is possible to represent structured knowledge systems by uniform M-spaces, in which all knowledge states are sets or multisets. In this setting, content epistemic information operators act on elements from sets $\underline{K}$ or multisets $\underline{\mathrm{K}}$, while bond epistemic information operators act on elements from Rel$K$.

**Definition 5.11.** a) A knowledge system (agent) $A$ is called *locally finite* if any knowledge state of $A$ is finite.

b) A knowledge system (agent) $A$ is called *finite* if it has only a finite number of knowledge states and any knowledge state of $A$ is finite.

c) An M-space $\boldsymbol{M}$ is called *locally finite* if each $K$ from $KS_{\boldsymbol{M}}$ contains only a finite number of knowledge items.

d) An M-space $\boldsymbol{M}$ is called *finite* if it has only a finite number of Mizzaro spaces (Mizzaro multispaces) $K$ each of which is also finite.

It looks like it might be sufficient to consider only finite or at least, locally finite agents. However, if knowledge is represented by logical statements and it is assumed (as, for example, in the theory of Bar-Hillel and Carnap [2]) that any knowledge system contains all logical consequences of all its elements, then an agent with such knowledge system is infinite. In information algebras, portions of information are represented by close subsets of sentences from a logical language $L$ [74].

However, in conventional logics, closed with respect to such information operators as deduction sets are infinite because any sentence $p$ implies $p \vee q$ for any sentence $q$ from $L$, which is, as a rule, infinite (*cf.*, for example, [75]). Thus, in the context of classical logic and information algebras any portion of information has infinitely many representations. Consequently, such a portion generates a system with the infinite number of knowledge items.

**Lemma 5.1.** An M-space $\boldsymbol{M}$ is finite if and only if it has a finite universe.

Stratification of the knowledge system and the corresponding M-space allows defining specific classes of epistemic information operators.

**Definition 5.12.** An epistemic information operator $A$ is called *stratified* if for any $j \in J$, there is $k \in J$ such that for any $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$, we have $A(K_{ij}) \subseteq K_{ik}$.

Stratified information operators preserve the structure, *i.e.*, stratification, of knowledge states. Note that adding and deletion operators are intrinsically stratified.

**Definition 5.13.** a) An epistemic information operator $A$ is called *closed* if for any $j \in J$ and for any $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$, we have $A(K_{ij}) \subseteq K_{ij}$.

b) An epistemic information operator $A$ is called *closed* in a Mizzaro space $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$ if $A(K_i) \subseteq K_i$.

**Lemma 5.2.** Any closed epistemic information operator $A$ is stratified.

**Definition 5.14.** a) A stratified epistemic information operator $A$ in a linearly stratified M-space $\boldsymbol{M}$ is called *monotone* (*antitone*) if for any $n \in \boldsymbol{N}$, there is $k \in \boldsymbol{N}$ such that $k \geq n$ ($k \leq n$) and for any $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$, we have $A(K_{ij}) \subseteq K_{ik}$.

b) A stratified epistemic information operator $A$ in a linearly stratified M-space $\boldsymbol{M}$ is called *strictly monotone* (*strictly antitone*) if for any $n \in \boldsymbol{N}$, there is $k \in \boldsymbol{N}$ such that $k > n$ ($k < n$) and for any $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$, we have $A(K_{ij}) \subseteq K_{ik}$.

Definitions imply the following property of epistemic information operators.

**Lemma 5.3.** Any strictly monotone (strictly antitone) epistemic information operator $A$ is monotone (antitone).

**Lemma 5.4.** In a finite linearly stratified M-space $\boldsymbol{M}$, there are no strictly monotone and strictly antitone information operators.

**Definition 5.15.** An epistemic information operator $A$ is called *contracting* if there is $k \in J$ such that for any $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$, we have $A(K_{ij}) \subseteq K_{ik}$.

Definitions imply the following result.

**Lemma 5.5.** Any contracting epistemic information operator $A$ is stratified.

There are five types of *basic epistemic operations*: adding, deleting, moving, replicating, and ttransforming knowledge, and five types of corresponding *basic epistemic information operators*:

adding AD, deletion DL, moving MV, replication REPL, generation GEN and transformation TR epistemic information operators.

**Definition 5.16.** A *transformation epistemic information operator TR* takes a group of knowledge items (may be, one item) from the current knowledge state and transforms it into another group of knowledge items (may be, into one item).

**Definition 5.17.** A *generation epistemic information operator TR* takes a group of knowledge items (may be, one item) from the current knowledge state and generates another group of knowledge items (may be, one item).

The difference between transformation and generation is that in generation the initial group of knowledge items is preserved, while in transformation it is not preserved.

**Lemma 5.4.** AD is equal to GEN with the empty set of the initial knowledge items.

**Definition 5.18.** A *moving epistemic information operator MV* moves a knowledge item from one stratum into another one.

**Definition 5.19.** a) A *replica* of a knowledge item is another knowledge item equivalent to the initial one.

b) A *replication epistemic information operator REPL* makes a replica of a knowledge item and adds it to the current knowledge state.

**Example 5.5.** Let us consider logical knowledge representation in which knowledge items are propositions. Then according to laws of logic there are equivalent propositions. For instance, taking the proposition (1) "*B* implies *A*", we have equivalent propositions (2) "*A* follows from *B*", (3) "If *B*, then *A*", and (4) "*A* is a consequence of *B*". All of them are replicas of one another although they are not copies.

If the proposition (1) belongs to the stratum $K_1$ , then its replication to the stratum $K_2$ can introduce either proposition (1) or proposition (2) or proposition (3) to the stratum $K_2$ , while its copying to the stratum $K_2$ can introduce only proposition (1) to the stratum $K_2$ .

An important special case of a replication epistemic information operator is a *copying epistemic information operator COPY*, which makes a copy of a knowledge item and adds it to the current knowledge state.

Another important special case of a replication epistemic information operator is a *restricted replication epistemic information operator* $REPL_0$ , which replicates a knowledge item and adds it only to a stratum of the current knowledge state that does not have the same replica. One of its special cases is a *restricted copying epistemic information operator* $COPY_0$ , which makes a copy of a knowledge item and adds it only to a stratum of the current knowledge state that does not have the same replica. Operators $REPL_0$ and $COPY_0$ are used in stratified M-spaces not to make these spaces stratified M-multispaces.

**Lemma 5.5.** The operator $COPY_0$ can copy a knowledge item only to a different stratum, *i.e.*, if $a \in K_i$ and $COPY_0 a \in K_j$ , then $i \neq j$.

Indeed, if this condition is violated, then the initial M-space is converted to an M-multispace.

Complex information operations and operators are studied in [76].

**Definition 5.20.** An epistemic information operator *C* is called the *sequential composition* of an epistemic information operator *A* with an epistemic information operator *B* if $C(x)$ is defined and equal

to $B(A(x))$ when: 1) $A(x)$ is defined and belongs to the domain of $B$ ; 2) $B(A(x))$ is defined. Otherwise, $C$ gives no result being applied to $x$, *i.e.*, $C(x) = *$.

Taking sequential composition of an epistemic information operator $A$ with itself, we obtain sequential powers $A^n$ of the operator $A$.

In the general case, the sequential composition of epistemic information operators is not commutative in M-spaces as the following example demonstrates.

**Example 5.1.** Let us consider a structured M-space $\boldsymbol{M} = \{\mathrm{KS}_{\boldsymbol{M}} ; \mathrm{OS}_{\boldsymbol{M}} \}$ where $KS_{\boldsymbol{M}} = = \cup_{i\in I} KS_{\boldsymbol{M}i}$ . In this space, the operator $MV_{ij}^{a}$ moves an element $a$ from the stratum $KS_{\boldsymbol{M}i}$ to the stratum $KS_{\boldsymbol{M}j}$ and does not change other elements from $KS_{\boldsymbol{M}}$ . Taking the sequential composition of such operators, we have

$$MV_{ij}^{a} \circ MV_{ik}^{a} = MV_{ij}^{a} \neq MV_{ik}^{a} \circ MV_{ij}^{a} = MV_{ik}^{a}$$

if $i \neq j$, $k \neq j$, and $i \neq k$. Thus, operators $MV_{ij}^{a}$ do not commute with one another.

At the same time, all these operators are idempotents, *i.e.*, $MV_{ij}^{a} \circ MV_{ij}^{a} = MV_{ij}^{a}$.

It is necessary to remark that in a structured M-multispace $\boldsymbol{M}$ with an infinite number of elements $a$ in each stratum $KS_{\boldsymbol{M}i}$ , operators $MV_{ij}^{a}$ and $MV_{ik}^{a}$ commute with one another. This demonstrates difference between M-spaces and M-multispaces.

**Proposition 5.1.** If $A$ and $B$ are closed (closed in a Mizzaro space $K_i$) operators, then their sequential composition $A \circ B$ is also a closed (closed in a Mizzaro space $K_i$) operator.

Indeed, if $A$ and $B$ are closed epistemic information operators in a structured M-multispace $\boldsymbol{M}$, then for any $j\in J$ and for any $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$ , we have $A(K_{ij}) \subseteq K_{ij}$ and $B(K_{ij}) \subseteq K_{ij}$ . Thus, $(A \circ B)(K_{ij}) = B(A(K_{ij}) \subseteq B(K_{ij}) \subseteq K_{ij}$ .

For closed in a Mizzaro space $K_i$ operators, the proof is similar.

**Proposition 5.2.** If $A$ and $B$ are contracting operators, then their sequential composition $A \circ B$ is also a contracting operator.

Proof is similar to the proof of Proposition 5.1.

**Proposition 5.3.** If $A$ and $B$ are stratified operators, then their sequential composition $A \circ B$ is also a stratified operator.

Proof is similar to the proof of Proposition 5.1.

**Proposition 5.4.** If $A$ and $B$ are (strictly) monotone [antitone] operators, then their sequential composition $A \circ B$ is also a (strictly) monotone [antitone] operator.

Indeed, if $A$ and $B$ are monotone epistemic information operators in a structured M-space $\boldsymbol{M}$, then for any $K_i$ from $\mathrm{KS}_{\boldsymbol{M}}$ , we have $A(K_{ij}) \subseteq K_{ik}$ with $k \geq j$ and $B(K_{ik}) \subseteq K_{ih}$ with $h \geq k$. Thus, $(A \circ B)(K_{ij}) = B(A(K_{ij}) \subseteq B(K_{ik}) \subseteq K_{ih}$ with $h \geq j$.

Considerations for strictly monotone, antitone and strictly antitone epistemic information operators are similar.

Let us consider an M-space $\boldsymbol{M}$ with a finite linear stratification.

**Proposition 5.5.** For any monotone and any antitone epistemic information operator $A$, there is a number $n$ such that the sequential power $A^n$ is also a closed epistemic information operator.

Indeed, if $A$ is a monotone epistemic information operator in a structured M-space $\boldsymbol{M}$, then making each step, it either increases the number of the stratum or the image of a stratum remains in the same stratum. If the second case is true for all strata of $\boldsymbol{M}$, then $A$ itself is a closed epistemic information operator. Otherwise, $A$ can increase the number of a stratum only for a finite number of steps because

there are only a finite number of strata in ***M***. Thus, after some number of repetitions, the image of a stratum remains in the same stratum. Taking the largest number of such steps, we obtain the necessary number $n$.

Note that $n$ cannot be larger than the number of strata in ***M***.

Let us explore relations between basic epistemic information operators.

**Definition 5.21** [77]. Two operators $A$ and $B$ are *functionally equivalent* if they have the same definability domain $D$ and $A(x) = B(x)$ for any element $x$ from $D$.

**Proposition 5.6.** A transformation epistemic information operator *TR* is functionally equivalent to the sequential composition of a deletion epistemic information operator *DEL* and adding epistemic information operator *AD* that act in the same stratum of the M-space.

Indeed, if *TR* takes items $a_1, a_2, \ldots, a_n$ from $KS_{\mathbf{M}}$ and transforms them into $b_1, b_2, \ldots, b_m$, then it is possible to achieve the same result by deleting $a_1, a_2, \ldots, a_n$ and adding $b_1, b_2, \ldots, b_m$ to the corresponding stratum of $KS_{\mathbf{M}}$.

**Proposition 5.7.** A moving epistemic information operator *MV* is functionally equivalent to deletion of a knowledge item in one stratum and adding the same knowledge item to another stratum.

**Proposition 5.8.** A replication epistemic information operator *REPL* is functionally equivalent to adding an equivalent knowledge item to the corresponding stratum.

**Proposition 5.9.** For any M-space ***M***, there is a superspace ***H***, in which all deletion and adding epistemic information operators *DEL* and *AD* in ***M*** are functionally equivalent to moving epistemic information operators *MV* in ***H***.

Proof. To build a superspace ***H*** with the necessary properties, we add one more stratum $E$ called the *external stratum* to the initial M-space ***M***. In addition, we assume that $E$ contains all elements from the universal set (multiset) ***W*** and each element has infinitely many copies in $E$. In this case, any deletion of an element $a$ from a state $K$ from ***M*** is equivalent to moving the same element $a$ to the stratum $E$. In a similar way, any addition of an element $a$ to a state $K$ from ***M*** is equivalent to moving the same element $a$ from the stratum $E$ to the state $K$.

Proposition is proved.

**Proposition 5.10.** A moving epistemic information operator *MV* can be (functionally) simulated by copy *COPY* and deletion *DEL* epistemic information operators.

Indeed, instead of moving a knowledge item $a$ from a stratum $K_i$ of a state $K$ to a stratum $K'_j$ of a state $K'$, it is possible to copy $a$ from $K_i$ to $K'_i$ and then to delete this element from $K_i$.

**Proposition 5.11.** A generation epistemic information operator *GEN* is functionally equivalent to the sequential composition of a transformation epistemic information operator *TR* and adding epistemic information operator *AD* that act in the same stratum of the M-space.

Proof is similar to the proof of Proposition 5.10.

**Proposition 5.12.** A transformation epistemic information operator *TR* is functionally equivalent to the sequential composition of a generation epistemic information operator *GEN* and deletion epistemic information operator *DEL* that act in the same stratum of the M-space.

Proof is similar to the proof of Proposition 5.10.

**Definition 5.22.** A system ***B*** of epistemic information operators is an *operator basis* of an M-space ***M*** if any $A$ from $OS_{\boldsymbol{M}}$ is a composition of elements from ***B***.

Operator bases can be useful in many situations. For instance, knowing properties of operators from such a base and properties of compositions, we can find properties of other operators.

Assuming that all operators in an M-space ***M*** are compositions of basic epistemic information operators, we have the following results.

**Proposition 5.13.** a) {*AD*, *DEL*} is an operator basis of an arbitrary (stratified) M-space ***M***.

b) {*TR*, *MV*} is an operator basis of an arbitrary (stratified) M-space ***M***.

c) {*TR*} is an operator basis of an arbitrary (*i.e.*, non-stratified) M-space ***M***.

Proof is based on Propositions 5.7 - 5.12.

**Proposition 5.14.** a) In a stratified M-space ***M*** with the external stratum, {$REPL_0$ , *DEL*} is an operator basis.

b) In a stratified M-space ***M*** with the external stratum, {*MV*} is an operator basis.

c) In a stratified M-multispace ***M*** with the external stratum, {*REPL*, *DEL*} is an operator basis.

Proof is based on Propositions 5.7 - 5.12.

## 6. Conclusions

Based on the principles of the general theory of information, epistemic information is singled out as a kind of antropic information and modeled by the algebraic construction of M-spaces. M-spaces represent information dynamics by information operators acting in knowledge spaces. The main emphasis of this study is made on stratified knowledge spaces and algebras of epistemic information operators in such spaces.

Obtained results bring us to the following problems.

It is possible to consider not only knowledge but also beliefs as basic components of cognitive infological systems and call information that acts on such systems by the name plausible epistemic information.

**Problem 1.** Mathematically describe and study plausible epistemic information.

**Problem 2.** Study other types of M-space stratifications and operators in these spaces.

**Problem 3.** Study M-spaces in which knowledge items are elements of logics, e.g., propositions or predicates and stratification of which includes the structure of the corresponding logic, e.g., the propositional logic or the first-order predicate logic.

**Problem 4.** Study categories of M-spaces and functors between these categories.

**Problem 5.** Study operations with M-spaces.

In this paper, we studied only content epistemic information operators, while bond epistemic information operators, which act on connections and relations between knowledge items, are also very important.

**Problem 6.** Study bond epistemic information operators.

In this paper, we studied only sequential composition of epistemic information operators, while other types of composition are also very important.

**Problem 7.** Study other compositions of epistemic information operators.

In [1,13], information explications of epistemic information operators is studied in uniform Mizzaro spaces.

**Problem 8.** Explore information explications of epistemic information operators in stratified M-spaces and Mizzaro spaces.

## References


1. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: Singapore, 2010.
2. Bar-Hillel, Y.; Carnap, R. Semantic information. *Br. J. Philos. Sci.* **1958**, *4*, 147-157.
3. Hintikka, J. The Varieties of Information and Scientific Explanation. In *Logic, Methodology and Philosophy of Science III*; van Rootselaar, B., Staal, J.F., Eds.; North-Holland Publishing Company: Amsterdam, The Netherlands, 1968; pp. 311-331.
4. Hintikka, J. Surface Information and Depth Information. In *Information and Inference*; Synthese Library, Humanities Press: New York, NY, USA, 1970; 263-297.
5. Hintikka, J. On Defining Information. *Ajatus* **1971**, *33*, 271–273.
6. Israel, D.; Perry, J. What is Information? In *Information, Language and Cognition*; University of British Columbia Press: Vancouver, BC, Canada, 1990; 1-19.
7. Shreider, Y.A. On the semantic characteristics of information. *Inf. Storage Retr.* **1965**, 2, 221-233.
8. MacKay, D.M. *Information*, *Mechanism and Meaning*; The MIT Press: Cambridge, MA, USA, 1969.
9. Brookes, B.C. The foundations of information science, pt. 1, Philosophical aspects. *J. Inf. Sci.* **1980**, *2*, 125-133.
10. Mizzaro, S. On the Foundations of Information Retrieval. In *Proceedings of the Atti del Congresso Nazionale (AICA'96)*, Roma, Italy, 25–27 September 1996; pp. 363-386.
11. Mizzaro, S. How many relevances in information retrieval? *Interact. Comput.* **1998**, *10*, 303–320.
12. Gackowski, Z.J. What to teach business students in mis courses about data and information. *Issues Informing Sci. Inf. Technol.* **2004**, *1*, 845-867.
13. Mizzaro, S. Towards a Theory of Epistemic Information. In *Information Modelling and Knowledge Bases*; IOS Press: Amsterdam, The Netherlands, 2001; Volume 12, pp. 1-20.
14. Baer, N.; Zeidman, B. Measuring Software Evolution with Changing Lines of Code. In *Proceedings of the 24th International Conference on Computers and Their Applications* (CATA-2009), New Orleans, LA, USA, 8–10 April 2009; pp. 264-170.
15. Burgin, M. *Fundamental Structures of Knowledge and Information*; Academy for Information Sciences: Kiev, Ukraine, 1997; (in Russian).
16. Gardner, R.; Cory, G.A. *The Evolutionary Neuroethology of Paul MacLean: Convergences and Frontiers*; Praeger: New York, NY, USA, 2002.
17. Russell, P. *The Brain Book*; Penguin Books: London, UK, 1992.
18. Assagioli, R. *Psychosynthesis: A Collection of Basic Writings*; Penguin Books: London, UK, 1993.
19. Kenny, A. *Action*, *Emotion and Will*; Routledge: London, UK, 2003.
20. Burgin, M. Information and transformation. *Transformation* **1998/1999**, *1*, 48-53, (in Polish).
21. Von Weizsäcker, C.F. *Die Einheit der Natur*; Deutscher Taschenbuch Verlag: Munich, Germany, 1974.

22. Von Weizsäcker, C.F. *Aufbau der Physik*; Hanser: Munich, Germany, 1985; (Eglish translation: *The Structure of Physics*; Springer: Berlin, Germany, Heidelberg, Germany and New York, NY, USA, 2006).
23. Minsky, M. *The Society of Mind*; Simon and Schuster: New York, NY, USA, 1986.
24. Capurro, R., Fleissner, P., and Hofkirchner, W. Is a Unified Theory of Information Feasible? In *The Quest for a unified theory of information*; Routledge: London, UK, 1999; pp. 9-30.
25. Melik-Gaikazyan, I.V. *Information Processes and Reality*; Nauka: Moscow, Russia, 1997; (in Russian, English summary).
26. Jung, C. On Psychic Energy. In *On the Nature of the Psyche*; Princeton University Press: Princeton, NJ, USA, 1928/1960.
27. Von Grot, N. Die begriffe der seele und der psychischen energie in der psychologie. *Arch. fur Syst. Philos.* **1898**, IV.
28. Colby, K. *Energy and Structure in Psychoanalysis*; Ronald: New York, NY, USA, 1955.
29. Freud, S. *The Standard Edition of the Complete Psychological Works of Sigmund Freud*, Hogarth and the Institute of Psycho-Analysis: London, UK, 1954.
30. Lieberman, H.R. Cognitive methods for assessing mental energy. *Nutr. Neurosci.* **2007**, *10*, 229-242.
31. O'Connor, P.J. Mental energy: Assessing the mood dimension. *Nutr. Rev.* **2006**, *64*, S7-S9.
32. Dahl, H. The panel on "Psychoanalytic Theory of the Instinctual Drives in Relation to Recent Developments". *J. Am. Psychoanal. Assoc.* **1968**, *XVI*, 613-637.
33. MacLean, P.D. *A Triune Concept of the Brain and Behavior*; University of Toronto Press: Toronto, ON, Canada, 1973.
34. MacLean, P.D. On the Origin and Progressive Evolution of the Triune Brain. In *Primate Brain Evolution*; Plenum Press: New York, NY, USA, 1982.
35. Smith, C.U.M. The triune brain in antiquity: Plato, Aristotle, Erasistratus. *J. Hist. Neurosci.* **2010**, *19*, 1-14.
36. Herrmann, N. *The Creative Brain*; Brain Books: Lake Lure, NC, USA, 1990.
37. Cory, G.A. *The Reciprocal Modular Brain in Economics and Politics: Shaping the Rational and Moral Basis of Organization, Exchange, and Choice*; Kluwer Academic/Plenum Publishers: New York, NY, USA, 1999.
38. Zametkin, A.J. Cerebral glucose metabolism in adults with hyperactivity of childhood onset. *N. Engl. J. Med.* **1990**, *323*, 1361-1366.
39. Levine, P.A. *Waking the Tiger: Healing Trauma*; North Atlantic Books: Berkeley, CA, USA, 1999.
40. Patton, P. One world, many minds: Intelligence in the animal kingdom. *Sci. Am.* **2008**, *19*, 72-79.
41. Anderson, R.C. The Notion of Schemata and the Educational Enterprise. In *Schooling and the Acquisition of Knowledge*; Anderson, R.C., Spiro, R.J., Montague, W.E., Eds.; Lawrence Erlbaum: Hillsdale, NJ, USA, 1977.
42. Arbib, M. Schema Theory. In *The Encyclopedia of AI*; Wiley-Interscience: New York, NY, USA, 1992; pp. 1427-1443.
43. Armbruster, B. Schema theory and the design of content-area textbooks. *Educ. Psychol.* **1996**, *21*, 253-276.

44. Burgin, M. Mathematical Schema Theory for Modeling in Business and Industry. In *Proceedings of the 2006 Spring Simulation Multi Conference* (*Spring Sim '06*), Huntsville, AL, USA, 2006; pp. 229-234.
45. Baars, B.J.; Gage, N.M. *Cognition, Brain, and Consciousness*: *Introduction to Cognitive Neuroscience*; Elsevier Science/Academic Press: Amsterdam, The Netherlands, 2007.
46. Carter, R. *Mapping the Brain*; Phoenix Books: Junction, VT, USA, 2003.
47. DeArmond, S.J.; Fusco, M.M.; Dewey, M. *Structure of the Human Brain: A Photographic Atlas*; Oxford University Press: New York, NY, USA, 1989.
48. Dehaene, S. *Reading in the Brain: The Science and Evolution of a Human Invention*; Viking: New York, NY, USA, 2009.
49. Berrios G.E.; Gili M. Will and its disorders. A conceptual history. *Hist. Psychiatry* **1995**, *6*, 87-104.
50. Spence, S.A. Between will and action. *J. Neurol. Neurol. Psychiatry* **2000**, *69*, doi:10.1136/jnnp.69.5.702.
51. Sternberg, R.J. *Beyond IQ*: *A Triarchic Theory of Intelligence*; Cambridge University Press: Cambridge, UK, 1985.
52. Neisser, U. *Cognition and Reality*: *Principles and Implications of Cognitive Psychology*; W.H. Freeman and Company: San Fransisco, CA, USA, 1976.
53. Beck, J. *Cognitive Therapy*: *Basics and Beyond*; Guilford: New York, NY, USA, 1995.
54. Freeman, A.; DeWolf, R. *The 10 Dumbest Mistakes Smart People Make and How to Avoid Them*; Harper Collins Publ.: New York, NY, USA, 1992.
55. Kolb, B.; Whishaw, I.Q. *Fundamentals of Human Neuropsychology*; W.H. Freeman and Co.: New York, NY, USA, 1990.
56. Benferhat, S.; Baida, R. A stratified first order logic approach for access control. *Int. J. Int. Syst.* **2004**, *19*, 817-836.
57. Benferhat, S.; Garcia, L. Handling locally stratified inconsistent knowledge bases. *Stud. Log.* **2002**, *70*, 77-104.
58. Lassez, C.; McAloon, K.; Port, G.S. Stratification and knowledge base management. *J. Symb. Comput.* **1989**, *7*, 509-522.
59. Hunter, A.; Liu, W.R. Knowledge Base Stratification and Merging Based on Degree of Support. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*; Springer-Verlag: Berlin, Germany, 2009; pp. 383-395.
60. Yue, A.; Liu, W.; Hunter, A. Approaches to Constructing a Stratified Merged Knowledge Base. In *Proceedings of the 9th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'07)*, Hammamet, Tunisia, 31 October–2 November 2007; pp. 54-65.
61. Cholewinski, P. Stratified default logic. *Proc. Comp. Sci. Logic'94* **1994**, *933*, 456-470.
62. Burgin, M. Structural Organization of Temporal Databases. In *Proceedings of the 17th International Conference on Software Engineering and Data Engineering* (SEDE-2008), ISCA, Los Angeles, CA, USA, 30 June–2 July 2008; pp. 68-73.
63. Snodgrass, R.T.; Jensen, C.S. *Developing Time-Oriented Database Applications in SQL*; Morgan Kaufmann: San Francisco, CA, USA, 1999.

64. Burgin, M. Measuring Power of Algorithms, Programs, and Automata. In *Artificial Intelligence and Computer Science*; Nova Science Publishers: New York, NY, USA, 2005; pp. 1-61.
65. Burgin, M. *Super-Recursive Algorithms*; Springer: Heidelberg, Germany, 2005.
66. Hamkins, J.D.; Lewis, A. Infinite time turing machines. *J. Symb. Log.* **2000**, *65*, 567-604.
67. Burgin, M. Grammars with Prohibition and Human-Computer Interaction. In *Proceedings of the Business and Industry Simulation Symposium*, Society for Modeling and Simulation International, San Diego, CA, USA, 3–7 April 2005b; pp. 143-147.
68. Okhotin, A. Boolean grammars. *Inf. Comput.* **2004**, *194*, 19-48.
69. Burgin, M. Data, information, and knowledge. *Information* **2004**, *7*, 47-57.
70. Barwise, J.; Perry, J. *Situations and Attitudes*; MIT Press: Cambridge, MA, USA, 1983.
71. Dretske, F.I. *Knowledge and the Flow of Information*; Basil Blackwell: Oxford, UK, 1981.
72. Burgin, M. Logical Tools for Program Integration and Interoperability. In *Proceedings of the IASTED International Conference on Software Engineering and Applications*, MIT, CA, USA, 9–11 November 2004a; pp. 743-748.
73. Cohn, P.M. *Universal Algebra*; Harper&Row, Publ.: New York, NY, USA, 1965.
74. Kohlas, J.; Stärk, R.F. Information algebras and consequence operators. *Log. Universalis* **2007**, *1*, 139-165.
75. Shoenfield, J.R. *Mathematical Logic*; Addison-Wesley: Reading, MA, USA, 1967.
76. Burgin, M. Information algebras. *Control Syst. Mach.* **1997**, *6*, 5-16, (in Russian).
77. Burgin, M. Functional equivalence of operators and parallel computations. *Program. Comput. Softw.* **1980**, *6*, 283-294.

© 2010 by the author; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).

*Information* **2013**, *4*, 124-168; doi:10.3390/info4020124

OPEN ACCESS

*information*

**ISSN 2078-2489**

www.mdpi.com/journal/information

*Article*

# Evolutionary Information Theory

**Mark Burgin**

Computer Science Department, University of California, Los Angeles, 405 Hilgard Ave. Los Angeles, CA 90095, USA; E-Mail: markburg@cs.ucla.edu

*Received: 27 December 2012; in revised form: 7 March 2013 / Accepted: 13 March 2013 / Published: 11 April 2013*

---

**Abstract:** Evolutionary information theory is a constructive approach that studies information in the context of evolutionary processes, which are ubiquitous in nature and society. In this paper, we develop foundations of evolutionary information theory, building several measures of evolutionary information and obtaining their properties. These measures are based on mathematical models of evolutionary computations, machines and automata. To measure evolutionary information in an invariant form, we construct and study universal evolutionary machines and automata, which form the base for evolutionary information theory. The first class of measures introduced and studied in this paper is evolutionary information size of symbolic objects relative to classes of automata or machines. In particular, it is proved that there is an invariant and optimal evolutionary information size relative to different classes of evolutionary machines. As a rule, different classes of algorithms or automata determine different information size for the same object. The more powerful classes of algorithms or automata decrease the information size of an object in comparison with the information size of an object relative to weaker4 classes of algorithms or machines. The second class of measures for evolutionary information in symbolic objects is studied by introduction of the quantity of evolutionary information about symbolic objects relative to a class of automata or machines. To give an example of applications, we briefly describe a possibility of modeling physical evolution with evolutionary machines to demonstrate applicability of evolutionary information theory to all material processes. At the end of the paper, directions for future research are suggested.

**Keywords:** information; evolution; evolutionary machine; information size; optimality; modeling; universality

---

## 1. Introduction

Evolutionary information theory studies information in the context of evolutionary processes. All operations with information, information acquisition, transmission and processing, are treated from the evolutionary perspective. There are many evolutionary information processes in nature and society. The concept of evolution plays important roles in physics, biology, sociology and other scientific disciplines. In general, evolution is one of the indispensable processes of life, as well as of many other natural processes. For instance, human cognition in general and scientific cognition in particular is an evolutionary information process. Indeed, over centuries, science has been obtaining more exact and comprehensive knowledge about nature. Although many important discoveries and scientific accomplishments were considered as scientific revolutions, the whole development of science was essentially evolutionary.

After biologists found basic regularities of biological evolution, computer scientists began simulating evolutionary processes and utilizing operations found in nature for solving problems with computers. In such a way, they brought forth evolutionary computation, inventing different kinds of strategies and procedures, such as genetic algorithms or genetic programming, which imitated natural biological processes. The development of evolutionary computation has essentially influenced computational science and information processing technology. As Garzon writes [1], computational models should always be embedded in an environment and therefore be subject to an evolutionary process that constantly seeks to add fundamental new components through learning, self-modification, automatic "re-programming", and/or evolution.

Here we consider evolution controlled by evolutionary algorithms and modeled by evolutionary automata and machines studied in [2–4]. Evolutionary automata and machines form mathematical foundations for evolutionary computations and genetic algorithms, which in turn, serve as a tool for the development of evolutionary information theory resembling the usage of conventional algorithms and automata for construction of algorithmic information theory [5].

Algorithmic information theory is based on the concept of Kolgmogorov or algorithmic complexity, which provides means to measure the intrinsic information related to objects via their algorithmic description length. In turn, algorithmic or Kolmogorov complexity is based on appropriate classes of Turing machines [6] and the inductive algorithmic complexity is based on inductive Turing machines [7,8].

Evolutionary information theory stems from the assumption that evolution is performed by some means, which are modeled by abstract automata, algorithms or machines, which work in the domain of strings and perform evolutionary computations. The input string is a carrier of information about the output string, which represents the result of evolution. Based on these considerations, it is natural to define the information size of the output string $x$ as the minimum quantity of information needed to evolve this string. This quantity of input information is naturally measured by the length of the shortest input string needed to evolve the string $x$ by means of the used system of evolutionary automata, algorithms or machines.

The algorithmic approach explicates an important property of information, connecting information to means used for accessing and utilizing information. Information is considered not as some inherent property of different objects but is related to algorithms that use, extract or produce this information. In

this context, a system (person) with more powerful algorithms for information extraction and management can get more information from the same carrier and use this information in a better way than a system that has weaker algorithms and more limited abilities. This correlates with the conventional understanding of information.

Evolutionary information reflects aspects and properties of information related to evolutionary processes. Many information processes, such as software development or computer information processing, have evolutionary nature. Evolutionary approach explicates important properties of information, connecting it to natural computations and biological systems. At the same time, in the context of pancomputationalism or digital physics (cf., for example, [9–14], the universe is considered as a huge computational structure or a network of computational processes, which following fundamental physical laws, compute (dynamically develop) its own next state from the current one. As a result, the universe or reality is essentially informational, while all information flows in the universe are carried out by computational processes, while all evolutionary processes are performed by evolutionary automata. There are several computational models, such as natural computing, that are suitable for the idea of pancomputationalism [9,10]. Thus, from the perspective of pancomputationalism, the algorithmic approach to evolutionary information theory is the most encompassing methodology in dealing with information processes going on in the universe.

Developing the main ideas of algorithmic information theory in the direction of evolutionary processes, here we introduce and study two kinds of evolutionary information: evolutionary information necessary to develop a constructive object by a given system of evolutionary algorithms (evolutionary automata) and evolutionary information in an object, e.g., in a text that allows making simpler development of another object by a given system of evolutionary algorithms (evolutionary automata). Respectively, we have two basic evolutionary information measures: the *quantity of evolutionary information about* an object, which is also called the evolutionary *information size* of an object, and the *quantity of evolutionary information in* an object.

This paper is organized as follows. In Section 2, we introduce the necessary concepts and constructions from the theory of evolutionary computations, machines and automata, further developing ideas from [2–4]. In Section 3, we construct and study universal evolutionary machines and automata, which form the base for evolutionary information size of symbolic objects and for evolutionary information in symbolic objects. In Section 4, we introduce and study evolutionary information size of symbolic objects with respect to a class of automata/machines or with respect to a single automaton/machine. In particular, it is proved that there is an invariant and optimal evolutionary information size with respect to different classes of periodic evolutionary machines.

Informally, the evolutionary information size of an object $x$ with respect to a class $\mathbf{H}$ shows how much information it is necessary for building (computing or constructing) this object by algorithms/automata from the class $\mathbf{H}$. The evolutionary information size of an object $x$ with respect to an automaton/machine $A$ shows how much information it is necessary for building (computing or constructing) this object by the automaton/machine $A$. Thus, it is natural that different automata need different quantity of evolutionary information about an object $x$ to build (compute or construct) this object.

In Section 5, evolutionary information in symbolic objects is studied based on the quantity of this information with respect to a class of automata/machines or with respect to a single automaton/machine. Informally, the quantity of evolutionary information in an object/word $y$ about an

object $x$ with respect to a class **Q** shows to what extent utilization of information in $y$ reduces information necessary for building (computing or constructing) this object in the class **Q** without any additional information. The quantity of evolutionary information in an object $y$ about an object $x$ with respect to an automaton/machine $A$ shows to what extent utilization of information in $y$ reduces information necessary for building (computing or constructing) the object $x$ by $A$ without any additional information. It is natural that evolutionary information in an object depends on the automata/machines that extract information from this object and use this information.

In Section 6, we briefly explain a possibility of modeling physical evolution with evolutionary machines to demonstrate applicability of evolutionary information theory to all material processes. The modeling technology is based on the basic physical theory called loop quantum gravity, in which geometry of space-time is described by spin networks and matter is represented by the nodes of these networks [15–17].

In Conclusion, open problems for further research are suggested.

The author is grateful to unknown reviewers for their useful comments.

## 2. Evolutionary Machines and Computations

Evolutionary computations are artificial intelligence processes based on natural selection and evolution. Evolutionary computations are directed by evolutionary algorithms and performed by evolutionary machines. In technical terms, an evolutionary algorithm is a probabilistic search algorithm directed by the chosen fitness function. To formalize this concept in mathematically rigorous terms, a formal algorithmic model of evolutionary computation—an *evolutionary automaton* also called an *evolutionary machine* is defined.

Let **K** be a class of automata/machines with input and two outputs.

**Definition 2.1.** A *general evolutionary* **K**-*machine* (**K**-GEM), also called *general evolutionary* **K**-*automaton*, is a (possibly infinite) sequence $E = \{A[t]; t = 1, 2, 3, ...\}$ of automata $A[t]$ from **K** each working on populations/generations $X[i]$ which are coded as words in the alphabet of the automata from **K** where:

- the goal of the **K**-GEM $E$ is to build a population $Z$ satisfying the search condition;
- the automaton $A[t]$ called a *component*, or a *level automaton*, of $E$ represents (encodes) a one-level evolutionary algorithm that works with input populations/generations $X[i]$ of the whole population by applying the variation operators $v$ and selection operator $s$;
- the first population/generation $X[0]$ is given as input to $E$ and is processed by the automaton $A[1]$, which generates/produces the second population/generation $X[1]$ as its transfer output, which goes to the automaton $A[2]$;
- for all $t = 1, 2, 3, ...$, the automaton $A[t]$, which receives the population/generation $X[i]$ as its input either from $A[t + 1]$ or from $A[t - 1]$, then $A[t]$ applies the variation operator $v$ and selection operator $s$ to the input population/generation $X[i]$, producing the population/generation $X[i + 1]$ as its transfer output and if necessary, sending this population/generation either to $A[t + 1]$ or to $A[t - 1]$ to continue evolution.

Each automaton $A[t]$ has one input channel for receiving its input and two output channels. One is called the *transfer channel* and used for transferring data (the population/generation $X[i + 1]$, which is called the *transfer output*) either to $A[t + 1]$ or to $A[t - 1]$. The second channel called the *outcome channel* is used for producing the *outcome* of the automaton $A[t]$ when it starts working with the input $X[i]$. When $A[t]$ receives its input from $A[t + 1]$, it is called the *upper input*. When $A[t]$ receives its input from $A[t - 1]$, it is called the *lower input*.

Note that $i$ is always larger than or equal to $t - 1$ in this schema of evolutionary processing. Besides, it is possible to simulate two output channels by one output channel separating two parts in the output words—one part serving as the transfer output and the other serving as the outcome output. Components of general evolutionary **K**-machines perform multiple computations in the sense of [18]. However, it is possible to code each population/generation $X[i]$ by a single word. This allows us to use machines/automata that work with words for building evolutionary machines/automata.

We denote the class of all general evolutionary **K**-machines **GEAK**.

The desirable search condition is the optimum of the fitness performance measure $f(x[i])$ of the best individual from the population/generation $X[i]$. There are different modes of the EM functioning and different termination strategies. When the search condition is satisfied, then working in the recursive mode, the EM $E$ halts ($t$ stops to be incremented), otherwise a new input population/generation $X[i + 1]$ is generated by $A[t]$. In the inductive mode, it is not necessary to halt to give the result. When the search condition is satisfied and $E$ is working in the inductive mode, the EM $E$ stabilizes (the population/generation $X[i]$ stops changing), otherwise a new input population/generation $X[i + 1]$ is generated by $A[t]$.

Let us consider some examples of evolutionary **K**-machines.

**Example 2.1.** A *general evolutionary finite automaton* (GEFA) is a general evolutionary machine $E = \{G[t]; t = 1, 2, 3, ...\}$ in which all level automata are finite automata $G[t]$ working on the input population/generation $X[i]$ with the generation parameter $i = 0, 1, 2, 3, ....$

We denote the class of all general evolutionary finite automata by **GEFA**.

It is possible to take as **K** deterministic finite automata, which form the class **DFA**, or nondeterministic finite automata, which form the class **NFA**. This gives us two classes of evolutionary finite automata: **GEDFA** of all deterministic general evolutionary finite automata and **GENFA** of all nondeterministic general evolutionary finite automata.

**Example 2.2.** A *general evolutionary Turing machine* (GETM) $E = \{T[t]; t = 1, 2, 3, ...\}$ is a general evolutionary machine $E$ in which all level automata are Turing machines $T[t]$ working on the input population/generation $X[i]$ with the generation parameter $i = 0, 1, 2, 3, ....$ We denote the class of all general evolutionary Turing machines by **GETM**.

Turing machines $T[t]$ as components of $E$ perform multiple computations [18]. Variation and selection operators are recursive to allow performing level computation by Turing machines.

**Example 2.3.** A *general evolutionary inductive Turing machine* (GEITM) $EI = \{M[t]; t = 1, 2, 3, ...\}$ is a general evolutionary machine $E$ in which all level automata are inductive Turing machines $M[t]$ [7,19] working on the input population/generation $X[i]$ with the generation parameter $i = 0, 1, 2, 3, ....$

Simple inductive Turing machines are abstract automata (models of algorithms) closest to Turing machines. The difference between them is that a Turing machine always gives the final result after a finite number of steps and after this it stops or, at least, informs when the result is obtained. Inductive Turing machines also give the final result after a finite number of steps, but in contrast to Turing machines, inductive Turing machines do not always stop the process of computation or inform when the final result is obtained. In some cases, they do this, while in other cases they continue their computation and give the final result. Namely, when the content of the output tape of a simple inductive Turing machine forever stops changing, it is the final result.

We denote the class of all general evolutionary inductive Turing machines by **GEITM**.

**Definition 2.2.** A general evolutionary inductive Turing machine (GEITM) $EI = \{M[t]; t = 1, 2, 3, ...\}$ has order $n$ if all inductive Turing machines $M[t]$ have order less than or equal to $n$ and at least, one inductive Turing machine $M[t]$ has order $n$.

We remind that inductive Turing machines with recursive memory are called *inductive Turing machines of the first order* [7]. The memory $E$ is called *n-inductive* if its structure is constructed by an inductive Turing machine of order $n$. Inductive Turing machines with $n$-inductive memory are called *inductive Turing machines of order* $n + 1$.

We denote the class of all general evolutionary inductive Turing machines of order $n$ by $\mathbf{GEITM}_n$.

**Example 2.4.** A *general evolutionary limit Turing machine* (GELTM) $EI = \{H[t]; t = 1, 2, ...\}$ is a general evolutionary machine $E$ in which all level automata are limit Turing machines $H[t]$ [7] working on the input population/generation $X[i]$ with the generation parameter $i = 0, 1, 2, ....$

When the search condition is satisfied, then the ELTM $EI$ stabilizes (the population $X[t]$ stops changing), otherwise a new input population/generation $X[i + 1]$ is generated by $H[t]$.

We denote the class of all general evolutionary limit Turing machines of the first order by **GELTM**.

**Definition 2.3.** General evolutionary **K**-machines from **GEAK** are called *unrestricted* because sequences of the level automata A[$t$] and the mode of the evolutionary machines functioning are arbitrary.

For instance, there are unrestricted evolutionary Turing machines when **K** is equal to **T** and unrestricted evolutionary finite automata when **K** is equal to **FA**.

Using different classes **K**, we obtain the *componential classification* of evolutionary machines defined by the type of the level automata, *i.e.*, automata used as their components. For instance, the automata in **K** can be deterministic, nondeterministic or probabilistic. Another classification of evolutionary machines is called the *sequential classification* and defined by the type of sequences of the level automata.

**Definition 2.4.** If $\boldsymbol{Q}$ is a type of sequences of the level automata, then evolutionary **K**-machines in which sequences of the level automata have type $\boldsymbol{Q}$ are called *$\boldsymbol{Q}$-formed general evolutionary* **K**-*machines* and their class is denoted by $\mathbf{GEAK}^{\boldsymbol{Q}}$ for general machines.

It gives us the following types of evolutionary **K**-machines:

1. When the type $\boldsymbol{Q}$ contains all sequences of the length $n$, we have *n-level general evolutionary* **K**-*machines*, namely, an evolutionary **K**-machine (evolutionary **K**-automaton) $E = \{A[t]; t = 1, 2, 3, ..., n\}$ is *n-level*. We denote the class of all $n$-level general evolutionary **K**-machines by $\mathbf{BGEAK}_n$ .

2. When the type $\boldsymbol{Q}$ contains all finite sequences, we have *bounded general evolutionary* **K**-*machines*, namely, an evolutionary **K**-machine (evolutionary **K**-automaton) $E = \{A[t]; t = 1, 2, 3, ..., n\}$ is called *bounded*. We denote the class of all bounded general evolutionary **K**-machines by **BGEAK**.

Some classes of bounded evolutionary **K**-machines are studied in [3,4] for such classes **K** as finite automata, push down automata, Turing machines, or inductive Turing machines, *i.e.*, such classes as bounded basic evolutionary Turing machines or bounded basic evolutionary finite automata

3. When the type $\boldsymbol{Q}$ contains all periodic sequences, we have *periodic general evolutionary* **K**-*machines*, namely, an evolutionary **K**-machine (evolutionary **K**-automaton) $E = \{A[t]; t = 1, 2, 3, ...\}$ of automata $A[t]$ from **K** is called *periodic* if there is a finite initial segment of the sequence $\{A[t]; t = 0, 1, 2, 3, ...\}$ such that the whole sequence is formed by infinite repetition of this segment. We denote the class of all periodic general evolutionary **K**-machines by **PGEAK**.

Some classes of periodic evolutionary **K**-machines are studied in [4] for such classes **K** as finite automata, push down automata, Turing machines, inductive Turing machines, and limit Turing machines. Note that while in a general case, evolutionary automata cannot be codified by finite words, bounded and periodic evolutionary automata can be codified by finite words.

4. A sequence $\{a_i; i = 1, 2, 3, \ldots\}$ is called *almost periodic* if this sequence consists of two parts—a finite sequence $\{a_1, a_2, a_3, \ldots, a_k\}$ and a periodic sequence $\{a_{k+i}; i = 1, 2, 3, \ldots\}$ that has a finite initial segment such that the whole sequence is formed by infinite repetition of this segment. When the type $\boldsymbol{Q}$ contains all almost periodic sequences, we have *almost periodic general evolutionary* **K**-*machines*. We denote the class of all almost periodic general evolutionary **K**-machines by **APGEAK**.

5. When for each sequence $E$ from $\boldsymbol{Q}$, there is a recursive algorithm that generates all components of $E$, we have *recursively generated general evolutionary* **K**-*machines*. We denote the class of all recursively generated general evolutionary **K**-machines by **RGEAK**.

6. When for each sequence $E$ from $\boldsymbol{Q}$, there is an inductive algorithm that generates all components of $E$, we have *inductively generated general evolutionary* **K**-*machines*. We denote the class of all inductively generated general evolutionary **K**-machines by **IGEAK**.

7. When for each sequence $E$ from $\boldsymbol{Q}$, any of its components $A[t]$ is generated by another component $A[l]$ of $E$ with $l < t$, we have *self-generated general evolutionary* **K**-*machines*. We denote the class of all self-generated general evolutionary **K**-machines by **SGEAK**.

There is a natural inclusion of these classes:

$$\mathbf{BGEAK}_n \subseteq \mathbf{BGEAK} \subseteq \mathbf{PGEAK} \subseteq \mathbf{APGEAK} \subseteq \mathbf{RGEAK} \subseteq \mathbf{IGEAK}$$

and

**IGEAK** ⊆ **SGEAK** when the class **K** is, at least, as powerful as the class **ITM** of all inductive Turing machines.

Definitions imply the following result.

Let us consider two classes **K** and **H** of automata and two types $\boldsymbol{Q}$ and $\boldsymbol{P}$ of sequences.

**Proposition 2.1.** If **K** ⊆ **H** and $\boldsymbol{Q} \subseteq \boldsymbol{P}$, then **GEAK** ⊆ **GEAH** and $\mathbf{GEAK}^{\boldsymbol{Q}} \subseteq \mathbf{GEAH}^{\boldsymbol{P}}$.

As it is possible to simulate any $n$-level general evolutionary **K**-machine by a periodic general evolutionary **K**-machine, we have the following result

**Proposition 2.2.** Any function computable in the class **BGEAK** is also computable in the class **PGEAK**.

*Proof*. Let us take a bounded general evolutionary **K**-machine (evolutionary **K**-automaton) $E = \{A[t]; t = 1, 2, 3, ..., n\}$. Then it is possible to consider a periodic general evolutionary **K**-machine (evolutionary **K**-automaton) $H = \{C[t]; t = 1, 2, 3, ..., n, \ldots\}$ in which $C[t] = A[r]$ with $0 < r \leq n$ and $t = r + kn$ for some natural number $k$. By this definition, $H$ is a periodic general evolutionary **K**-machine and its period coincides with the evolutionary machine $E$. As the evolutionary machine $H$ does not have the rules for transmission of results from the component $C[n]$ to the component $C[n + 1]$, all computations in $H$ go in its first $n$ components. Thus, the evolutionary machine $H$ exactly simulates the evolutionary machine $E$ computing the same function.

Proposition is proved.

As any periodic general evolutionary **K**-machine is almost periodic, while any almost periodic general evolutionary **K**-machine is recursively generated, we have the following results.

**Proposition 2.3.** (a) Any function computable in the class **PGEAK** is also computable in the class **APGEAK**; (b) Any function computable in the class **APGEAK** is also computable in the class **RGEAK**.

It is proved that it is possible to simulate any Turing machine by some inductive Turing machine [7]. Thus, any recursively generated general evolutionary **K**-machine is also an inductively generated general evolutionary **K**-machine. This gives us the following result.

**Proposition 2.4.** Any function computable in the class **RGEAK** is also computable in the class **IGEAK**.

Another condition on evolutionary machines determines their mode of functioning or computation. There are three *cardinal global modes* (*types*) of evolutionary automaton computations:

1. *Bounded finite evolutionary computations* are performed by an evolutionary automaton $E$ when in the process of computation, the automaton $E$ activates (uses) only a finite number of its components $A[t]$ and there is a constant $C$ such that time of all computations performed by each $A[t]$ is less than $C$.

2. *Unbounded finite* (*potentially infinite*) *evolutionary computations* are performed by an evolutionary automaton $E$ when in the process of computation, the automaton $E$ activates (uses) only a finite number of its components $A[t]$ although this number is not bounded and time of computations performed by each $A[t]$ is finite but not bounded.

3. *Infinite evolutionary computations* are performed by an evolutionary automaton $E$ when in the process of computation, the automaton $E$ activates (uses) an infinite number of its components $A[t]$.

There are also three *effective global modes* (types) of evolutionary automata computations:

1. In the *halting global mode*, the result of computation is defined only when the process halts. This happens when one of the level automata $A[t]$ of the evolutionary machine $E$ does not give the transfer output and stops itself giving some outcome.

2. In the *inductive global mode*, the result is defined by the following rule: if for some $t$, the outcome $O[t]$ produced by the level automata $A[t]$ stops changing, *i.e.*, $O[t] = O[q]$ for all $q > t$, then $O[t]$ is the result of computation.

3. In the *limit global mode*, the result of computation is defined as the limit of the outcomes $O[t]$.

In [3], inductive and limit global modes were studied for basic evolutionary automata/machines but they were defined in a little bit different way: (a) an evolutionary automaton/machine functions in the inductive global mode when the result is defined by the following rule: if for some $t$, the generation $X[t]$ stops changing, *i.e.*, $X[t] = X[q]$ for all $q > t$, then $X[t]$ is the result of computation; (b) an evolutionary automaton/machine functions in the limit global mode when the result of computation is defined as the limit of the generations $X[t]$.

The new definition given here is more flexible because when the outcomes of all level automata of an evolutionary machine coincide with the transfer outputs of the same automata, the new definitions coincide with the previously used definitions.

Effective modes can be also local. There are three *effective local modes* (types) of evolutionary automaton/machine computations:

1. In the *halting local mode*, the result of computation of each component $A[t]$ is defined only when the computation of $A[t]$ halts;

2. In the *inductive local mode*, the result of computation of each component $A[t]$ is defined by the following rule: if at some step $i$, the result of $A[t]$ stops changing, then it is the result of computation of $A[t]$;

3. In the *limit local mode*, the result of computation of $A[t]$ is defined as the limit of the intermediate outputs of $A[t]$.

Local modes of evolutionary **K**-machines are determined by properties of the automata from **K**. For instance, when **K** is the class of finite automata or Turing machines, then the corresponding evolutionary **K**-machines, *i.e.*, evolutionary finite automata and evolutionary Turing machines [2], can work only in the halting local mode. At the same time, evolutionary inductive Turing machines [3] can work both in the halting local mode and in the inductive local mode.

Local and global modes are orthogonal to the three traditional modes of computing automata: *computation*, *acceptation* and *decision*/*selection* [20].

Existence of different modes of computation shows that the same algorithmic structure of an evolutionary automaton/machine $E$ provides for different types of evolutionary computations.

There are also three *directional global modes* (types) of evolutionary automaton/machine computations:

1. The *direct mode* when the process of computation goes in one direction from $A[t]$ to $A[t + 1]$.

2. The *recursive mode* when in the process of computation, it is possible to reverse the direction of computation, *i.e.*, it is possible to go from higher levels to lower levels of the automaton, and the result is defined after finite number of steps.

3. The *recursive mode with n reversions* when in the process of computation only $n$ reversions are permissible.

Evolutionary **K**-machines that work only in the direct mode are called basic evolutionary **K**-machines. Namely, we have the following definition.

**Definition 2.5.** A *basic evolutionary* **K**-*machine* (**K**-BEM), also called *basic evolutionary* **K**-*automaton*, is a (possibly infinite) sequence $E = \{A[t]; t = 1, 2, 3, ...\}$ of automata $A[t]$ from **K** each working on the population $X[t]$ ($t = 0, 1, 2, 3, ...$) where: the goal of the **K**-BEM $E$ is to build a population $Z$ satisfying the search condition; the automaton $A[t]$ called a *component*, or more exactly, a *level automaton*, of $E$ represents (encodes) a one-level evolutionary algorithm that works with the generation $X[t - 1]$ of the population by applying the variation operators $v$ and selection operator $s$; the first generation $X[0]$ is given as input to $E$ and is processed by the automaton $A[1]$, which generates/produces the first generation $X[1]$ as its transfer output, which goes to the automaton $A[2]$; for all $t = 1, 2, 3, ...$, the generation $X[t + 1]$ is obtained as the transfer output of $A[t + 1]$ by applying the variation operator $v$ and selection operator $s$ to the generation $X[t]$ and these operations are performed by the automaton $A[t + 1]$, which receives $X[t]$ as its input.

We denote the class of all basic evolutionary machines with level automata from **K** by **BEAK**. When the class **K** is fixed we denote the class of all basic evolutionary machines (BEM) with level automata from **K** by **BEA**.

As any basic evolutionary **K**-machine is also a general evolutionary **K**-machine, we have inclusion of classes **BEAK** ⊆ **GEAK**.

Similar to the class **GEAK** of general evolutionary **K**-machines, the class **BEAK** also contains important subclasses:

**BBEAK**$_n$ denotes the class of all $n$-level basic evolutionary **K**-machines.

**BBEAK** denotes the class of all bounded basic evolutionary **K**-machines.

**PBEAK** denotes the class of all periodic basic evolutionary **K**-machines.

**APBEAK** denotes the class of all almost periodic basic evolutionary **K**-machines.

**RBEAK** denotes the class of all recursively generated basic evolutionary **K**-machines.

**IBEAK** denotes the class of all inductively generated basic evolutionary **K**-machines.

**SBEAK** denotes the class of all self-generated basic evolutionary **K**-machines.

There is a natural inclusion of these classes:

$$\mathbf{BBEAK}_n \subseteq \mathbf{BBEAK} \subseteq \mathbf{PBEAK} \subseteq \mathbf{APBEAK} \subseteq \mathbf{RBEAK} \subseteq \mathbf{IBEAK}$$

and **IBEAK** ⊆ **SBEAK** when the class **K** is, at least, as powerful as the class **ITM** of all inductive Turing machines.

Note that all considered above modes of evolutionary computations are used both in general and in basic evolutionary **K**-machines.

Many results demonstrate that evolutionary **K**-machines have, as a rule, more computing power than automata from **K**. Here one more of such results is presented.

**Theorem 2.1.** For any simple inductive Turing machine $M$, there is a basic periodic evolutionary Turing machine $E$ that is functionally equivalent to $M$.

*Proof.* Taking a simple inductive Turing machine $M$, we build a basic periodic evolutionary Turing machine $E = \{A[t]; t = 1, 2, 3, ...\}$ by the following procedure. First, we equip the machine $M$ with one more output tape identical to the first one, obtaining the Turing machine $M_q$. One of these tapes is used for the transition output and the other one stores the outcome.

The Turing machine $M_q$ works in the following way. Taking the input word, $M_q$ processes it by the rules of the machine $M$. However, when the machine $M$ produces the first output, the machine $M_q$ writes the same output in both output tapes and stops functioning. Thus, $M_q$ is a Turing machine because it either halts and gives the result or does not produce any output (result).

On the next step of building $E$, we define all its components $A[t]$ equal to the Turing machine $M_q$. So, the Turing machine $A[t]$ takes the transition output of the Turing machine $A[t - 1]$ and works with it until it produces its first output writing it in both output tapes and stopping functioning. In such a way, the evolutionary Turing machine $E$ simulates behavior of the inductive Turing machine $M$, while the string of outcomes of $E$ is exactly the same as the string of outcomes of $M$. It means that when the evolutionary Turing machine $E$ works in the inductive mode, its result coincides with the result of the inductive Turing machine $M$. By construction, $E$ is a basic periodic evolutionary Turing machine with the period 1.

Theorem is proved.

As simple inductive Turing machines are more powerful than Turing machines [7], basic periodic evolutionary Turing machines are also more powerful than Turing machines. Consequently, general periodic evolutionary Turing machines are as well more powerful than Turing machines.

At the same time, some classes of evolutionary **K**-machines are computationally, functionally or linguistically equivalent to the class **K** of abstract automata/machines. For instance, we have the following results.

**Theorem 2.2.** If the class **K** is closed with respect to the sequential composition, then any basic $n$-level evolutionary **K**-machine $F$ is functionally equivalent to an automaton from **K**.

*Proof.* We prove this statement by induction. When there is only one level, then any 1-level evolutionary **K**-machine is an automaton from **K**.

Let us assume that the result is proved for all $(n - 1)$-level evolutionary **K**-machines. Thus, any $n$-level evolutionary **K**-machine $F$ is functionally equivalent to a 2-level evolutionary **K**-machine $E = \{A[1], A[2]\}$. By Definition 2.5, $E$ works as the sequential composition of $A[1]$ and $A[2]$. By the assumption of the theorem, the class **K** is closed with respect to the sequential composition. Thus, the machine $E$ is functionally equivalent to an automaton from **K**. Consequently, the initial evolutionary **K**-machine $F$ is functionally equivalent to an automaton from **K**.

By the principle of induction, theorem is proved.

**Corollary 2.1.** If the class **K** is closed with respect to the sequential composition, then the class of all bounded basic evolutionary **K**-machines is functionally equivalent to the class **K**.

**Corollary 2.2.** Any basic $n$-level evolutionary finite automaton $A$ is functionally equivalent to a finite automaton.

**Corollary 2.3.** Any basic $n$-level evolutionary Turing machine $E$ is functionally equivalent to a Turing machine.

**Corollary 2.4.** If the class **BBEFA** of all bounded basic evolutionary finite automata is functionally equivalent to the class **FA** of all finite automata.

**Corollary 2.5.** If the class **BBETM** of all bounded basic evolutionary Turing machines is functionally equivalent to the class **T** of all Turing machines.

**Theorem 2.3.** If the class **K** is closed with respect to the sequential composition and automata from **K** can perform cycles, then any general $n$-level evolutionary **K**-machine $F$ is functionally equivalent to an automaton from **K**.

Proof is similar to the proof of Theorem 2.2.

**Corollary 2.6.** If the class **K** is closed with respect to the sequential composition and automata from **K** can perform cycles, then the class of all bounded general evolutionary **K**-machines is functionally equivalent to the class **K**.

**Corollary 2.7.** Any general $n$-level evolutionary Turing machine $E$ is functionally equivalent to a Turing machine.

**Corollary 2.8.** The class **BGETM** of all bounded general evolutionary Turing machines is functionally equivalent to the class **T** of all Turing machines.

These results show that in some cases, evolutionary computations do not add power to computing devices.

**Theorem 2.4.** If the class **K** is closed with respect to the sequential composition, then: any basic periodic evolutionary **K**-machine $F$ with the period $k > 1$ is functionally equivalent to a basic periodic evolutionary **K**-machine $E$ with the period 1; any basic almost periodic evolutionary **K**-machine $F$ with the period $k > 1$ is functionally equivalent to a basic almost periodic evolutionary **K**-machine $E$ with the period 1.

*Proof*. Let us consider an arbitrary basic periodic evolutionary **K**-machine $E = \{A[t]; t = 1, 2, 3, ...\}$. By Definition 2.5, the sequence $\{A[t]; t = 1, 2, 3, ...\}$ of evolutionary **K**-machines $A[t]$ is either finite or periodic, *i.e.*, there is a finite initial segment of this sequence such that the whole sequence is formed by infinite repetition of this segment. When the sequence $\{A[t]; t = 1, 2, 3, ...\}$ of automata $A[t]$ from **K** is finite, then by Theorem 2.1, the evolutionary **K**-machine $E$ is functionally equivalent to an automaton $A_E$ from **K**. By definition, $A_E$ is a basic periodic evolutionary **K**-machine with the period 1. Thus, in this case, theorem is proved.

Now let us assume that the sequence $\{A[t]; t = 0, 1, 2, 3, ...\}$ of automata $A[t]$ is infinite. In this case, there is a finite initial segment $H = \{A[t]; t = 0, 1, 2, 3, ..., n\}$ of this sequence such that the whole sequence is formed by infinite repetition of this segment $H$. By definition, $H$ is an $n$-level evolutionary **K**-machine. Then by Theorem 2.1, there is an automaton $A_H$ from **K** functionally equivalent to $H$. Thus, evolutionary **K**-machine $E$ is functionally equivalent to the periodic evolutionary **K**-machine $B = \{B[t]; t = 0, 1, 2, 3, ...\}$ of automata $B[t] = A_H$ for all $t = 0, 1, 2, 3, ....$ Thus, $B$ is a periodic evolutionary **K**-machine with the period 1.

Part (a) is proved.

Proof of part (b) is similar. Theorem is proved.

**Corollary 2.9.** If the class **K** is closed with respect to the sequential composition, then the class of all basic periodic evolutionary **K**-machines is functionally equivalent to the class of all basic periodic evolutionary **K**-machines with the period 1.

**Corollary 2.10.** Any basic (almost) periodic evolutionary finite automaton $A$ is functionally equivalent to a basic (almost) periodic evolutionary finite automaton with the period 1.

**Corollary 2.11.** Any basic (almost) periodic evolutionary Turing machine $E$ is functionally equivalent to a basic (almost) periodic evolutionary Turing machine $H$ with the period 1.

**Corollary 2.12.** The class **PBEFA** (**APBEFA**) of all basic (almost) periodic evolutionary finite automata is functionally equivalent to the class $\mathbf{PBEFA}_1$ ($\mathbf{APBEFA}_1$) of all basic (almost) periodic evolutionary finite automata with the period 1.

**Corollary 2.13.** The class **PBETM** (**APBETM**) of all basic (almost) periodic evolutionary Turing machines is functionally equivalent to the class $\mathbf{PBETM}_1$ ($\mathbf{APBETM}_1$) of all basic (almost) periodic evolutionary Turing machines with the period 1.

**Corollary 2.14.** Any basic (almost) periodic evolutionary Turing machine $G$ with the period $k > 1$ is functionally equivalent to a basic (almost) periodic evolutionary Turing machine $E$ with the period 1.

Now let us consider general evolutionary machines.

**Theorem 2.5.** If the class **K** is closed with respect to the sequential composition and automata from **K** can perform cycles, then any general periodic evolutionary **K**-machine $F$ with the period $k > 1$ is functionally equivalent to a periodic evolutionary **K**-machine $E$ with the period 1.

Proof is similar to the proof of Theorem 2.4.

**Corollary 2.15.** Any general periodic evolutionary finite automaton $E$ is equivalent to a one-dimensional cellular automaton.

Proof directly follows from Theorem 2.5 as any periodic evolutionary finite automaton with the period 1 is a cellular automaton.

One-dimensional cellular automata are functionally equivalent to Turing machines, while Turing machines are more powerful than finite automata [7]. Thus, Corollary 2.15 shows that periodic evolutionary finite automata are more powerful than finite automata.

**Corollary 2.16.** Any general (almost) periodic evolutionary Turing machine $G$ with the period $k > 1$ is functionally equivalent to a general (almost) periodic evolutionary Turing machine $E$ with the period 1.

Proof is similar to the proof of Theorem 2.1.

Classes **GEAK**, **BEAK** and some of their subclasses inherit many properties from the class **K**. Here are some of these properties.

Let **K** be a class of abstract automata/machines.

**Proposition 2.5.** If the class **K** has an identity automaton, *i.e.*, an automaton $E$ such that $E(x) = x$ for all $x$ from the language of **K**, then classes **GEAK**, **BEAK**, **BGEAK**, **BBEAK**, **PGEAK**, **PBEAK**, **RGEAK**, **RBEAK**, **IGEAK**, and **IBEAK** also have identity automata.

Indeed, any automaton from the class **K** is an evolutionary **K**-machine with one level and thus, the identity automaton belongs to all these classes of evolutionary machines.

**Proposition 2.6.** If the class **K** is closed with respect to the sequential composition, then the classes **BGEAK** of bounded general evolutionary **K**-machines and **BBEAK** of bounded basic evolutionary **K**-machines are also closed with respect to the sequential composition.

For general classes of automata/machines, it is possible to find much more tentatively inherited properties in [20].

For finding properties of evolutionary information, we need the following property of inductive Turing machines.

**Theorem 2.6.** The class of all multitape inductive Turing machines is closed with respect to sequential composition.

*Proof*. For simplicity, we show that there is an inductive Turing machine with five tapes that exactly simulates (computes the same function as) the sequential composition of two inductive Turing machines with three tapes. The general case is proved in a similar way when the number of tapes in the simulating inductive Turing machine is larger than the maximum of the numbers of tapes in both inductive Turing machines from the sequential composition.

Let us consider two inductive Turing machines $M$ and $Q$ each having three tapes: the input tape, working tape, and output tape. It is possible to assume that the inductive Turing machines $M$ and $Q$ never stop functioning given some input, producing the result if and only the words in the output tape stop changing after some step [7].

Their sequential composition $M \circ Q$ is defined in a natural way. When given an input $x$, the machine $M$ does not give the result, then $M \circ Q$ also does not give the result. When the machine $M$ gives the result $M(x)$, it goes as input to the machines $Q$, which starts computing with this input. When given the

input $M(x)$, the machines $Q$ does not give the result, then $M \circ Q$ also does not give the result. Otherwise, the result of $Q$ starting with the input $M(x)$ is the result of the sequential composition $M \circ Q$.

Now let us build the inductive Turing machine $W$ with the following properties. The machine $W$ contains submachines $M_0$ and $Q_0$ computationally equivalent to the inductive Turing machines $M$ and $Q$. It is possible to realize these by subprograms of the program of inductive Turing machine $W$ [7]. The input tape of the machine $W$ coincides with the input tape of the machine $M_0$. The output tape of the machine $W$ coincides with the output tape of the machine $Q_0$.

In addition, $W$ has a subprogram $C$ that organizes interaction of $M_0$ and $Q_0$ in the following way. Whenever the machine $M_0$ writes its new partial result in its output tape, the machine $C$ compares this output with the word written in its working tape. When both words coincide, the machine $C$ halts and the machine $Q_0$ makes its next step of computation and the machine $M_0$ makes its next step of computation. When the compared words are different, the machine $C$ rewrites the word from the output tape of $M_0$ into the working tape of $C$ and the input tape of $Q_0$. After this the machine $Q_0$ erases everything from its working and output tapes, writes 1 into its output tape, changes 1 to 0, erases 0 and makes first step of computation with the new input written by the machine $C$ in its input tape. Then the machine $M_0$ makes its next step of computation.

Now we can show that the inductive Turing machine $W$ exactly simulates—computes the same function as—the sequential composition $M \circ Q$ of the inductive Turing machines $M$ and $Q$. Indeed, if given an input $x$, the machine $M$ does not give the result, then the output of $M_0$ does not stop changing. Thus, by construction of $W$, the output of $W$ also does not stop changing. It means that $W$ does not give the result. When the machine $M$ gives the result $M(x)$, then its copy $M_0$ also produces the same result $M(x)$. It means that the output of $M_0$ stops changing after some step of computation. Consequently, all comparisons of the machine $C$ will give positive results and it will not interfere into the functioning of $Q_0$, which will perform all computations with the input $M(x)$. At the same time, $M(x)$ goes as input to the machines $Q$, which starts computing with this input. When given an input $M(x)$, the machine $Q$ does not give the result, then its output does not stop changing. Consequently the output of its copy $Q_0$ also does not stop changing and $Q_0$ does not give the result. Thus, by construction of $W$, the output of $W$ also does not stop changing. It means that $W$ does not give the result. Otherwise, the result of $Q_0$ starting with the input $M(x)$ is the result of the inductive Turing machine $W$. This shows that the inductive Turing machine $W$ functions exactly as the sequential composition $M \circ Q$, giving the same result.

Theorem is proved.

Theorems 2.6, 2.2 and 2.3 give the following result.

**Corollary 2.17.** Any basic (general) $n$-level evolutionary inductive Turing machine $E$ is functionally equivalent to an inductive Turing machine.

**Corollary 2.18.** Any basic (general) periodic evolutionary inductive Turing machine $G$ with the period $k > 1$ is functionally equivalent to a general periodic evolutionary inductive Turing machine $E$ with the period 1.

## 3. Universal Evolutionary Automata

Let **H** be a class of automata. The standard construction of universal automata and algorithms is usually based on some codification (symbolic description) **c**: **H** → $L$ of all automata/algorithms in **H**. Here $L$ is the language of the automata from **H**, *i.e.*, $L$ consists of words with which automata/algorithms from **H** work. Note that in a general case, these words can be infinite. It is useful to distinguish finitary languages, in which all words are finite, and infinitary languages, in which all words are infinite. In a natural way, a coding in a finitary language is called *finitary* and a coding in an infinitary language is called *infinitary*.

**Definition 3.1.** (a) An automaton/algorithm/machine $U$ is *universal for* the class **H** if given a description (coding) **c**($A$) of an automaton/algorithm $A$ from **H** and some input data $x$ for it, $U$ gives the same result as $A$ for the input $x$ or gives no result when $A$ gives no result for the input $x$; (b) An automaton/algorithm/machine $U$ is *universal in* the class **H** if it is universal for the class **H** and belongs to **H**.

For instance, a universal Turing machine $U$ is universal for the class **FA** of all finite automata but it is not universal in the class **FA** because $U$ does not belong to **FA**. Moreover, the class **FA** does not have automata universal in **FA** [7].

We define a universal evolutionary automaton/algorithm/machine as an automaton/algorithm/machine that can simulate all machines/automata/algorithms from **H** in a similar way, as a universal Turing machine has been defined as a machine that can simulate all possible Turing machines.

The pair (**c**($A$), $x$) belongs to the direct product $X^* \times X^*$ where $X^*$ is the set of all words in an alphabet $X$ of automata/machines from **H**. To be able to process this pair by automata from **H**, we need a coding of pairs of words by single words. To have necessary properties of the coding, we consider three mappings $<\cdot,\cdot>$: $X^* \times X^* \to X^*$, $\lambda$: $X^* \to X^*$ and $\pi$: $X^* \to X^*$ such that for any words $u$ and $v$ from $X^*$, we have $\lambda(<u, v>) = u$ and $\pi(<u, v>) = v$. The standard coding of pairs of words [6] has the following form

$$<u, v> = 1^{l(u)}0uv$$

where $l(v)$ is the length of the word $v$ and $1^n$ denotes the sequence of $n$ symbols 1. Often we will need the following property of the coding $<\cdot,\cdot>$.

**Condition L.** For any word $u$ from $X^*$, there is a number $c_u$ such that for any word $v$ from $X^*$, we have

$$l(<u, v>) \leq l(v) + c_u$$

For the standard coding $<\cdot,\cdot>$, the constant $c_u$ is equal to $2l(u) + 1$.

In this context, an automaton/algorithm/machine $U$ is *universal for* the class **H** if for any automaton/algorithm $A$ from **H** and some input data $x$ for it, $U(<\mathbf{c}(A), x>) = A(x)$ if $A(x)$ is defined and $U(<\mathbf{c}(A), x>)$ is undefined when $A(x)$ is undefined. This type of the representation of the pair (**c**($A$), $x$) is called the *direct pairing*.

It is also possible to give a dual definition: an automaton/algorithm/machine $U$ is *universal for* the class **H** if for any automaton/algorithm $A$ from **H** and some input data $x$ for it, $U(<x, \mathbf{c}(A)>) = A(x)$ if $A(x)$ is defined and $U(<x, \mathbf{c}(A)>)$ is undefined when $A(x)$ is undefined. This type of the representation of the pair (**c**($A$), $x$) is called the *dual pairing*.

**Definition 3.2.** A coding **c** used in a universal evolutionary automaton/algorithm/machine for simulation is called a *simulation coding*.

Note that existence of a universal automaton in the class **K** implies existence of a simulation coding **k**: **K** → $X^*$ where $X$ is the alphabet of the automata from **K** and $X^*$ is the set of all words in the alphabet $X$.

**Example 3.1.** A *basic universal evolutionary Turing machine* (BUETM) is a BETM $EU$ with the optimization space $Z = I \times X$ that has the following properties. Given a pair (**c**($E$), $X[0]$) where $E = \{\mathrm{TM}[t]; t = 0, 1, 2, 3, ...\}$ is a BETM and $X[0]$ is the start population, the machine $EU$ takes this pair as its input and produces the same population $X[1]$ as the Turing machine TM[0] working with the same population $X[0]$ [2]. Then $EU$ takes the pair (**c**($E$), $X[1]$) as its input and produces the same population $X[2]$ as the Turing machine TM[1] working with the population $X[1]$. In general, $EU$ takes the pair (**c**($E$), $X[t]$) as its input and produces the same population $X[t + 1]$ as the Turing machine TM[$t$] working with the population $X[t]$ where $t = 0, 1, 2, 3, ....$

In other words, a basic universal evolutionary Turing machine can simulate behavior of an arbitrary BETM $E$. This is similar to the concept of a universal Turing machine as a Turing machine that can simulate all possible Turing machines that work with words in a given alphabet.

**Example 3.2.** A *basic universal evolutionary inductive Turing machine* (BUEITM) is a BEITM $EU$ with the optimization space $Z = I \times X$ that has the following properties. Given a pair (**c**($E$), $X[0]$) where $E = \{\mathrm{ITM}[t]; t = 0, 1, 2, 3, ...\}$ is a BEITM and $X[0]$ is the start population, the machine $EU$ takes this pair as its input and produces the same population $X[1]$ as the inductive Turing machine ITM[0] working with the same population $X[0]$ (cf., [2]). Then $EU$ takes the pair (**c**($E$), $X[1]$) as its input and produces the same population $X[2]$ as the inductive Turing machine ITM[1] working with the population $X[1]$. In general, $EU$ takes the pair (**c**($E$), $X[t]$) as its input and produces the same population $X[t + 1]$ as the inductive Turing machine ITM[$t$] working with the population $X[t]$ where $t = 0, 1, 2, 3, ....$

In other words, a basic universal evolutionary inductive Turing machine can simulate behavior of an arbitrary BEITM $E$.

**Proposition 3.1.** If the language $L$ of automata from **K** is finitary and there is a coding **k** of **K** in a language $L$, then there is a coding **c** of all general (basic) evolutionary **K**-machines in an extension of the language $L$.

Indeed, it is possible to construct the code of an evolutionary **K**-machine $E$ as the concatenation of the codes of all its components, *i.e.*, taking a coding **k** of **K** in $L$ and an evolutionary **K**-machine $E = \{A[t]; t = 1, 2, 3, ...\}$, we can use the sequence $\mathbf{c}(E) = \mathbf{k}(A[1]) \circ \mathbf{k}(A[2]) \circ \mathbf{k}(A[3]) \circ ... \circ \mathbf{k}(A[n]) \circ \ldots$ where the symbol $\circ$ does belong to the language $L$ as a code of the evolutionary **K**-machine $E$.

Note that the code **c**($E$) of an evolutionary Turing machine $E$ can be infinite in a general case. However, there are many reasons to consider algorithms that work with infinite objects. There are abstract automata (machines) that work with infinite words [21] or with such infinite objects as real

numbers [22]. The construction of evolutionary Turing machines, evolutionary inductive Turing machines, and evolutionary limit Turing machines allows these machines to work with inputs in the form of infinite words.

Definition 3.1 gives properties of universal evolutionary **K**-machines but does not imply existence of such machines. Thus, we need the following result to determine existence of universal evolutionary **K**-machines.

**Theorem 3.1.** (a) If the class **K** has universal automata in **K** and a finitary coding, then the class **GEAK** also has universal general evolutionary automata in **GEAK**; (b) If the class **K** has universal automata for **K** and a finitary coding, then the class **GEAK** also has universal general evolutionary automata for **GEAK**.

*Proof.* (a) To build a universal evolutionary **K**-machine, we use the structure of a universal automaton (machine) in **K**, which exists according to our assumption. Note that existence of a universal automaton in **K** implies existence of a coding $\mathbf{k}$: $\mathbf{K} \to X^*$ where $X$ is the alphabet of the automata from **K** and $X^*$ is the set of all words in the alphabet $X$.

A universal evolutionary **K**-machine $U$ is constructed in a form of the series $U = \{U[t]; t = 1, 2, 3, ...\}$ of the instances $U[t]$ of a universal in **K** automaton $V$, *i.e.*, $U[t]$ is a copy of $V$, working on pairs ( $\mathbf{k}(A[t])$, $X[t]$) in generations $t = 0, 1, 2, 3, ....$ Each automaton $U[t]$ has one input channel for receiving its input and two output channels. One is called the transfer channel and used for transferring data (the generation $X[i + 1]$) either to $U[t + 1]$ or to $U[t - 1]$. The second channel called the outcome channel is used for producing the outcome of the automaton $U[t]$ when he starts working with the input $X[i]$.

Note that although components of the evolutionary **K**-machine $U$ perform multiple computations [18], it is possible to code each generation $X[i]$ by a single word. This allows us to use machines/automata that work with words for building evolutionary machines/automata.

For simulation by $U$, a coding $\mathbf{k}$ of the machines/automata in **K** is used and an evolutionary **K**-machine $E = \{A[t]; t = 1, 2, 3, ...\}$ is coded by the sequence $\mathbf{c}(E) = \mathbf{k}(A[1]) \circ \mathbf{k}(A[2]) \circ \mathbf{k}(A[3]) \circ ... \circ \mathbf{k}(A[n]) \circ \ldots$ where the symbol $\circ$ does belong to the language $L$ as a code of the evolutionary **K**-machine $E$. Then to simulate $E$, the initial generation $X[0]$ and the sequence $\mathbf{c}(E)$ go as the input to the first component $U[1]$ of the evolutionary **K**-machine $U$. The first component $U[1]$ uses $X[0]$ and the first part $\mathbf{k}(A[1])$ from the sequence $\mathbf{c}(E)$ to produce the next generation $X[1]$ and its outcome $O[1]$. Then $X[1]$ and the second part $\mathbf{k}(A[2])$ from the sequence $\mathbf{c}(E)$ go to the next component $U[2]$ of the evolutionary **K**-machine $U$. As $V$ is a universal automaton in **K**, the evolutionary **K**-machine $U$ exactly simulates the evolutionary **K**-machine $E$.

Consequently, the evolutionary **K**-machine $U$ is universal in the class of all general evolutionary **K**-machines because given the input ($\mathbf{c}(M[t])$, $X[t]$), each $U$[t] simulates the corresponding automaton $M[t]$ when it works with input $X[t]$. Part (a) is proved.

(b) Taking a universal automaton (machine) $V$ for **K**, which exists according to our assumption, we construct a universal for **GEAK** evolutionary machine in a form of the series $U = \{U[t]; t = 0, 1, 2, 3, ...\}$ of the instances of a universal in **K** automaton $U$, *i.e.*, $U[t]$; is a copy of $U$, working on pairs ($\mathbf{c}(M[t])$, $X[t]$) in generations $t = 0, 1, 2, 3, ....$ The only difference from basic universal evolutionary machines is that instead of the alphabet $X$ of the automata from **K**, we use the alphabet $Z$ of the automaton $U$ for coding $\mathbf{c}$ of the automata from **K**.

Theorem is proved.

As there are universal Turing machines, Theorem 3.1 implies the following result.

**Corollary 3.1.** The class **GETM** of all general evolutionary Turing machines has a general universal evolutionary inductive Turing machine.

As there are universal inductive Turing machines of the first order [7], Theorem 3.1 implies the following result.

**Corollary 3.2**. The class $\mathbf{GEITM_1}$ of all general evolutionary inductive Turing machines of the first order has a general universal evolutionary inductive Turing machine.

As there are universal inductive Turing machines of order $n$ [7], Theorem 3.1 implies the following result.

**Corollary 3.3** [2]. The class $\mathbf{GEITM}_n$ of all general evolutionary inductive Turing machines of order $n$ has a general universal evolutionary inductive Turing machine.

As there are universal limit Turing machines of order $n$ [23], Theorem 3.1 implies the following result.

**Corollary 3.4**. The class $\mathbf{GELTM}_n$ of all general evolutionary limit Turing machines of order $n$ has a general universal evolutionary limit Turing machine.

Theorem 3.1 and Proposition 3.1 also imply the following result.

**Proposition 3.2.** If the automata from **K** have a finitary simulation coding, then all general almost periodic evolutionary **K**-machines, all general periodic evolutionary **K**-machines and all general finite evolutionary **K**-machines have a finitary simulation coding.

*Proof*. Let us consider a finitary coding $\mathbf{c}$: $\mathbf{K} \to X^*$. Then taking a periodic evolutionary **K**-machine $E = \{A[t];\ t = 1, 2, 3, ...\}$ and its period $\{A[1], A[2], A[3], ..., A[n]\}$, we can use the sequence $\mathbf{c}(A[1]) \circ \mathbf{c}(A[2]) \circ \mathbf{c}(A[3]) \circ ... \circ \mathbf{c}(A[n])$ where the symbol $\circ$ does belong to the language $L = X^*$ as a (simulation) code of the evolutionary **K**-machine $E$ because this machine is generated by repeating its period infinitely many times.

Taking a bounded evolutionary **K**-machine $E = \{A[t];\ t = 0, 1, 2, 3, ..., n\}$, we can use the sequence $\mathbf{c}(A[1]) \circ \mathbf{c}(A[2]) \circ \mathbf{c}(A[3]) \circ ... \circ \mathbf{c}(A[n])$ where the symbol $\circ$ does belong to $L = X^*$ as a (simulation) code of the evolutionary **K**-machine $E$.

In the case of an almost periodic evolutionary **K**-machine $E = \{A[t];\ t = 1, 2, 3, ...\}$, we combine codes of its bounded initial segment and its period, separating them by a word from $X^*$ that is not used for coding and is not a part of any code.

Proposition is proved

**Proposition 3.3.** If the language $L$ of automata from **K** is finitary and there is a coding **k** of **K** in $L$, then there is a finitary coding of all basic periodic evolutionary **K**-machines, of all basic almost periodic evolutionary **K**-machines and of all basic bounded evolutionary **K**-machines.

Proof is similar to the proof of Proposition 3.2.

**Theorem 3.2.** (a) If the class **K** has universal automata in **K** and a finitary coding, then the class **BEAK** also has a universal basic evolutionary automata in **BEAK**; (b) If the class **K** has universal automata for **K** and a finitary coding, then the class **BEAK** also has a universal basic evolutionary automata for **BEAK**.

Proof is similar to the proof of Theorem 3.1.

As there are universal Turing machines, Theorem 3.1 implies the following result.

**Corollary 3.5** [2]. The class **BETM** of all basic evolutionary Turing machines has a basic universal evolutionary inductive Turing machine.

As there are universal inductive Turing machines of the first order [7], Theorem 3.1 implies the following result.

**Corollary 3.6** [2]. The class $\mathbf{BEITM}_1$ of all basic evolutionary inductive Turing machines of the first order has a basic universal evolutionary inductive Turing machine.

As there are universal inductive Turing machines of order $n$ [7], Theorem 3.1 implies the following result.

**Corollary 3.7** [2]. The class $\mathbf{BEITM}_n$ of all basic evolutionary inductive Turing machines of order $n$ has a basic universal evolutionary inductive Turing machine.

As there are universal limit Turing machines of order $n$ [23], Theorem 3.1 implies the following result.

**Corollary 3.8**. The class $\mathbf{BELTM}_n$ of all basic evolutionary limit Turing machines of order $n$ has a basic universal evolutionary limit Turing machine.

Theorem 3.2 and Proposition 3.1 also imply the following result.

By construction, the universal general evolutionary automaton in **BEAK** is periodic with the period 1. Thus, we have the following results.

**Theorem 3.3.** (a) If the class **K** has universal automata in **K** and a finitary coding, then the class **PGEAK** of all periodic general basic evolutionary automata in **GEAK** also has a universal periodic general evolutionary automata in **PGEAK**; (b) If the class **K** has universal automata in **K** and a finitary coding, then the class $\mathbf{PGEAK}_1$ of all periodic general evolutionary automata in **GEAK** with the period 1 also has a universal periodic general evolutionary automata in $\mathbf{PGEAK}_1$ .

The same is true for basic evolutionary automata.

**Theorem 3.4.** (a) If the class **K** has universal automata in **K** and a finitary coding, then the class **PBEAK** of all periodic basic evolutionary automata in **BEAK** also has a universal periodic basic evolutionary automata in **PBEAK**; (b) If the class **K** has universal automata in **K** and a finitary coding,

then the class $\mathbf{PBEAK}_1$ of all periodic basic evolutionary automata in **BEAK** with the period 1 also has a universal periodic basic evolutionary automata in $\mathbf{PBEAK}_1$ .

The situation with universal machines in arbitrary classes of bounded evolutionary automata is more complicated. Some of these classes have universal machines and some do not have.

**Theorem 3.5.** (a) If the class **K** is closed with respect to the sequential composition, has universal automata in **K**, a finitary coding and automata from **K** can perform cycles, then the class **BGEAK** of all bounded general evolutionary automata in **GEAK** also has universal bounded general evolutionary automata in **BGEAK**; (b) If the class **K** has universal automata in **K**, a finitary coding and is closed with respect to the sequential composition, then the class **BBEAK** of all bounded basic evolutionary automata in **BEAK** also has universal bounded basic evolutionary automata in **BBEAK**.

Indeed, by Theorem 2.3, any general bounded evolutionary **K**-machine is functionally equivalent to an automaton from **K**. Thus, a universal automaton $V$ from **K**, can simulate any general bounded evolutionary **K**-machine. At the same time, any automaton from **K** can be treated as a general bounded evolutionary **K**-machine.

In a similar way, by Theorem 2.2, any basic bounded evolutionary **K**-machine is functionally equivalent to an automaton from **K**. Thus, a universal automaton $V$ from **K**, can simulate any basic bounded evolutionary **K**-machine. At the same time, any automaton from **K** can be treated as a basic bounded evolutionary **K**-machine.

As the following example demonstrates, the condition of containing all sequential compositions is essential for Theorem 3.5.

**Example 3.3.** Let us build a finite automaton $A$ such that given a word $w = a_1a_2a_3 \ldots a_n$, it gives the word $u = a_1a_2a_3 \ldots a_{n-1}$ as its output. We define $A = (\Sigma, Q, (q_0, \varepsilon), F, R)$ where:

$\Sigma = \{0, 1\}$ is the alphabet of $A$,
$Q = \{(q_0 , \varepsilon), (q_0, \varepsilon), (q_0, \varepsilon)\}$ is the set of states of $A$,
$(q_0, \varepsilon)$ is the start state,
$F = Q$ is the set of states of $A$,
and the set $R$ of rules of $A$ consists of the following rules:

$$\varepsilon, (q_0, \varepsilon) \to (q_0, \varepsilon), \varepsilon;$$
$$0, (q_0, \varepsilon) \to (q_0, 0), \varepsilon;$$
$$1, (q_0, \varepsilon) \to (q_0, 1), \varepsilon;$$
$$0, (q_0, 0) \to (q_0, 0), 0;$$
$$1, (q_0, 0) \to (q_0, 1), 0;$$
$$0, (q_0, 1) \to (q_0, 0), 1;$$
$$1, (q_0, 1) \to (q_0, 1), 1;$$
$$\varepsilon, (q_0, 1) \to (q_0, \varepsilon), \varepsilon;$$
$$\varepsilon, (q_0, 0) \to (q_0, \varepsilon), \varepsilon;$$

The expression $\boldsymbol{a}$, $(\boldsymbol{q_0}, \boldsymbol{b}) \rightarrow (\boldsymbol{q_0}, \boldsymbol{d})$, $\boldsymbol{c}$ means that given an input $a$ and being in the state $(q_0, b)$, the automaton $A$ given $c$ as its transition output and its outcome and make the transition to the state $(q_0, d)$.

$A$ is a deterministic automaton. By definition, it is universal in the class $\mathbf{K}_A = \{A\}$ that consists only of the automaton $A$.

Let us consider bounded basic evolutionary $\mathbf{K}_A$-machines. They all have the form $E = \{A[t]; t = 1, 2, 3, ..., n\}$ where all $A[t]$ are copies of the automaton $A$. Such an evolutionary $\mathbf{K}_A$-machine $E$ works in the following way. Receiving a word $w = a_1a_2a_3 \ldots a_n$ as its input, the automaton $A[1]$ produces the word $u = a_1a_2a_3 \ldots a_{n-1}$ as its transition output and the empty word ε as its outcome. Then when $n > 2$, the automaton $A[2]$ receives the word $u = a_1a_2a_3\ldots a_{n-1}$ as its input and produces the word $v = a_1a_2a_3\ldots a_{n-2}$ as its transition output and the empty word ε as its outcome. This process continues until the automaton $A[n]$ receives the word $a_1$ as its input. Then $A[n]$ does not produce the transition output (it is the empty word ε), while its outcome is equal to the word $a_1$.

Any evolutionary machine in the class $\mathbf{BBEAK}_A$ of all bounded basic evolutionary $\mathbf{K}_A$-machines has the form $E = \{A[t]; t = 1, 2, 3, ..., n\}$ where all $A[t]$ are copies of the automaton $A$. However, such a machine cannot be universal in the class $\mathbf{BBEAK}_A$ because it cannot simulate the machine $E_{n+1} = \{A[t]; t = 1, 2, 3, ..., n, n + 1\}$ where all $A[t]$ are copies of the automaton $A$. Indeed, on the input word $r = a_1a_2a_3\ldots a_na_{n+1}$ and on any longer input word, the machine $E$ gives the empty outcome, while the machine $E$ gives the outcome $a_1$ on the input word $r$. It means that the class $\mathbf{BBEAK}_A$ does not have universal automata although $\mathbf{K}_A$ has universal automata.

## 4. Evolutionary Information Size

Here we introduce and study evolutionary information necessary to develop a constructive object by a given system of evolutionary algorithms (evolutionary automata/machines). It is called *evolutionary* or *genetic information about* an object or the *evolutionary information size* of an object. Here we consider symbolic representation of objects. Namely, it is assumed that all automata/machines work with strings of symbols (words) and all objects are such strings of symbols (words).

Let us consider a class **H** of evolutionary automata/machines. Note that that the initial population $X[0]$ is coded as one word in the alphabet of evolutionary automata/machines from **H** and its is possible to consider this input as a representation $p$ of the genetic information used to produce the output population (object) $x$ that satisfies the search condition. Evolutionary automata/machines from **H** represent the environment in which evolution goes.

**Definition 4.1.** The *quantity* $\mathrm{EIS}_A(x)$ *of evolutionary information about* an object/word $x$, also called the *evolutionary information size* $\mathrm{EIS}_A(x)$ of an object/word $x$ with respect to an automaton/machine $A$ from **H** is defined as

$$\mathrm{EIS}_A(x) = \min \{l(p); A(p) = x\}$$

where $l(p)$ is the length of the word $p$, while in the case when there are no words $p$ such that $A(p, y) = x$, $\mathrm{EIS}_A(x)$ is not defined.

Informally, the quantity of evolutionary information about an object $x$ with respect to an automaton $A$ shows how much information it is necessary for building (computing or constructing) this object by

means of the automaton $A$. Thus, it is natural that different automata need different quantity of evolutionary information about an object $x$ to build (compute or construct) this object.

However, people, computers and social organizations use systems (classes) of automata/algorithms and not a single automaton/algorithm. To define the quantity of information about an object $x$ with respect to a class of automata/algorithms, algorithmic information theory suggests taking universal automata/algorithms as representatives of the considered class. Such choice is grounded because it is proved that so defined quantity of information about an object is additively invariant with respect to the choice of the universal automaton/algorithm and is also additively optimal [5,8]. Similar approach works in the case of evolutionary information.

Taking a universal automaton $U$ in the class **H**, we can define relatively invariant optimal evolutionary information about an object in the class **H**.

**Definition 4.2.** The *quantity of evolutionary information about* an object/word $x$, also called the *evolutionary information size*, $\mathrm{EIS}_{\mathbf{H}}(x)$ of an object/word $x$ with respect to the class **H** is defined as

$$\mathrm{EIS}_{\mathbf{H}}(x) = \min \{l(p);\ U(p) = x\}$$

while in the case when there are no words $p$ such that $U(p) = x$, $\mathrm{EIS}_{\mathbf{H}}(x)$ is not defined.

Note that if there are no words $p$ such that $U(p) = x$, then for any automaton $A$ from **H**, there are no words $p$ such that $A(p) = x$. In particular, the function $\mathrm{EIS}_{\mathbf{H}}(x)$ is defined or undefined independently of the choice of universal algorithms for its definition.

Informally, quantity of evolutionary information about an object $x$ with respect to the class **H** shows how much information it is necessary for building (computing or constructing) this object by means of some automaton $U$ universal in the class **H**.

Note that when **H** consists of a single automaton/machine $A$, functions $\mathrm{EIS}_{\mathbf{H}}(x)$ and $\mathrm{EIS}_A(x)$ coincide.

**Example 4.1.** Recursive evolutionary information size of (evolutionary information about) an object $x$ is defined as $\mathrm{EIS}_{\mathbf{H}}(x)$ when **H** is a class of evolutionary Turing machines.

**Example 4.2.** Inductive evolutionary information size of (evolutionary information about) an object $x$ is defined as $\mathrm{EIS}_{\mathbf{H}}(x)$ when **H** is a class of evolutionary inductive Turing machines.

It is possible to show that evolutionary information size (evolutionary information about an object $x$) with respect to the class **H** is additively optimal when the automata from **H** have finite simulation coding.

**Theorem 4.1.** If the class **H** has a finitary coding and the coding simulation $<\cdot,\cdot>$ satisfies Condition L, then for any evolutionary automaton/machine $A$ from the class **H** and any universal evolutionary automaton $U$ in **H**, there is a constant number $c_{AU}$ such that for any object/word $x$, we have

$$\mathrm{EIS}_U(x) \le \mathrm{EIS}_A(x; y) + c_{AU} \tag{1}$$

*i.e.*, $\mathrm{EIS}_{\mathbf{H}}(x) = \mathrm{EIS}_U(x)$ is additively optimal.

*Proof*. Let us take an evolutionary automaton/machine $A$ from the class **H** and a word $x$. Then by Definitions 3.1 and 4.1,

$$\mathrm{EIS}_A(x) = \min \{l(q); A(q) = x\} = l(q_0)$$

and

$$\mathrm{EIS}_U(x) = \min \{l(p); U(<\mathbf{c}(A), p>) = x\} = l(p_0)$$

where <> is a coding of pairs of words. By Condition L, there is a number $c_v$ such that for any word $u$ from $X^*$, we have

$$l(<\mathbf{c}(A), p_0>) \leq l(p_0) + c_{\mathbf{c}(A)}$$

Then as $U(<\mathbf{c}(A), p_0>) = x$, we have

$$\mathrm{EIS}_U(x) = l(q_0) \leq l(<\mathbf{c}(A), p_0>) = l(p_0) + c_{\mathbf{c}(A)}$$

Thus,

$$l(q_0) \leq l(p_0) + c_{\mathbf{c}(A)}$$

and

$$\mathrm{EIS}_U(x) \leq \mathrm{EIS}_A(x) + c_{AU}$$

where $c_{AU} = c_{\mathbf{c}(A)}$. For the standard coding $<\cdot,\cdot>$, $c_{\mathbf{c}(A)} = 2l(\mathbf{c}(A)) + 1$. It means that $\mathrm{EIS}_{\mathbf{H}}(x)$ is additively optimal.

Theorem is proved.

Inequality (1) defines the relation $\preccurlyeq$ on functions, which is called the *additive order* [6–8], namely, $f(x) \preccurlyeq g(x)$ if there is a constant number $c$ such that for any $x$, we have

$$f(x) \leq g(x) + c \tag{2}$$

**Proposition 4.1.** The relation $\preccurlyeq$ is a partial preorder, *i.e.*, it is reflexive and transitive.

Indeed, for any function $f(x)$, we have $f(x) \leq f(x) + 0$, *i.e.*, the relation $\preccurlyeq$ is reflexive.

Besides, if $f(x) \leq g(x) + c$ and $g(x) \leq h(x) + k$, then we have $f(x) \leq h(x) + (c + k)$. It means that the relation $\preccurlyeq$ is transitive.

Thus, Theorem 4.1 means that information size with respect to a universal automaton/machine in **H** is additively minimal in the class of information sizes with respect to automata/machines in **H**.

Theorems 3.3, 3.4, 4.1 and Proposition 3.3 imply the following result.

**Corollary 4.1.** If the class **K** has universal automata in **K** and a finitary simulation coding, then evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to a universal evolutionary automaton/machine $EU$ in the class **PGEAK** (**PBEAK**) of all periodic general (basic) evolutionary **K**-automata/**K**-machines is additively minimal in the class of evolutionary information sizes with respect to evolutionary **K**-automata/**K**-machines in **PGEAK** (**PBEAK**).

The evolutionary information size $\mathrm{EIS}_{\mathbf{PGEAK}}(x)$ with respect to the class **PGEAK** is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal in **PGEAK** evolutionary automaton/machine *EU*.

The evolutionary information size $\mathrm{EIS}_{\mathbf{PBEAK}}(x)$ with respect to the class **PBEAK** is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal in **PBEAK** evolutionary automaton/machine *EU*.

As the class of all Turing machines has universal Turing machines and a finitary simulation coding, Corollary 4.1 implies the following result.

**Corollary 4.2.** There is an additively minimal evolutionary information size in the class of evolutionary information sizes with respect to periodic general (basic) evolutionary Turing machines.

The evolutionary information size $\mathrm{EIS}_{\mathbf{PGETM}}(x)$ with respect to the class **PGETM** of all periodic general evolutionary Turing machines is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal periodic general evolutionary Turing machine *EU*.

The evolutionary information size $\mathrm{EIS}_{\mathbf{PBETM}}(x)$ with respect to the class **PBETM** of all periodic basic evolutionary Turing machines is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal periodic basic evolutionary Turing machine *EU*.

As the class of all inductive Turing machines has universal inductive Turing machines and a finitary simulation coding [7], Corollary 4.1 implies the following result.

**Corollary 4.3.** There is an additively minimal evolutionary information size in the class of evolutionary information sizes with respect to periodic general (basic) evolutionary inductive Turing machines.

The evolutionary information size $\mathrm{EIS}_{\mathbf{PGEITM}}(x)$ with respect to the class **PGEITM** of all periodic general evolutionary inductive Turing machines is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal periodic general evolutionary inductive Turing machine *EU*.

The evolutionary information size $\mathrm{EIS}_{\mathbf{PBEITM}}(x)$ with respect to the class **PBEITM** of all periodic basic evolutionary inductive Turing machines is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal periodic basic evolutionary inductive Turing machine *EU*.

Theorem 4.1 and Proposition 3.3 imply the following result.

**Corollary 4.4.** If the class **K** has universal automata in **K**, is closed with respect to the sequential composition and has a finitary simulation coding, then evolutionary information size with respect to a universal evolutionary **K**-automaton/**K**-machine in the class **BGEAK** (**BBEAK**) of all bounded general (basic) evolutionary **K**-automata/ **K**-machines is additively minimal in the class of evolutionary information sizes with respect to evolutionary **K**-automata/**K**-machines in **BGEAK** (**BBEAK**).

The evolutionary information size $\mathrm{EIS}_{\mathbf{BGEAK}}(x)$ with respect to the class **BGEAK** is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal evolutionary automaton/machine *EU* the class **BGEAK**.

The evolutionary information size $\mathrm{EIS}_{\mathbf{BBEAK}}(x)$ with respect to the class **BBEAK** is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal evolutionary automaton/machine *EU* the class **BBEAK**.

As the class of all Turing machines has universal Turing machines, is closed with respect to the sequential composition and has a finitary simulation coding, Corollary 4.4 implies the following result.

**Corollary 4.5.** There is an additively minimal evolutionary information size in the class of evolutionary information sizes with respect to bounded general (basic) evolutionary Turing machines.

The evolutionary information size $\mathrm{EIS}_{\mathbf{BGEITM}}(x)$ with respect to the class **BGEITM** of all bounded general evolutionary inductive Turing machines is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal bounded general evolutionary inductive Turing machine *EU*.

The evolutionary information size $\mathrm{EIS}_{\mathbf{BBEITM}}(x)$ with respect to the class **BBEITM** of all bounded basic evolutionary inductive Turing machines is defined as evolutionary information size $\mathrm{EIS}_{EU}(x)$ with respect to with respect to some universal bounded basic evolutionary inductive Turing machine *EU*.

As the class of all multitape inductive Turing machines has universal inductive Turing machines, is closed with respect to the sequential composition (Theorem 2.3) and has a finitary simulation coding [7], Corollary 4.4 implies the following result.

**Corollary 4.6.** There is an additively minimal evolutionary information size in the class of evolutionary information sizes with respect to bounded general (basic) evolutionary multitape inductive Turing machines.

There are specific relations between information sizes relative to different classes of automata/algorithms.

**Proposition 4.2.** If $\mathbf{Q} \subseteq \mathbf{H}$, then $\mathrm{EIS}_{\mathbf{H}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{Q}}(x)$.

As it is possible to consider any automaton from **K** as a bounded general or basic evolutionary **K**-machine, Proposition 4.2 implies the following results.

**Corollary 4.7.** $\mathrm{EIS}_{\mathbf{K}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{BBETM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{PBETM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{APBETM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{APBEITM}}(x)$ and $\mathrm{EIS}_{\mathbf{BBETM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{BBEITM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{PBEITM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{APBEITM}}(x)$.

**Corollary 4.8.** $\mathrm{EIS}_{\mathbf{K}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{BGETM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{PGETM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{APGETM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{APGEITM}}(x)$ and $\mathrm{EIS}_{\mathbf{BGETM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{BGEITM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{PGEITM}}(x) \preccurlyeq \mathrm{EIS}_{\mathbf{APGEITM}}(x)$.

Inequality $\preccurlyeq$ defines the relation $\asymp$ on functions, which is called the *additive equivalence*, namely,

$$f(x) \asymp g(x) \text{ if } f(x) \preccurlyeq g(x) \text{ and } g(x) \preccurlyeq f(x)$$

**Proposition 4.3.** The relation $\asymp$ is an equivalence relation.

Indeed, as the relation $\preccurlyeq$ is reflexive and transitive, the relation $\asymp$ is also reflexive and transitive. In addition, by definition, the relation $\asymp$ is symmetric, *i.e.*, it is an equivalence relation.

**Corollary 4.9.** If the class **H** has a finitary coding, then for any two universal automata $V$ and $U$ in **H**, there is a constant number $k_{UV}$ such that for any object/word $x$ we have

$$\mathrm{EIS}_U(x) \leq \mathrm{EIS}_V(x) + c_{UV}$$

This shows that evolutionary information size (evolutionary information about an object $x$) with respect to the class **H** is additively invariant when the automata from **H** have finite simulation coding. Namely, we have the following result.

**Theorem 4.2.** If the class **H** has a finitary coding, then for any two universal automata $V$ and $U$ in **H**, then evolutionary information sizes $\mathrm{EIS}_U(x)$ and $\mathrm{EIS}_V(x)$ are additively equivalent, *i.e.*, $\mathrm{EIS}_U(x) \asymp \mathrm{EIS}_V(x)$.

*Proof*. As $V$ and $U$ are both universal automata in **H**, by Theorem 4.1, for all words $x$, we have

$$\mathrm{EIS}_U(x) \leq \mathrm{EIS}_V(x) + c_{UV}$$

and

$$\mathrm{EIS}_V(x) \leq \mathrm{EIS}_U(x) + c_{VU}$$

Thus,

$$\mathrm{EIS}_U(x; y) \geq \mathrm{EIS}_V(x; y) - c_{V\mathrm{y}}$$

It means that the quantities $\mathrm{EIQ}_U(y;\ x)$ and $\mathrm{EIQ}_V(y;\ x)$ of information in an object $y$ are additively equivalent.

Theorem is proved.

Definition 4.2 and properties of universal evolutionary algorithms imply the following result.

**Proposition 4.4.** If the class **H** has universal automata and an identity automaton, *i.e.*, an automaton $H$ such that $H(x) = x$ for any word $x$ in the language of the automata from **H**, then $\mathrm{EIS}_{\mathbf{H}}(x)$ is a total function on the set of all words the alphabet of the automata/machines from **H**.

**Lemma 4.1.** If the class **K** has an identity automaton, then all classes **GEAK**, B**GEAK**, P**GEAK**, **RGEAK**, **BEAK**, **BBEAK**, **PBEAK** and **RBEAK**, have an identity automaton.

Lemma 4.1 and Proposition 4.4 show that evolutionary information size with respect to each of the classes **GEAK**, B**GEAK**, P**GEAK**, **RGEAK**, **BEAK**, **BBEAK**, **PBEAK** and **RBEAK** is a total function.

**Corollary 4.10.** Evolutionary information size $\mathrm{EIS}_{\mathbf{BGETM}}(x)$ is a total function with respect to the class **BGETM** of all bounded general evolutionary Turing machines is a total function.

**Corollary 4.11.** Evolutionary information size $\mathrm{EIS}_{\mathbf{BGEITM}}(x)$ is a total function with respect to the class **BGEITM** of all bounded general evolutionary inductive Turing machines is a total function.

**Corollary 4.12.** Evolutionary information size $\mathrm{EIS}_{\mathbf{PGETM}}(x)$ is a total function with respect to the class **PGEITM** of all periodic general evolutionary Turing machines is a total function.

**Corollary 4.13.** Evolutionary information size $\mathrm{EIS}_{\mathbf{PGEITM}}(x)$ is a total function with respect to the class **PGEITM** of all periodic general evolutionary inductive Turing machines is a total function.

**Corollary 4.14.** Evolutionary information size $\mathrm{EIS}_{\mathbf{BBETM}}(x)$ is a total function with respect to the class **BGETM** of all bounded basic evolutionary Turing machines is a total function.

**Corollary 4.15.** Evolutionary information size $\mathrm{EIS}_{\mathbf{BBEITM}}(x)$ is a total function with respect to the class **BGEITM** of all bounded basic evolutionary inductive Turing machines is a total function.

**Corollary 4.16.** Evolutionary information size $\mathrm{EIS}_{\mathbf{PBETM}}(x)$ is a total function with respect to the class **PGEITM** of all periodic basic evolutionary Turing machines is a total function.

**Corollary 4.17.** Evolutionary information size $\mathrm{EIS}_{\mathbf{PBEITM}}(x)$ is a total function with respect to the class **PGEITM** of all periodic basic evolutionary inductive Turing machines is a total function.

Let us study other properties of the evolutionary information size.

**Lemma 4.2.** If $A$ is a deterministic evolutionary automaton, then for any positive number $n$, there is a positive number $m$ such that inequality $l(A(x)) > m$ implies inequality $l(x) > n$.

*Proof*. Let us take all words $x_1, x_2, x_3, \ldots, x_k$ with the length less than or equal to $n$. Then the lengths of all words $A(x_1)$, $A(x_2)$, $A(x_3)$, …, $A(x_k)$ are bounded and we can define $m = \max\ \{l(A(x_i));\ i = 1, 2, 3, \ldots, k\}$. Taking a word $w$ for which $l(A(w)) > m$, we see that $w$ does not belong to the set $\{x_1, x_2, x_3, \ldots, x_k\}$. Thus, $l(w) > n$.

Lemma is proved.

Lemma 4.2 implies the following result.

**Theorem 4.3.** If a class **H** of evolutionary automata/machines has an identity automaton and universal evolutionary automata/machines, while all automata from **H** are deterministic and have a finitary coding, then $\mathrm{EIS}_{\mathbf{H}}(x) \to \infty$ when $l(x) \to \infty$.

Indeed, taking a universal evolutionary automaton/machine $U$ from **H**, we see that by Lemma 4.2, for any positive number $n$, there is a positive number $m$ such that inequality $l(A(x)) > m$ implies inequality $l(x) > n$. It means that if $l(z) > m$ and $\mathrm{EIS}_{\mathbf{H}}(z) = \mathrm{EIS}_U(z) = r$, then $z = U(x)$ and $\mathrm{EIS}_U(z) = l(x)$. Thus, inequality $l(U(x)) > m$ implies inequality $\mathrm{EIS}_{\mathbf{H}}(x) = \mathrm{EIS}_U(x) > n$. So, when $l(x) \to \infty$, $\mathrm{EIS}_{\mathbf{H}}(x)$ also grows without limits.

**Corollary 4.18.** If all machines in **TM** are deterministic, then $\mathrm{EIS}_{\mathbf{PGETM}}(x) \to \infty$ when $l(x) \to \infty$.

**Corollary 4.19.** If all machines in **ITM** are deterministic, then $\mathrm{EIS}_{\mathbf{PGEITM}}(x) \to \infty$ when $l(x) \to \infty$.

**Corollary 4.20.** If all machines in **TM** are deterministic, then $\mathrm{EIS}_{\mathbf{PBETM}}(x) \to \infty$ when $l(x) \to \infty$.

**Corollary 4.21.** If all machines in **ITM** are deterministic, then $\mathrm{EIS}_{\mathbf{PBEITM}}(x) \to \infty$ when $l(x) \to \infty$.

There are interesting relations between the length of a word and the evolutionary information size of the same word.

**Proposition 4.5.** If the class **H** has universal automata and an identity automaton, *i.e.*, an automaton $H$ such that $H(x) = x$ for any word $x$ in the language of the automata from **H**, then there is a number $c$ such that for any word $x$, we have

$$\mathrm{EIS}_{\mathbf{H}}(x) \leq l(x) + c$$

Indeed, for an identity automaton $H$ from **H**, we have

$$\mathrm{EIS}_{H}(x) = l(x)$$

By Theorem 4.1, for all words $x$, we have

$$\mathrm{EIS}_{\mathbf{H}}(x) \leq \mathrm{EIS}_{H}(x) + c_H = l(x) + c_H$$

and we can take $c = c_{UH}$ to get the necessary result.

Note that similar relations exist for other information sizes, such as recursive information size [6] or inductive information size [7,8].

It is interesting to compare evolutionary information size $\mathrm{EIS}_{\mathbf{PBETM}}(x)$ with respect to the class **PBETM** of all periodic basic evolutionary Turing machines and information size $\mathrm{EIS}_{\mathbf{TM}}(x)$ with respect to the class **TM** of all Turing machines.

**Theorem 4.4.** For any increasing recursive function $h(x)$ that tends to infinity when $l(x) \to \infty$ and any inductively decidable set of finite objects (words) $V$, there are infinitely many elements $x$ from $V$ for which $h(\mathrm{EIS}_{\mathbf{TM}}(x)) > \mathrm{EIS}_{\mathbf{PBETM}}(x)$.

*Proof.* As it is proved in [3], working in the inductive mode, evolutionary Turing machines can simulate any simple inductive Turing machine computing the same word with the same input. As it is demonstrated above a universal evolutionary Turing machine is periodic. Thus, for any object (word) $w$, we have

$$\mathrm{EIS}_{\mathbf{PBETM}}(w) \leq \mathrm{EIS}_{\mathbf{TM}}(w)$$

At the same time, by Theorem 18 from [19], For any increasing recursive function $h(x)$ that tends to infinity when $l(x) \to \infty$ and any inductively decidable set $V$, there are infinitely many elements $w$ from $V$ for which $h(\mathrm{EIS}_{\mathbf{TM}}(w)) > \mathrm{EIS}_{\mathbf{ITM}}(w)$ where **ITM** is the class of all inductive Turing machines. Consequently, we have

$$h(\mathrm{EIS}_{\mathbf{TM}}(x)) > \mathrm{EIS}_{\mathbf{PBETM}}(x)$$

Infinitely many elements $x$ from $V$.

Theorem is proved.

Theorem 4.4 means that evolution can essentially reduce information size of objects because for infinitely many objects, their information size defined by periodic basic evolutionary Turing machines is essentially less than their information size defined by Turing machines.

**Corollary 4.22.** In any recursive set $V$, there are infinitely many elements $x$ for which $\ln_2(\mathrm{EIS}_{\mathbf{TM}}(x)) > \mathrm{EIS}_{\mathbf{PBETM}}(x)$.

**Corollary 4.23.** For any natural number $n$ and any recursive set $V$, there are infinitely many elements $x$ for which $(\mathrm{EIS}_{\mathbf{TM}}(x))^{1/n} > \mathrm{EIS}_{\mathbf{PBETM}}(x)$.

## 5. Evolutionary Information in an Object

Here we introduce and study the *quantity of evolutionary information in* an object, e.g., in a text, that allows making simpler development/construction of another object by a given system of evolutionary algorithms (evolutionary automata).

To define evolutionary information in an object, we consider the set $X^*$ of all words in an alphabet $X$ and three mappings $<\cdot,\cdot>: X^* \times X^* \to X^*$, $\lambda: X^* \to X^*$ and $\pi: X^* \to X^*$ such that for any words $u$ and $v$ from $X^*$, we have $\lambda(<u, v>) = u$ and $\pi(<u, v>) = v$. The standard coding of pairs of words has the following form

$$<u, v> = 1^{l(u)}0uv$$

where $l(v)$ is the length of the word $v$. Its dual coding of pairs has the form

$$<u, v> = 1^{l(v)}0vu$$

To study relative information size and relative quantity of information, we need Condition L introduced in Section 3 and its dual condition.

**Condition L°.** For any word $v$ from $X^*$, there is a number $c_v$ such that for any word $u$ from $X^*$, we have

$$l(<u, v>) \leq l(u) + c_v$$

For the coding of pairs dual to the standard coding $<\cdot,\cdot>$, the constant $c_v$ is equal to $2l(v) + 1$.

As before, **K** is an arbitrary class of automata with input and two outputs, and **H** is an arbitrary class of evolutionary automata/machines/algorithms. Taking automata from **H**, we can define the *relative quantity of evolutionary information about*, or *relative evolutionary information size* of, an object (word) $x$.

**Definition 5.1.** (a) The *left quantity* $\mathrm{EIS}^{\mathrm{L}}{}_A(x; y)$ *of evolutionary information about* an object/word $x$, also called the *left evolutionary information size* $\mathrm{EIS}^{\mathrm{L}}{}_A(x; y)$ *of* an object/word $x$, *with respect to* an automaton/machine/algorithm $A$ from **H** *relative* to an object/word $y$ is defined as

$$\mathrm{EIS}^{\mathrm{L}}{}_A(x; y) = \min \{l(p); A(<p, y>) = x\}$$

where $l(p)$ is the length of the word $p$, while in the case when there are no words $p$ such that $A(<p, y>) = x$, $\mathrm{EIS}^{\mathrm{L}}{}_A(x; y)$ is not defined; (b) The *right quantity* $\mathrm{EIS}^{\mathrm{R}}{}_A(x; y)$ *of evolutionary information about* an object/word $x$, also called the *right evolutionary information size* $\mathrm{EIS}^{\mathrm{R}}{}_A(x; y)$ *of* an

object/word $x$, *with respect to* an automaton/machine/algorithm $A$ from **H** *relative* to an object/word $y$ is defined as

$$\mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y) = \min \{l(p); A(<y, p>) = x\}$$

where $l(p)$ is the length of the word $p$, while in the case when there are no words $p$ such that $A(<p, y>) = x$, $\mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y)$ is not defined.

Informally, the relative quantity of evolutionary information about an object $x$ relative to (with respect to) an evolutionary automaton $A$ shows how much information it is necessary for building (computing or constructing) this object by means of the automaton $A$ that has some additional information $y$. For instance, some evolutionary automata, when they are given the object $x$, which naturally contains all information about the construction (structure) of itself, do not need any additional information to build (compute or construct) $x$. So, for these automata, the evolutionary information size of relative to $x$ is zero. At the same time, other evolutionary automata need a lot of additional information to build (compute or construct) $x$ even when the object $x$ is given to them.

**Proposition 5.1.** If the class **H** has a finitary coding and the coding simulation $<\cdot,\cdot>$ satisfies Condition L, then for any evolutionary automaton/machine $A$ from the class **H**, there is a constant number $c_{AU}$ such that for any object/word $x$, we have

$$\mathrm{EIS}_{A}(x) \leq \mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y) + c_{y}$$

*Proof*. Let us take an evolutionary automaton/machine $A$ from the class **H** and a word $x$. Then by Definitions 3.1 and 4.1,

$$\mathrm{EIS}_{A}(x) = \min \{l(q); A(q) = x\} = l(q_0)$$

and

$$\mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y) = \min \{l(p); U(<y, p>) = x\} = l(p_0)$$

By Condition L, there is a number $c_y$ such that for the word $p_0$ from $X^*$, we have

$$l(q_0) \leq l(<y, p_0>) = l(p_0) + c_y$$

Thus,

$$\mathrm{EIS}_{A}(x) \leq \mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y) + c_{y}$$

Proposition is proved.

**Corollary 5.1.** For any algorithm $A$ from **H**, we have $\mathrm{EIS}_{A}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y)$.

We can see that in a general case, the right relative evolutionary information size of an object $x$ does not coincide with the left relative evolutionary information size of an object $x$. This peculiarity shows that the relative evolutionary information size depends not only on the additional information in the word $y$ but also how this word $y$ is processed by the automaton $A$.

**Definition 5.2.** (a) The *right quantity* $\mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y)$ *of evolutionary information in* an object/word $y$ about an object/word $x$ *with respect to* an automaton/machine/algorithm $A$ from **H** is defined as

$$EIQ^R_A(y; x) = EIS^R_A(x) - EIS^R_A(x; y)$$

when both quantities are defined and $EIQ^R_A(y; x)$ is undefined otherwise; (b) The *left quantity* $EIS^L_A(x; y)$ *of evolutionary information in* an object/word $y$ about an object/word $x$ *with respect to* an automaton/machine/algorithm $A$ from **H** is defined as

$$EIQ^L_A(y; x) = EIS^L_A(x) - EIS^L_A(x; y)$$

when both quantities are defined and $EIQ^L_A(y; x)$ is undefined otherwise.

Informally, the quantity of evolutionary information in an object/word $y$ about an object $x$ with respect to an automaton/machine/algorithm $A$ shows to what extent utilization of information in $y$ reduces information necessary for building (computing or constructing) this object by means of the automaton $A$ without any additional information.

Here we encounter the problem of *negative information* because the quantity $EIQ^L_A(x; y)$ *of* evolutionary information can be negative when the size $EIS^L_A(x; y)$ is larger than the size $EIS^L_A(x)$. This happens when, for example, the automaton $A$ needs an additional program to extract the word $y$ from the code $<y, p>$ given to it as the input for computing $x$.

In essence, there are three meanings of negative information:

(1) Information about something negative, *i.e.*, information with a negative connotation, is called negative information.

(2) Information expressed with a negation of something is called negative information.

(3) Information that has negative measure is called negative information.

Here we study evolutionary measures of information. Consequently, we are interested in the third meaning. In the traditional approach to information, it is always assumed that the measure, e.g., quantity, of information is always positive. However, recently in their exploration of quantum information, researchers came to the conclusion that there is quantum information with negative measure [24,25].

The general theory of information also permits different types of information with negative measures [8]. For instance, misinformation can be treated as a kind of negative information, or more exactly, as information with the negative measure of correctness.

Evolutionary information theory, as well as algorithmic information theory, explains how information can be negative. Indeed, information in a word (an object) $y$ about a word (an object) $x$ can contain noise that makes computation (construction) of $x$ more complicated. As a result, this information will be negative.

Misleading information can also increase the information size of an object because it will demand additional information for automata or algorithms to detect misleading, to eliminate it and to guide automata or algorithms in the right direction. Thus, it becomes understandable that algorithmic information in general and evolutionary information in particular can be negative in many situations.

Evolutionary information theory, as well as algorithmic information theory, also gives a grounded solution to the information paradox called by Hintikka "scandal of deduction" [26]. Its essence is that the traditional approach to information claims that deduction does give new information or new knowledge, having no empirical content. For instance, if an agent knows $p$ and $q$ is deduced from $p$, then it is assumed that the agent knows $q$.

To understand the situation, we remind that deduction is an evolutionary process of and a tool for evolutionary knowledge integration. Thus, it is natural to use evolutionary information theory for making sense of the "scandal of deduction".

According to this theory, extraction of information from a message (statement) is an evolutionary process, while information content of the message (statement) depends not only on the message itself but mostly on algorithms used for information extraction.

To illustrate this principle of evolutionary information theory, let us consider the following situation that involved Mr. B and Mr. F.

Two men, Mr. B and Mr. F, are sitting at a train station not far from London. Little Mr. F feels himself very important. He has read different articles and even some books about information. He even wrote something in that area. So, he thinks he knows everything about information.

Eventually a third man comes and asks, "When does the train to London come?" Little Mr. F starts thinking how to better share his wisdom advising that man to look into the schedule. But before Mr. F is ready, Mr. B says: "The train to London either comes at 8 p.m. or does not come at 8 p.m." Then he gets up and leaves the station. The third man follows Mr. B.

Little Mr. F is shocked—why instead of telling something reasonable that passenger uttered a sentence that was not informative at all. Indeed, according to MTI (Mathematical Theory of Information), TWSI (Theory of Weakly Semantic Information) and TSSI (Theory of Strongly Semantic Information) tautologies are not informative. Little Mr. F read this in the book "The Philosophy of Information".

May be, little Mr. F thinks, the goal of the response was mocking at the third man and now the third man is going to punish that stupid passenger. Little Mr. F is happy with this thought.

However, the third man got some information from the tautological statement of Mr. B because after leaving the station, the third man asks Mr. B, "Are you Mr. James Bond?" "Yes, I am," answers Mr. B. "Then", says the third man, "let's go to my car."

This imaginary episode shows that information is a more complicated phenomenon than many think and even tautologies can be informative depending on the context, *a priory* knowledge of the communicating systems and on algorithms used for information extraction. The third man and Mr. F applied different algorithms for understanding the answer of Mr. B and came to different results: due to incorrect conjectures, Mr. F was not able to obtain any information, while the third man got information he needed at that time.

As we did before, it is necessary to go from the quantity of evolutionary information with respect to one algorithm to the quantity of evolutionary information with respect to systems of algorithms because people, computers and social organizations use systems (classes) of automata/algorithms and not a single automaton/algorithm. To define the quantity of information in an object $y$ about an object $x$ with respect to a class of automata/algorithms, algorithmic information theory suggests using universal automata/algorithms as representatives of the considered class [5,8]. Similar approach works in the case of evolutionary information.

Taking a universal automaton $U$ in the class **H**, we can define relatively invariant optimal evolutionary information about an object in the class **H**.

**Definition 5.3.** (a) The *right quantity* $\mathrm{EIS}^{R}_{\mathbf{H}}(x; y)$ *of evolutionary information about* an object/word $x$, also called the *evolutionary information size* $\mathrm{EIS}^{R}_{\mathbf{H}}(x; y)$ *of* an object/word $x$, with respect to the class **H** *relative* to an object/word $y$ is defined as

$$\mathrm{EIS}^{R}_{\mathbf{H}}(x; y) = \min \{l(p); U(<y, p>) = x\}$$

in the case when there are no words $p$ such that $U(p, y) = x$, $\mathrm{EIS}_{\mathbf{H}}(x; y)$ is not defined; (b) The *left quantity* $\mathrm{EIS}^{L}_{A}(x; y)$ *of evolutionary information about* an object/word $x$, also called the *left evolutionary information size* $\mathrm{EIS}^{L}_{A}(x; y)$ *of* an object/word $x$, *with respect to* the class **H** *relative* to an object/word $y$ is defined as

$$\mathrm{EIS}^{L}_{\mathbf{H}}(x; y) = \min \{l(p); U(<p, y>) = x\}$$

in the case when there are no words $p$ such that $U(<p, y>) = x$, $\mathrm{EIS}^{L}_{U}(x; y)$ is not defined.

Examples of quantities of evolutionary information about an object $x$ with respect to a class of evolutionary automata/machines are recursive quantity of evolutionary information about an object $x$ in this case, e.g., **H** is one of the classes **BGETM**, **PGETM**, **APGETM BGETM**, **PGETM** or **APGETM**) or inductive quantity of evolutionary information about an object $x$ in this case, e.g., **H** is one of the classes **BGEITM**, **PGEITM**, **APGEITM BGEITM**, **PGEITM** or **APGEITM**).

Note that if there are no words $p$ such that $U(p, y) = x$, then for any evolutionary automaton $A$ from **H**, there are no words $p$ such that $A(p, y) = x$. In particular, functions $\mathrm{EIS}^{R}_{\mathbf{H}}(x; y)$ and $\mathrm{EIS}^{L}_{\mathbf{H}}(x; y)$ are defined or undefined independently of the choice of universal algorithms for their definition.

Informally, quantity of evolutionary information about an object $x$ with respect to the class **H** shows how much information it is necessary for building (computing or constructing) this object by means of some automaton $U$ universal in the class **H** and having additional information $y$, *i.e.*, $\mathrm{EIS}_{\mathbf{H}}(x; y) = \mathrm{EIS}_{U}(x; y)$.

Observe that when **H** consists of a single evolutionary automaton/machine $A$, the functions $\mathrm{EIS}_{\mathbf{H}}(y; x)$ and $\mathrm{EIS}_{A}(y; x)$ coincide.

It is possible to show that relative evolutionary information size (relative evolutionary information about an object $x$) with respect to the class **H** is additively optimal when the automata from **H** have finite simulation coding.

**Theorem 5.1.** If the class **H** has a finitary coding $<\cdot,\cdot>$, then for any evolutionary automaton/machine $A$ from the class **H**, any word $y$ and any universal evolutionary automaton $U$ in **H**, there is a constant number $c_{AUy}$ such that for any object/word $x$, we have

$$\mathrm{EIS}^{R}_{U}(x; y) \leq \mathrm{EIS}^{R}_{A}(x; y) + c_{AUy}$$

when the coding $<\cdot,\cdot>$ satisfies Condition L and universal algorithm $U$ uses the direct pairing, while

$$\mathrm{EIS}^{L}_{U}(x; y) \leq \mathrm{EIS}^{L}_{A}(x; y) + c_{AUy}$$

when the coding $<\cdot,\cdot>$ satisfies Condition L° and universal algorithm $U$ uses the dual pairing.

*Proof.* Let us assume that the coding $<\cdot,\cdot>$ satisfies Condition L and take an evolutionary automaton/machine $A$ from the class **H** and a word $x$. Then by Definitions 3.1 and 4.1,

$$\mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y) = \min\ \{l(q);\ A(<y, p>) = x\} = l(q_0)$$

and

$$\mathrm{EIS}^{\mathrm{R}}{}_{U}(x; y) = \min\ \{l(p);\ U(<\mathbf{c}(A), <y, p>>) = x\} = l(p_0)$$

where <·,·> is a coding of pairs of words and $U$ uses the direct pairing. By Condition L, there are numbers $c_y$ and $c_{\mathbf{c}(A)}$ such that for any word $u$ from $X^*$, we have

$$l(<\mathbf{c}(A), <p_0\,, y>>) \le l(<p_0\,, y>>) + c_{\mathbf{c}(A)} \le l(p_0) + c_y + c_{\mathbf{c}(A)} = l(p_0) + c_{AUy}$$

Then as $U(<\mathbf{c}(A), <p_0, y>>) = x$, we have

$$\mathrm{EIS}^{\mathrm{R}}{}_{U}(x; y) = l(q_0) \le l(<\mathbf{c}(A), <p_0\,, y>>) \le l(p_0) + c_{AUy}$$

Thus,

$$l(q_0) \le l(p_0) + c_{AUy}$$

and

$$\mathrm{EIS}^{\mathrm{R}}{}_{U}(x; y) \le \mathrm{EIS}^{\mathrm{R}}{}_{A}(x; y) + c_{AUy}$$

For the standard coding <·,·>, $c_{AUy} = 2l(y) + 2l(\mathbf{c}(A)) + 2$. It means that $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{H}}(x; y)$ is additively optimal.

Proof of the second part is similar.

Theorem is proved.

There are various relations between relative and absolute evolutionary information sizes. For instance, Proposition 5.1 implies the following result.

**Corollary 5.2.** $\mathrm{EIS}_{\mathbf{H}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{H}}(x; y)$.

As the class of all Turing machines satisfies all necessary conditions [20], Proposition 5.1 implies the following results.

**Corollary 5.3.** (a) $\mathrm{EIS}_{\mathbf{BGETM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{BGETM}}(x;\ y)$; (b) $\mathrm{EIS}_{\mathbf{PGETM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{PGETM}}(x;\ y)$; (c) $\mathrm{EIS}_{\mathbf{APGETM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{APGETM}}(x; y)$.

**Corollary 5.4.** (a) $\mathrm{EIS}_{\mathbf{BBETM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{BBETM}}(x;\ y)$; (b) $\mathrm{EIS}_{\mathbf{PBETM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{PBETM}}(x;\ y)$; (c) $\mathrm{EIS}_{\mathbf{APBETM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{APBETM}}(x; y)$.

As the class of all inductive Turing machines satisfies all necessary conditions [20], Proposition 5.1 implies the following results.

**Corollary 5.5.** (a) $\mathrm{EIS}_{\mathbf{BGEITM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{BGEITM}}(x;\ y)$; (b) $\mathrm{EIS}_{\mathbf{PGEITM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{PGEITM}}(x;\ y)$; (c) $\mathrm{EIS}_{\mathbf{APGEITM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{APGEITM}}(x; y)$.

**Corollary 5.6.** (a) $\mathrm{EIS}_{\mathbf{BBEITM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{BBEITM}}(x;\ y)$; (b) $\mathrm{EIS}_{\mathbf{PBEITM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{PBEITM}}(x)$; (c) $\mathrm{EIS}_{\mathbf{APBEITM}}(x) \preccurlyeq \mathrm{EIS}^{\mathrm{R}}{}_{\mathbf{APBEITM}}(x)$.

To compare left relative and absolute evolutionary information sizes, we use conditions introduced in [20].

**Weak Comparison Condition CPW$_v$.** Algorithms/automata from the class **K** allow one to compute the following comparison predicate:

$$P_v(w) = \begin{cases} 1, \text{when } w = v \\ \text{undefined}, \text{when } w \neq v \end{cases}$$

*i.e.*, for any words $w$ and $v$, there is an algorithm/automaton $A_v$ in **K** that computes this predicate.

**Right Weak Composition Comparison Condition RWCPA$_v$.** For any algorithm/automaton $A$ in **K**, the computing sequential composition $A \circ A_v$ is a member of **K**.

**Constant Condition CC**. For any element $v \in$ R*(**K**), the function $g_v$, which always takes one and the same value $v$, *i.e.*, $g_v(u) = v$ for all $u \in DD^*(\mathbf{K})$, is computed by some algorithm/automaton $E_v$ from **K**.

**Right Compositional Constant Condition RCCC**. For any algorithm/automaton $A$ from **K**, the computing sequential composition $A \circ E_v$ belongs to **K**.

Let us assume that the class **K** satisfies Conditions CPW$_v$, RWCPA$_v$ , CC, and RCCC, while the class **H** of evolutionary **K**-machines has a finitary coding and the coding <·,·> of pairs satisfies Condition L, while the function $\lambda$: $X^* \to X^*$ is computed by some machine $C_\lambda$ from **K**.

**Proposition 5.2.** For any evolutionary automaton/machine $A$ from the class **H** and any universal evolutionary automaton $U$ in **H**, there is a constant number $c_{AU}$ such that for any object/word $x$, we have

$$\mathrm{EIS}^{\mathrm{L}}{}_U(x; y) \leq \mathrm{EIS}^{\mathrm{L}}{}_U(x) + c_{Uy}$$

*Proof*. Let us take, a universal automaton/machine $V$ in the class **K**, a word $y$ and build an automaton/machine $A$ from the class **K** such that gives the result $x$ with an input $<p, y>$ if and only if $V(p) = x$. To do this, we define

$$\mathrm{A} = \mathrm{C}_\lambda \circ \mathrm{V} \circ \mathrm{A}_\mathrm{x} \circ \mathrm{E}_\mathrm{x}$$

By the listed properties of the class **K**, the automaton/machine $A$ belongs to **K**.

Given the input $<p, y>$, the automaton/machine $A$ works in the following way. At first, the automaton/machine $C_\lambda$ converts the word $<p, y>$ into the word $p$. Then the universal automaton/machine $V$ works with $p$. When $V$ does not give the result, $A$ also does not give the result. When $V$ gives the result $w$, then $w$ goes to the machine $A_x$ as its input. If $w$ is not equal to $x$, $A_x$ does not give the result. If $w$ is equal to $x$, $A_x$ gives the result 1. This result goes to the machine $E_x$ as its input. By properties of the machine $E_x$ , its output is $x$. Thus, the automaton/machine $A$ gives the result $x$ with an input $<p, y>$ if and only if $V(p) = x$.

This allows us to build the evolutionary automaton/machine $E = \{A[t]; t = 1, 2, 3, ...\}$ in which all automata $A[t] = A$ and the evolutionary automaton/machine $U = \{V[t]; t = 1, 2, 3, ...\}$ in which all automata $V[t] = V$ ($t = 1, 2, 3, ...$). If **H** is the class of general evolutionary **K**-machines, then by Theorem 3.1, $U$ is a universal evolutionary automaton/machine in **H**. If **H** is the class of basic evolutionary **K**-machines, then by Theorem 3.2, $U$ is a universal evolutionary automaton/machine in **H**. Thus,

$$\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{H}}(x; y) = \mathrm{EIS}^{\mathrm{L}}{}_U(x; y)$$

By construction, the evolutionary **K**-machine $E$ gives the result $x$ with an input $<p, y>$ if and only if $U(p) = x$. Consequently, $\mathrm{EIS}^{\mathrm{L}}{}_{A}(x; y) = \mathrm{EIS}_{U}(x) = \mathrm{EIS}_{\mathbf{H}}(x)$.

By Theorem 5.1, for the evolutionary automaton/machine $E$, there is a constant number $c_{EUy}$ such that for any object/word $x$, we have

$$\mathrm{EIS}^{\mathrm{L}}{}_{U}(x; y) \leq \mathrm{EIS}^{\mathrm{L}}{}_{A}(x; y) + c_{EUy}$$

Consequently,

$$\mathrm{EIS}^{\mathrm{L}}{}_{U}(x; y) \leq \mathrm{EIS}_{U}(x) + c_{EUy}$$

Proposition is proved.

As $\mathrm{EIS}^{\mathrm{L}}{}_{U}(x; y) = \mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{H}}(x; y)$, we have the following result.

**Corollary 5.7.** $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{H}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{H}}(x)$.

As the class of all Turing machines satisfies all necessary conditions [20], Proposition 5.2 implies the following results.

**Corollary 5.8.** (a) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{BGETM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{BGETM}}(x)$; (b) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{PGETM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{PGETM}}(x)$; (c) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{APGETM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{APGETM}}(x)$.

**Corollary 5.9.** (a) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{BBETM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{BBETM}}(x)$; (b) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{PBETM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{PBETM}}(x)$; (c) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{APBETM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{APBETM}}(x)$.

As the class of all inductive Turing machines satisfies all necessary conditions [20], Proposition 5.2 implies the following results.

**Corollary 5.10.** (a) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{BGEITM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{BGEITM}}(x)$; (b) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{PGEITM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{PGEITM}}(x)$; (c) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{APGEITM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{APGEITM}}(x)$.

**Corollary 5.11.** (a) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{BBEITM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{BBEITM}}(x)$; (b) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{PBEITM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{PBEITM}}(x)$; (c) $\mathrm{EIS}^{\mathrm{L}}{}_{\mathbf{APBEITM}}(x; y) \preccurlyeq \mathrm{EIS}_{\mathbf{APBEITM}}(x)$.

Let us find what relations exist between right relative evolutionary information size and left relative evolutionary information size of the same object. To do this, we consider the function $sw$: $X^* \to X^*$ such that $sw(<x, y>) = <y, x>$ for any elements $x$ and $y$ from $X^*$.

**Switching Code Condition SCC**. The function $sw$ is computed by some algorithm/automaton $A_{sw}$ from **K**.

For instance, the code $<u, v> = 1^{l(u)}0uv$ can be converted to the code $<v, u> = 1^{l(v)}0vu$ by an appropriate Turing machine.

**Right Compositional Switching Code Condition RCSCCC**. For any algorithm/automaton $A$ from **K**, the computing sequential composition $A_{sw} \circ A$ belongs to **K**.

**Proposition 5.3.** If the class **H** has a finitary coding, class **K** satisfies Conditions RCSCCC and SCC, has free automata, while the coding simulation $<\cdot,\cdot>$ satisfies Condition L, then for any word $y$ and for any universal evolutionary automaton/machine $U$ from the class **H**, the right evolutionary

information size $EIS^R_U(x)$ and the left evolutionary information size $EIS^L_U(x)$ are additively equivalent, *i.e.*, $EIS^R_U(x) \asymp EIS^L_U(x)$.

*Proof*. Let us take, an evolutionary automaton/machine $U = \{U[t]; t = 1, 2, 3, ...\}$ universal in the class **H** in which all level automata $U[t]$ are copies of a universal in **K** automaton $V$ (cf. Theorem 3.1) and a word $y$. This allows us to build the evolutionary **K**-machine $D = A_{sw} \circ U$ from the class **H** by changing the first level automaton $U[1]$ in $U$ by the sequential composition $A_{sw} \circ V$. This sequential composition exists in the **K** because **K** has free automata and satisfies Conditions RCSCCC and SCC.

Given the input $<y, p>$, the automaton/machine $D$ works in the following way. At first, the automaton/machine $A_{sw}$ converts the word $<y, p>$ into the word $<p, y>$. Then the universal automaton/machine $U[1] = V$ works with $<p, y>$. Then the evolutionary process continues involving machines $U[t]$ with $t = 2, 3, ....$ Thus, $D(<y, p>) = U(<p, y>)$. Consequently,

$$EIS^R_D(x; y) = EIS^L_U(x; y)$$

At the same time, by Theorem 5.1, for the evolutionary automaton/machine $D$, there is a constant number $c_{DUy}$ such that for any object/word $x$, we have

$$EIS^R_U(x; y) \leq EIS^R_D(x; y) + c_{DUy}$$

Consequently,

$$EIS^R_U(x; y) \leq EIS^L_U(x; y) + c_{DUy}$$

In a similar way, we prove

$$EIS^L_U(x; y) \leq EIS^R_U(x; y) + c_{GUy}$$

for a constant number $c_{Guy}$. Thus, we have

$$EIS^R_U(x; y) \asymp EIS^L_U(x; y)$$

Proposition is proved.

Optimality results allow us to define the quantities $EIS^R_{\mathbf{H}}(x; y)$ and $EIS^L_{\mathbf{H}}(x; y)$ of evolutionary information in an object/word $y$ about an object/word $x$ with respect to the class **H**.

**Definition 5.4.** (a) The *right quantity* $EIS^R_{\mathbf{H}}(x; y)$ *of evolutionary information in* an object/word $y$ about an object/word $x$ *with respect to* the class **H** is defined as

$$EIQ^R_{\mathbf{H}}(y; x) = EIS^R_U(x) - EIS^R_U(x; y)$$

for some automaton $U$ universal in the class **H** when both quantities are defined and $EIQ_{\mathbf{H}}(y; x)$ is undefined otherwise; (b) The *left quantity* $EIS^L_A(x; y)$ *of evolutionary information in* an object/word $y$ about an object/word $x$ *with respect to* the class **H** is defined as

$$EIQ^L_{\mathbf{H}}(y; x) = EIS^L_U(x) - EIS^L_U(x; y)$$

for some automaton $U$ universal in the class **H** when both quantities are defined and $EIQ_{\mathbf{H}}(y; x)$ is undefined otherwise.

Informally, quantity of evolutionary information in an object/word $y$ about an object $x$ with respect to the class **H** shows to what extent utilization of evolutionary information in $y$ reduces evolutionary

information necessary for building (computing or constructing) this object by means of some evolutionary automaton $U$ universal in the class **H** without any additional information.

As $\mathrm{EIS}^{L}{}_{U}(x;\ y) = \mathrm{EIS}^{L}{}_{\mathbf{H}}(x;\ y)$ and $\mathrm{EIS}^{R}{}_{U}(x;\ y) = \mathrm{EIS}^{R}{}_{\mathbf{H}}(x;\ y)$, Proposition 5.3 implies the following result.

**Corollary 5.12.** If the class **H** satisfies all necessary conditions from Proposition 5.3, $\mathrm{EIS}^{R}{}_{\mathbf{H}}(x; y) \asymp \mathrm{EIS}^{L}{}_{\mathbf{H}}(x; y)$.

Note that when the class **H** consists of a single evolutionary automaton/machine $A$, functions $\mathrm{EIQ}_{\mathbf{H}}(y; x)$ and $\mathrm{EIQ}_{A}(y; x)$ coincide.

If the class **H** satisfies all necessary conditions from Proposition 5.2, then we have the following results.

**Corollary 5.13.** $\mathrm{EIQ}^{L}{}_{\mathbf{H}}(x; y) \succcurlyeq 0$.

As the class of all Turing machines satisfies all necessary conditions [20], Proposition 5.2 implies the following results.

**Corollary 5.14.** (a) $\mathrm{EIQ}^{L}{}_{\mathbf{BGETM}}(x; y) \succcurlyeq 0$; (b) $\mathrm{EIQ}^{L}{}_{\mathbf{PGETM}}(x; y) \succcurlyeq 0$; (c) $\mathrm{EIQ}^{L}{}_{\mathbf{APGETM}}(x; y) \succcurlyeq 0$.

**Corollary 5.15.** (a) $\mathrm{EIQ}^{L}{}_{\mathbf{BBETM}}(x; y) \succcurlyeq 0$; (b) $\mathrm{EIQ}^{L}{}_{\mathbf{PBETM}}(x; y) \succcurlyeq 0$; (c) $\mathrm{EIQ}^{L}{}_{\mathbf{APBETM}}(x; y) \succcurlyeq 0$.

As the class of all inductive Turing machines satisfies all necessary conditions [20], Proposition 5.2 implies the following results.

**Corollary 5.16.** (a) $\mathrm{EIQ}^{L}{}_{\mathbf{BGEITM}}(x; y) \succcurlyeq 0$; (b) $\mathrm{EIQ}^{L}{}_{\mathbf{PGEITM}}(x; y) \succcurlyeq 0$; (c) $\mathrm{EIQ}^{L}{}_{\mathbf{APGEITM}}(x; y) \succcurlyeq 0$.

**Corollary 5.17.** (a) $\mathrm{EIQ}^{L}{}_{\mathbf{BBEITM}}(x; y) \succcurlyeq 0$; (b) $\mathrm{EIQ}^{L}{}_{\mathbf{PBEITM}}(x; y) \succcurlyeq 0$; (c) $\mathrm{EIQ}^{L}{}_{\mathbf{APBEITM}}(x; y) \succcurlyeq 0$.

It is possible to show that evolutionary information in an object $y$ with respect to the class **H** is additively invariant when the automata from **H** have finite simulation coding. Namely, we have the following result.

**Theorem 5.2.** If the class **H** has a finitary coding, then for any two universal automata $V$ and $U$ in **H**, the quantities $\mathrm{EIQ}^{L}{}_{U}(y; x)$ and $\mathrm{EIQ}^{L}{}_{V}(y; x)$, as well as the quantities $\mathrm{EIQ}^{R}{}_{U}(y; x)$ and $\mathrm{EIQ}^{R}{}_{V}(y; x)$ of evolutionary information in an object $y$ are additively equivalent.

*Proof*. Let us consider two universal automata $V$ and $U$ in **H**. Then

$$\mathrm{EIQ}^{L}{}_{U}(y; x) = \mathrm{EIS}_{U}(x) - \mathrm{EIS}^{L}{}_{U}(x; y)$$

and

$$\mathrm{EIQ}^{L}{}_{V}(y; x) = \mathrm{EIS}_{V}(x) - \mathrm{EIS}^{L}{}_{V}(x; y)$$

As $V$ and $U$ are both universal automata in **H**, by Theorem 4.1, we have

$$\mathrm{EIS}_{U}(x) \leq \mathrm{EIS}_{V}(x) + c_{UV}$$

and by Theorem 5.1, we have

$$\mathrm{EIS}_V(x; y) \leq \mathrm{EIS}_U(x; y) + c_{V\mathrm{y}}$$

Thus,

$$\mathrm{EIS}^{\mathrm{L}}{}_U(x; y) \geq \mathrm{EIS}^{\mathrm{L}}{}_V(x; y) - c_{V\mathrm{y}}$$

Consequently,

$$\begin{aligned}
&\mathrm{EIQ}^{\mathrm{L}}{}_U(y; x) = \mathrm{EIS}_U(x) - \mathrm{EIS}^{\mathrm{L}}{}_U(x; y) \leq \mathrm{EIS}_V(x) + c_{UV} - (\mathrm{EIS}^{\mathrm{L}}{}_V(x; y) - c_{V\mathrm{y}}) \\
&= \mathrm{EIS}_V(x) + c_{UV} - \mathrm{EIS}^{\mathrm{L}}{}_V(x; y) + c_{V\mathrm{y}} \\
&= \mathrm{EIS}_V(x) - \mathrm{EIS}^{\mathrm{L}}{}_V(x; y) + (c_{UV} + c_{V\mathrm{y}}) \\
&= \mathrm{EIQ}^{\mathrm{L}}{}_V(y; x) + c
\end{aligned}$$

where

$$c = c_{UV} + c_{V\mathrm{y}}$$

In a similar way, we prove

$$\mathrm{EIQ}^{\mathrm{L}}{}_V(y; x) \leq \mathrm{EIQ}^{\mathrm{L}}{}_U(y; x) + k$$

Thus, the quantities $\mathrm{EIQ}^{\mathrm{L}}{}_U(y; x)$ and $\mathrm{EIQ}^{\mathrm{L}}{}_V(y; x)$ of information in an object $y$ are additively equivalent.

Proof for the quantities $\mathrm{EIQ}^{\mathrm{R}}{}_U(y; x)$ and $\mathrm{EIQ}^{\mathrm{R}}{}_V(y; x)$ is similar.

Theorem is proved.

Theorem 5.2 shows additive invariance of the evolutionary information in an object $y$ with respect to the classes **BGETM**, **PGETM**, **APGETM**, **BBEITM**, **PBEITM** and **APBEITM**.

However, it is necessary to remark that evolutionary information is additively invariant only as a function. For an individual object, choice of different universal algorithms for defining evolutionary information can result in big differences in evolutionary information estimation for this object.

## 6. Modeling Physical Evolution with Evolutionary Machines

To demonstrate applicability of evolutionary information theory to all material processes, in this section, we describe modeling physical evolution with evolutionary machines. In physics, evolution is often called *time evolution* and is described as changes of the physical system states, which are brought about by the passage of time and are subject to the corresponding physical laws. For instance, in the Newtonian mechanics, time evolution of a collection of rigid bodies is portrayed by Newton's laws of motion. In Lagrangian mechanics, the laws of time evolution are represented by Lagrangian equations, while in Hamiltonian mechanics, they are represented by Hamiltonian equations. The classical quantum mechanics describes time evolution by the Schrödinger equation.

The standard interpretations and applications of evolutionary machines and computations include optimization and modeling of biological and social processes [3,27,28]. However, evolutionary machines allow other interesting and important interpretations and applications. One area of such interpretations and applications concerns the very foundations of physics.

According to the basic physical theory called *loop quantum gravity*, geometry of space-time is described by *spin networks*, while matter exists at the nodes of these spin networks [15–17]. In

essence, a spin network is a labeled graph. In it, nodes represent volumes (or informally, "lumps" or "bricks") of space, which are basic spatial elements, and edges (called *lines* of spin networks) describe boundaries (spatial surface areas) and connections of these spatial elements. Each node is marked by a number and this number specifies the volume (quantity of space) represented by this node. In a similar way, each line (link) is marked by a number and informally, this number specifies the area of the surface represented by this line [16]. In a formal representation, a half integer number $s_i$ is associated with each link $\gamma_i$, or equivalently, an integer number $p_i$, the "color", marks this link and $p_i = 2s_i$ . In such a way, a spin network is labeled (named) by numbers.

Every quantum state corresponds to a spin network. The mathematics that describes the quantum states of spatial volumes and areas provides rules for how the nodes and lines (links) can be connected and what numbers can be assigned to them. For instance, when a node has three links, then the labels (colors) on the links satisfy the well-known Clebsh-Gordon condition: Each color is not larger than the sum of the other two, and the sum of the three colors is even [17]. As a result, every spin network that obeys the necessary rules corresponds to a quantum state.

Subatomic particles, such as quarks or electrons, correspond to certain types of nodes in a spin network, and are represented by adding more labels on nodes [17]. These labels signify physical characteristics of the corresponding particles. Physical fields, such as the electromagnetic field, are represented by additional labels on lines in a spin network. These labels indicate physical characteristics of the corresponding fields.

This shows that spin networks are special cases of named sets, namely they are labeled networks or labeled graphs, while matter is represented in loop quantum gravity by naming (labeling) operations [29].

Material particles change their positions and interact with one another and with physical fields, while physical fields change their characteristics. These changes are naturally modeled by naming operations, such as renaming and reinterpreting [15,29]. Moreover, according to general relativity, the geometry of space also changes with time. The bends and curves of the physical space change as matter and energy move, and waves can pass through it like ripples on a lake [30]. In loop quantum gravity, these processes are modeled by operations with the spin networks [15], which can be mathematically described by operations with named sets [29].

Interestingly, there are computational models that work with labeled networks or labeled graphs. The first of such models is called a *Kolmogorov algorithm* [7,31]. In contrast to the most popular algorithmic models, such as Turing machines or formal grammars, Kolmogorov algorithms work not with words but with configurations. A *configuration* consists of groups of symbols connected by relations, *i.e.*, a configuration is a labeled graph, in which nodes are labeled (named) by groups of symbols, while edges are labeled by names of relations. Hypertexts [32] give an example of configurations in which nodes are labeled by texts. Other examples of configurations are discrete labeled graphs. Naturally, a configuration can represent a spin network. Making Kolmogorov algorithms work with spin networks allows them to describe dynamics of material particles and fields. Evolutionary Kolmogorov algorithms, *i.e.*, evolutionary **K**-machines where the class **K** consists of Kolmogorov algorithms, can naturally model evolution of space time and matter.

One more type of models that represent dynamics in labeled networks is *Petri nets* and their advanced version—*colored Petri nets* [33,34]. A Petri net $A$ is a system ($P$, $T$, **in**, **out**) that consists of

two sets—the set of *places P* and the set of *transitions T*—and two functions—the *input function* **in** that defines directed arcs from places to transitions and *output function* **out** that defines directed arcs from transitions to places. Places usually represent states or resources in the system, while transitions model the activities of the system. In such a way, Petri nets provide an efficient and mathematically rigorous modeling framework for discrete event concurrent dynamically systems. Interpreting Petri nets as spin networks allows them to model dynamics of material particles and fields. Consequently, evolutionary Petri nets, *i.e.*, evolutionary **K**-machines where the class **K** consists of Petri nets, can naturally depict evolution of space time and matter.

The third type of models that represent dynamics in labeled networks comprises Turing machines with a structured memory, inductive Turing machines with a structured memory and limit Turing machines with a structured memory [7]. Structured memory, which is a system of cells with a variety of connections between them, provides flexible tools for representation of spin networks, while machines that work with structured memory can reflect dynamics of spin networks, and thus, portray dynamics of material particles and fields. Consequently, evolutionary Turing machines with a structured memory, evolutionary inductive Turing machines with a structured memory and evolutionary limit Turing machines with a structured memory can naturally model evolution of space time and matter.

Note that Petri net modeling of spin network dynamics is different from the realization of these processes by (evolutionary) Kolmogorov algorithms, (evolutionary) Turing machines with a structured memory, (evolutionary) inductive Turing machines with a structured memory and (evolutionary) limit Turing machines with a structured memory. While a computational machine or a Kolmogorov algorithm has a unified system of rules for changing spin networks, Petri nets work guided by local rules. These local rules represent local dynamics of spin networks. However, it is possible to show that (evolutionary) Turing machines with a structured memory can model both (evolutionary) Petri nets and (evolutionary) Kolmogorov algorithms [7], *i.e.*, local rules can be represented by a system of context dependent global rules. At the same time, the structured memory allows using local rules of transformation, which depend on the processed area in the structured memory.

## 7. Conclusions

Using universal evolutionary machines/automata, we introduced and studied several measures of evolutionary information, *i.e.*, information used in evolutionary processes. These measures include relative and absolute evolutionary information sizes and evolutionary information in one object about another object. Conditions were found when these measures are optimal and invariant.

This brings us to the following problems:

**Problem 1.** Study relations between evolutionary information and quantum information.

Evolution is very important in biology, being directed by genetic processes. This gives us a natural direction for evolutionary information theory application.

**Problem 2.** Study biological information in general and genetic information in particular from the evolutionary information perspective.

The role of social evolution in society makes principal the following problem.

**Problem 3.** Explore social processes by means of evolutionary information.

The general theory of information [8] provides a universal context for evolutionary information theory, supplying necessary tools and structures for efficient research. Thus, it would be interesting to examine the following directions.

**Problem 4.** Study evolutionary processes in spaces of information operators.

Spaces of information operators are built and studied based on functional analysis. Another mathematical model for the general theory of information utilizes category theory. Thus, we have a similar problem in the context of algebraic categories.

**Problem 5.** Study evolutionary processes in categories of information operators.

Evolutionary computations in general and genetic algorithms in particular are efficient tools for optimization [27,28].

**Problem 6.** Study complexity of optimization problems using evolutionary information theory.

Evolutionary automata are utilized for modeling and exploration of collaboration processes [35].

**Problem 7.** Apply evolutionary information theory to find how collaboration can decrease complexity in solving various problems.

In this paper, the main concern was evolutionary information for classes of evolutionary automata/algorithms that have universal automata/algorithms.

**Problem 8.** Study properties of evolutionary information for classes of evolutionary automata/algorithms that do not have universal automata/algorithms.

Evolutionary computations have been used simulation of communication and language evolution [36,37]. Thus, it would be natural to find information characteristics of these processes.

**Problem 9.** Study communication and language evolution using evolutionary information theory.

It is possible to put this problem in a more general context.

**Problem 10.** Study cultural evolution using evolutionary information theory.

## References

1. Garzon, M.H. Evolutionary computation and the processes of life: On the role of evolutionary models in computing. In Proceeding of Ubiquity Symposium, November 2012.
2. Burgin, M.; Eberbach, E. Universality for Turing machines, inductive Turing machines and evolutionary algorithms. *Fundam. Inform.* **2009**, *91*, 53–77.

3. Burgin, M.; Eberbach, E. On foundations of evolutionary computation: An evolutionary automata approach. In *Handbook of Research on Artificial Immune Systems and Natural Computing: Applying Complex Adaptive Technologies*; Mo, H., Ed.; IGI Global: Hershey, PA, USA, 2009; pp. 342–360.
4. Burgin, M.; Eberbach, E. Bounded and periodic evolutionary machines. In Proceeding of 2010 Congress on Evolutionary Computation (CEC′2010), Barcelona, Spain, 2010; pp. 1379–1386.
5. Chaitin, G.J. *Algorithmic Information Theory*; Cambridge University Press: Cambridge, UK, 1987.
6. Li, M.; Vitanyi, P. *An Introduction to Kolmogorov Complexity and its Applications*; Springer-Verlag: New York, NY, USA, 1997.
7. Burgin, M. *Superrecursive Algorithms*; Springer: New York, NY, USA, 2005.
8. Burgin, M. *Theory of Information*: *Fundamentality, Diversity and Unification*; World Scientific Publishing: Singapore, 2010.
9. Dodig-Crnkovic, G. Significance of models of computation from Turing model to natural computation. *Minds Mach.* **2011**, *21*, 301–322.
10. Dodig-Crnkovic, G.; Müller, V. A dialogue concerning two world systems: Info-computational *vs.* mechanistic. In *Information and Computation*; Dodig-Crnkovic, G., Burgin, M., Eds.; World Scientific: Singapore, 2011; pp. 149–184.
11. Fredkin, E. Digital mechanics. *Physica D* **1990**, *45*, 254–270.
12. Lloyd, S. A theory of quantum gravity based on quantum computation. Available online: http://arxiv.org/abs/quant-ph/0501135 (accessed on 12 March 2013).
13. Wolfram, S.A. *New Kind of Science*; Wolfram Media: Champaign, IL, USA, 2002.
14. Zuse, K. *Rechnender Raum*; Friedrich Vieweg & Sohn: Braunschweig, Germany, 1969.
15. Smolin, L. *Three Roads to Quantum Gravity*; Basic Books: New York, NY, USA, 2001.
16. Smolin, L. Atoms of space and time. *Sci. Am.* **2004**, *15*, 66–75.
17. Rovelli, C. *Quantum Gravity*; Cambridge University Press: Cambridge, UK, 2004.
18. Burgin, M. Multiple computations and Kolmogorov complexity for such processes. *Not. Acad. Sci. USSR* **1983**, *27*, 793–797.
19. Burgin, M. Algorithmic complexity of recursive and inductive algorithms. *Theor. Comput. Sci.* **2004**, *317*, 31–60.
20. Burgin, M. *Measuring Power of Algorithms*, *Computer Programs*, *and Information Automata*; Nova Science Publishers: New York, NY, USA, 2010.
21. Burks, A.; Wright, J. The theory of logical nets. *Proc. Inst. Radio Eng.* **1953**, *41*, 1357–1365.
22. Blum, L.; Cucker, F.; Shub, M.; Smale, S. *Complexity and Real Computation*; Springer-Verlag: New York, NY, USA, 1997.
23. Burgin, M. Universal limit Turing machines. *Not. Rus. Acad. Sci.* **1992**, *325*, 654–658.
24. Wright, B.D. Negative information. *Rasch Meas. Trans.* **1996**, *10*, 504.
25. Horodecki, M.; Oppenheim, J.; Winter, A. Partial quantum information. *Nature* **2005**, 436, 673–676.
26. Hintikka, J. *Logic, Language Games and Information*; Clarendon: Oxford, UK, 1973.
27. Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*; Springer-Verlag: New York, NY, USA, 1996.
28. Michalewicz, Z.; Fogel, D.B. *How to Solve It*: *Modern Heuristics*, 2nd ed.; Springer-Verlag: New York, NY, USA, 2004.

29. Burgin, M. *Theory of Named Sets*; Nova Science Publisher: New York, NY, USA, 2011.
30. Gibbs, W.W. Ripples in spacetime. *Sci. Am.* **2002**, *286*, 62–71.
31. Kolmogorov, A.N. On the concept of algorithm. *Rus. Math. Surv.* **1953**, *8*, 175–176.
32. Nielsen, J. *Hypertext and Hypermedia*; Academic Press: New York, NY, USA, 1990.
33. Peterson, J.L. *Petri Net Theory and the Modeling of Systems*; Prentice-Hall, Inc.: Englewood Cliffs, NJ, USA, 1981.
34. Petri, C. Kommunikation mit Automaten. Ph.D. Thesis, University of Bonn, Bonn, Germany, 1962.
35. Burgin, M.; Eberbach, E. Cooperative combinatorial optimization: Evolutionary computation case study. *BioSystems* **2008**, *91*, 34–50.
36. Christiansen, M.H.; Kirby, S. Language evolution: Consensus and controversies. *Trends Cogn. Sci.* **2003**, *7*, 300–307.
37. Nowak, M.A.; Krakauer, D.C. The evolution of language. *Proc. Natl. Acad. Sci. USA* **1999**, *96*, 8028–8033.

© 2013 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).

*Information* **2014**, *5*, 357-388; doi:10.3390/info5020357

OPEN ACCESS

*information*

**ISSN 2078-2489**

www.mdpi.com/journal/information

*Article*

# Weighted E-Spaces and Epistemic Information Operators

**Mark Burgin**

Department of Mathematics, University of California, 405 Hilgard Ave., Los Angeles, CA 90095, USA; E-Mail: markburg@cs.ucla.edu; Tel.: +1-310 251 4897

*Received: 19 July 2013; in revised form: 22 May 2014 / Accepted: 22 May 2014 / Published: 11 June 2014*

---

**Abstract:** Information is usually related to knowledge. Here, we present a broader picture in which information is associated with epistemic structures, which form cognitive infological systems as basic recipients and creators of cognitive information. Infological systems are modeled by epistemic spaces, while operators in these spaces are mathematical models of information. Information that acts on epistemic structures is called cognitive information, while information that acts on knowledge structures is called epistemic information. The latter brings new and updates existing knowledge, being of primary importance to people. In this paper, both types of information are studied as operators in epistemic spaces based on the general theory of information. As a synthetic approach, which reveals the essence of information, organizing and encompassing all main directions in information theory, the general theory of information provides efficient means for such a study. Different types of information dynamics representation use tools from various mathematical disciplines, such as the theory of categories, functional analysis, mathematical logic and algebra. In this paper, we base our exploration of information and knowledge dynamics on functional analysis further developing the mathematical stratum of the general theory of information.

**Keywords:** information; knowledge; epistemic structure; epistemic space; weighted epistemic space; epistemic information operator; vector bundle; continuity; boundedness

---

## 1. Introduction

Mathematical models play the pivotal role in science. In information theory mathematical models have been mostly used for efficient and constructive representation and study of information measures. The most popular mathematical information measures are Shannon's entropy, Kolmogorov complexity

and Fisher information (measure). At the same time, the most advanced approach to modeling information and its measures is developed in the general theory of information, which systematizes and encompasses all main directions in information theory [1].

In this paper, we further develop the mathematical stratum of the general theory of information. According to the main principles of the general theory of information, information causes changes in infological systems. Thus, it is natural to model information by epistemic information operators that change mathematical representation of infological systems. Here, we are concerned with the cognitive information, which acts on cognitive infological systems, which are modeled by epistemic information spaces. They comprise symbolic representations of epistemic structures, which are basic units in cognition. Examples of epistemic structures employed by cognitive processes are concepts, notions, statements, ideas, images, opinions, texts, beliefs, knowledge, values, measures, problems, schemas, procedures, tasks, goals, *etc.* Epistemic structures form epistemic spaces.

Here, we construct and utilize weighted epistemic spaces, which represent, not only epistemic structures, but also their characteristics. The mathematical structure used for representing weighted epistemic spaces in the formal context is called a generalized vector bundle [2]. Informally, it consists of epistemic elements connected by relations—the base of the vector bundle-and a vector space attached to each of these elements. Note that as an epistemic space is a set of epistemic structures or their representations with relations and operations, it encompasses many other mathematical structures, such as lattices, groups or partially ordered sets.

Epistemic information operators are transformations and mappings of epistemic spaces. A special case of epistemic spaces—knowledge spaces—and knowledge information operators were studied in [1,3,4]. Here, we study knowledge spaces and information operators in a more general context of epistemic structures, epistemic spaces, and epistemic information operators. In addition, we extend epistemic spaces to weighted epistemic spaces, which include weighted knowledge spaces, and study properties of information operators in these spaces. To do this, we use structures and methods from functional analysis.

Note that there are information operators that are not epistemic. Examples of such operators are emotional and instructional information operators [1]. They are studied elsewhere as here we explore only epistemic information operators.

This paper is organized in the following way. In Section 2, we define epistemic structures and study their properties. In Section 3, we introduce epistemic spaces and study their properties. In Section 4, we show how epistemic structures acquire weights turning epistemic spaces into weighted epistemic spaces. Mathematical models of epistemic and cognitive information in the form of epistemic information operators acting in epistemic and weighted epistemic spaces are studied in Section 5. In Conclusion, some open problems related to epistemic and cognitive information are given.

## 2. Epistemic Structures in the Context of Information

Information is closely related to transformations [5]. Cognitive information is associated with transformations of epistemic structures [1]. Knowledge is a kind of epistemic structures. Thus, it is natural to treat knowledge in the context of epistemic structures.

**Definition 1.** An *epistemic structure* is a basic structure of cognition.

It is possible to find different definitions of *structure* in general and a unifying approach to this concept in [6].

Note that although according to the conventional understanding, a domain usually comprises several objects one object can be also treated as the domain of an epistemic structure. In essence, a domain is any part of reality, while reality as a whole includes all three types—physical reality, mental reality and structural reality [6]. Actually it is possible to consider any object, e.g., a system or a process, as a domain.

The essence of epistemic structures is represented by the following diagrams in Figures 1 and 2:

**Figure 1.** A reflection epistemic triad (direct epistemic unit *U*).

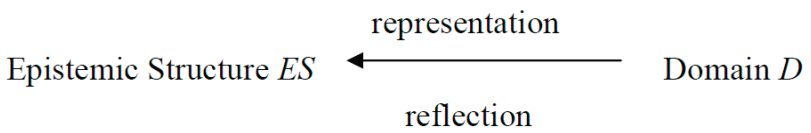
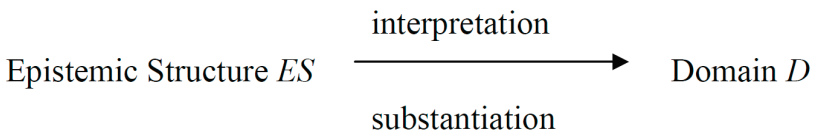



**Figure 2.** A substantiation epistemic triad (inverse epistemic unit *V*).




Both pivotal cognitive and behavioral concepts—beliefs and knowledge—are basic epistemic structures. At the same time, there are many other epistemic structures, such as concepts, notions, statements, questions, ideas, images, algorithms, tasks, procedures, problems, values, measures, opinions, and goals. Note that rational behavior of people is essentially based on their beliefs and knowledge.

Definition 1 is essentially informal. A formalized definition of epistemic structures is recursive as it is constructed based on recursion.

Epistemic triads (Figures 1 and 2) describe and represent extended epistemic units, each of which reflects a definite domain or an aspect of definite domain.

**Definition 2.** (a) A reflection epistemic triad (cf. Figure 1) describes a *direct extended epistemic unit*, while a substantiation epistemic triad (cf. Figure 2) describes an *inverse extended epistemic unit*.
(b) The epistemic structure *ES* of an epistemic triad *U* (or *V*) is called the *cognitive part* of the epistemic unit *U* (unit *V*) or simply, an *epistemic unit*.
(c) The domain *D* of the epistemic triad *U* (or *V*) is called the *substantial part* of the extended epistemic unit *U* (unit *V*).

Usually epistemic structures are represented by systems of symbols, which are called symbolic epistemic structures. *Symbolic knowledge units* are examples of symbolic epistemic units. For instance, the sentence "Ten is larger than five" is a symbolic knowledge unit, as well as a symbolic epistemic unit. The logical expressions, such as $\varphi \vee \psi$, $\varphi \wedge \psi$, and $\varphi \rightarrow \psi$, are also symbolic knowledge units, as well as symbolic epistemic units.

In general, epistemic structures are components of extended epistemic units.

In what follows, we consider only symbolic epistemic units and symbolic knowledge units. That is why for simplicity, symbolic epistemic units are called *epistemic units* and symbolic knowledge units are called *knowledge units*. In this, we follow the longstanding tradition where knowledge means only symbolic representation of knowledge.

It is natural to separate the domain of all epistemic structures into two classes (types)—static and dynamic epistemic structures.

**Definition 3.** *Static epistemic structures* describe domains that are not changing.

**Definition 4.** *Dynamic epistemic structures* either describe domains that are changing or have changes as their domain.

For instance, if a system is described as a collection of elements and relations between these elements, this description is a static epistemic structure. We obtain a dynamic epistemic structure when the processes going on in the system are also included in the description. An algorithm or a procedure is another example of a dynamic epistemic structure, the domain of which is a process or a system of processes.

We also separate dynamic epistemic structures into two classes (types)—functional epistemic structures and process epistemic structures.

**Definition 5.** *Functional epistemic structures* represent changes as transitions from the initial state to the final state.

For instance, operations, relations and goals are functional epistemic structures.

**Definition 6.** *Process epistemic structures* represent changes as processes.

For instance, algorithms, flowcharts, scenarios, inferences and stories are process epistemic structures. In these examples, algorithms are compressed epistemic structures, inferences and stories are expanded epistemic structures, while flowcharts and scenarios can be either compressed epistemic structures or expanded epistemic structures.

*Epistemic item* is a more general concept than *epistemic unit* because an epistemic item can be a part of an epistemic unit or consists of several epistemic units. In particular, *knowledge item* is a more general concept than *knowledge unit* because a knowledge item can be a part of a knowledge unit or consists of several knowledge units. For instance, taking a knowledge base, it is possible to consider all knowledge from this knowledge base as one knowledge item. However, usually this knowledge consists of many knowledge units. All knowledge from a textbook is a knowledge item.

Epistemic structures form the base for the traditional interpretation of information as the essence that gives or changes knowledge. According to the general theory of information, symbolic epistemic items (structures) constitute cognitive infological systems [1].

This understanding is formalized in the general theory of information based on the basic ontological principle O2c [1].

**Ontological Principle O2c** (**the *Cognitive Transformation Principle***). *Cognitive information* for a system *R* is a (potential) capacity to cause changes in the cognitive infological system CIF(*R*) of the system *R*.

In this context, a *cognitive infological system* CIF(*R*) contains, acquires, stores and processes various epistemic structures, such as knowledge, data, ideas, beliefs, images, algorithms, tasks, procedures, problems, schemas, scenarios, values, measures, opinions, goals, ideals, fantasies, abstractions, *etc.* Cognitive infological systems are very important, especially, for intelligent systems. Indeed, the majority of researchers believe that information in general is intrinsically connected to cognition, while cognitive information is one of the three basic types of anthropic information studied in [3]. Moreover, some researchers believe that people's knowledge about physical reality is the result of information they obtain from external sources [7–10]. Understanding that physicists study physical systems not directly but only through information they get from these systems has created a school of thought about the role of information processing in physical processes and its influence on physical theories. According to one of the outstanding physicists of the 20th century, John Archibald Wheeler (1911–2008), it means that every physical quantity derives its ultimate significance from information. He called this idea "It from Bit" where It stands for things, while Bit impersonates information as the most popular information unit [10]. For Wheeler and his followers, space-time itself must be understood and described in terms of a more fundamental pregeometry without dimensions and classical causality. These features of the physical world only appear as emergent properties in the ideal modeling the physical reality based on information about complex interactions of very simple basic elements, such as subatomic particles.

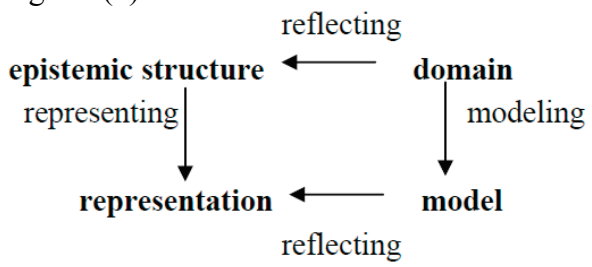
As the cognitive infological system contains knowledge of the system it belongs to, cognitive information is the main source of knowledge changes. When a system (it may be a person, a group of people, a community, society as whole or an intelligent agent) receives cognitive information, this system may convert it to knowledge or miss this information. For instance, a teacher is not giving knowledge to his students but only provides cognitive information and students themselves have to convert this information into knowledge [11].

It is important to discern epistemic structures and their representations, as well as knowledge and its representations. Epistemic structures are usually represented by symbolic systems in general and symbols, in particular. As a rule, one epistemic structure (knowledge unit) has several representations. For instance, knowledge that a person *A* is seven feet tall can be represented by: the statement/sentence/proposition "*A* is seven feet tall"; the statement/sentence/proposition "the height of *A* is seven feet"; the equality $H(A) = 7$ ft where $H(X)$ is the property *height* of a person; the truth of the predicate $H(A, 7)$ where $H(X, h)$ is the predicate with *X* being a person and *h* being a number of feet; and the element $(A; 7)$ of a relational database.

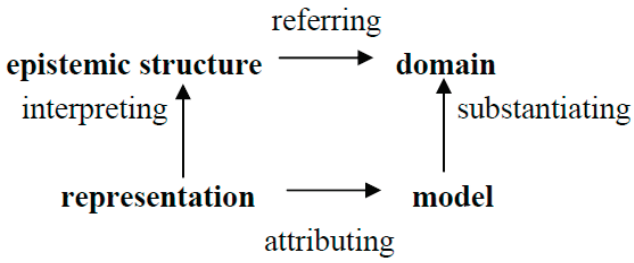
At the same time, the same symbolic system can represent different epistemic structures. For instance, a statement can represent knowledge, a belief or an idea.

A symbolic representation of an epistemic structure (cognitive epistemic unit) is called a *symbolic epistemic structure* (*symbolic epistemic unit*). Thus, it is possible to discern structural cognitive infological systems, which consist of epistemic structures, and symbolic infological systems, which consist of symbolic representations of epistemic structures.

In a similar way, domains are usually represented by their models. This brings us to the *extended direct epistemic unit* characterized by Diagram (1) and *extended inverse epistemic unit* described by Diagram (2).




(1)




(2)

Very often people make no distinction between epistemic structures and their representations. For instance, people assume that the statement "The Earth rotates around the Sun" is knowledge. However, it is only some representation of knowledge about the Earth and the Sun. It is possible to represent the same knowledge by the statement "The Earth moves about the Sun" or by the statement "The Earth travels about the Sun". This shows that one epistemic structure can have and usually has several symbolic, e.g., linguistic, representations.

## 3. Epistemic Spaces

Cognitive systems, e.g., intelligent agents or cognitive actors, store and employ a variety of epistemic structures in different forms and shapes. The memory of a cognitive system contains epistemic structures, which are organized in the system called an epistemic space.

Informally, an epistemic space is a set of epistemic structures or their representations with relations and operations. Epistemic structures (cognitive epistemic units) form *abstract epistemic spaces*, while their representations (symbolic epistemic units) create *symbolic epistemic spaces*. However, in what follows, we consider only symbolic epistemic spaces.

In modeling epistemic systems in general and knowledge systems in particular and studying information processes in these systems, we consider two basic structures—sets and multisets. Sets of epistemic structures enhanced by relations between these structures become *epistemic spaces*, while multisets of epistemic structures enhanced by relations between these structures become *epistemic multispaces*. In essence, using the classical mathematical background it is possible to consider only sets, which make the model simpler serving as the first approximation to real epistemic/knowledge systems and information processes. However, many real cognitive systems contain several copies of the same element. For instance, the same element of knowledge can be stored in different parts of a computer memory or of the brain. This makes utilization of multisets necessary. We remind that a *multiset* is a collection that is like a set but can include identical or indistinguishable elements. For instance, $M = \{a, a, b, b, b\}$ is a multiset that contains two elements $a$ and three elements $b$. It is usually assumed that in a multiset, elements are indistinguishable if and only if they have the same type.

If $a$ is an element from a multiset $M$, then the number of copies of an element $a$ is called the *multiplicity* of $a$ in $M$ and is denoted by $m_M(a)$. In the considered example, $m_M(a) = 2$ and $m_M(b) = 3$.

A multiset $M$ is a *multisubset* of a multiset $N$ if $m_M(a) \leq m_N(a)$ for all elements $a$ from $M$.

However, when for building epistemic spaces, we use not only sets but structures, *i.e.*, sets with relations, we are able to include multisets in this schema because multisets can be treated as sets with the indistinguishability relation. That is why we use only sets and structures forming epistemic spaces from epistemic structures.

To give an exact definition of an abstract epistemic space, we consider a set $\boldsymbol{W}_{\mathbf{es}}$ of epistemic structures, while for an exact definition of an symbolic epistemic space, we consider a set $W_{\text{ses}}$ of symbolic epistemic units, e.g., of symbolic knowledge units or knowledge items, taking it as the *state base* of an epistemic space. For instance, it is possible to regard a set $W_K$ of symbolic parts of elementary knowledge units as a set $W_{\text{es}}$. This allows us obtaining an efficient formalization of the concept of a knowledge space. The set $W_L$ of propositions and/or predicates in a logical language $L$ gives an example of $W_{\text{ses.}}$ Propositions and predicates are symbolic knowledge units in the logical approach to information theory developed in works of Bar-Hillel and Carnap [12], Hintikka [13,14] and some other authors. Shreider [15] interpreted symbolic knowledge units as texts in a thesaurus. Many researchers employ mental schemas as cognitive/symbolic knowledge units in the brain (cf., for example, [16–18]) One more possibility for $W_{\text{ses}}$ is the set, or more exactly, a multiset, $W_{Sit}$ of logical representations of situations possible in a world $U$. Situations themselves form the substantial parts of knowledge units, while their logical descriptions, e.g., in the form of propositions, are the cognitive/symbolic parts of knowledge units. These logical descriptions are also called situations playing the role of knowledge items or knowledge units (cf., for example, [19]).

Some readers may be confused by variability of epistemic structures. However, mathematics provides means to decrease non-uniformity in the diversity of epistemic structures. Let us assume that we study such diversity as a collection $B$ of epistemic structures by means of epistemic spaces. One way to reduce complexity is to use homogeneous approximations and, at first, to study only uniform epistemic spaces, which model uniform collections of epistemic structures. For instance, a knowledge space is a uniform collection of epistemic structures or algorithmic space is a uniform collection of epistemic structures.

On the second step, it is possible to introduce and study epistemic spaces with a low degree of uniformity (A formal definition of degrees of uniformity based on measures of uniformity of general systems is given in [20,21]). For instance, the degree of uniformity of a knowledge space is lower than the degree of uniformity of a space of knowledge, beliefs and ideas. At the same time, logical knowledge space has higher degree of uniformity than the predicate knowledge space

Another way to deal with extremely non-uniform systems is unification. Here, we describe unification of collections of epistemic structures. To do this, we observe that it is possible to represent each epistemic structure by a concept. For instance, taking a knowledge item "Now it is 5 p.m.", we express the content of this statement by the equality relation between two temporal points (intervals) 5 p.m. and "now". This relation is a concept. The meaning of this concept is expressed by (consists of) its relations to other concepts, such as "time", "equality", "identity", "point", *etc.* In such a way, unification converts any collection of epistemic structures into a uniform system, which consists only of concepts. It is possible to model this system by a semantic network or by a more advanced epistemic space.

In the theory of epistemic spaces, often it is possible not to distinguish sets $W_{\text{es}}$ and $W_{\text{ses}}$ because they have many common properties and only these properties are important in many theoretical

constructions. When this is the case, we denote $W_{es}$ or $W_{ses}$ by the same letter $W$ without causing confusion.

Some sets $W$ can reflect (represent) more objects (large domains), while others reflect (represent) less objects (small domains).

**Definition 7.** We call the set $W$ *universal* for a collection **CIF** of cognitive infological systems, e.g., for systems of knowledge, when the following axiom is true.

**UCIF1 (the Internal Representation Axiom)**. For any infological system $R$ from **CIF**, any state of $R$ is a subset of the set $W$.

For instance, we can take a group $G$ of intelligent agents and define the collection **CIF** as the set of their knowledge systems playing the role of cognitive infological systems from **CIF**. Then the Internal Representation Axiom states that any possible state $K_{Ai}$ of the knowledge system $K_A$ of an agent $A$ from $G$ is a subset of the set $W$. In this case, it is possible to interpret $W$ as the base of all knowledge that agents are able to have about their environment.

Another aspect of universality of the set $W$ is expressed by the possibility to describe all possible (existing) worlds or/and situations utilizing epistemic elements (symbolic epistemic elements), e.g., knowledge (symbolic knowledge units), only from $W$. For instance, when $W$ is the set $W_L$ of propositions and/or predicates from some logical language $L$, then universality implies that it is possible to build all descriptions of all possible worlds by combining elements from $W_L$. This possibility is reflected in the following concept.

Let us consider a domain, $D$. This domain may be a part of the real world, the set of all (possible) situations in a part of the real world, the set of all possible (existing) worlds in the sense of logical semantics or the set of all possible (existing) states of the environment in some area.

**Definition 8.** We call the set $W$ *universal* for the domain $D$ when the following axiom is true.

**UCEF2 (the External Representation Axiom).** For any environment (situation, world or state) $R$ of the domain $D$, there is a subset $W_R$ of the set $W$ that contains all epistemic structures that reflect $R$.

In particular, it means that if $W_{es}$ consists only of knowledge structures, then for any environment (situation, world or state) $R$ of the domain $D$, there is a subset $W_R$ of the set $W_{es}$ that contains all (accessible or representable) knowledge about $R$.

Taking axioms UCIF1 and UCEF2 as the foundation, we develop a theory of cognitive systems (cognitive agents) called the *theory of E-spaces*. At first, we define free epistemic spaces, taking $W_{es}$ for abstract epistemic spaces and $W_{ses}$ for symbolic epistemic spaces.

Usually an epistemic space $E$ is a dynamic system, which is permanently changing its content. This content at a given moment of time is called the state of the epistemic space $E$.

**Definition 9.** (a) An *epistemic space*, also called an *E-space*, $V$ with the base $W$ is a subset of the set $W$.
(b) Appropriate subsets of $V$ are called *states of the epistemic space V*.
(c) An epistemic space $V$ with a collection $\mathrm{St}V \subseteq 2^V$ is called an *epistemic system*.

For instance, we can take the set *Id* of all ideas about which Professor Angstrem can think as an epistemic space. However, at any moment of time, he can think only about one or two ideas. Thus, modeling his thinking, only one-element and two-element subsets of *Id* will be appropriate as states of this epistemic space. These states will represent the projection of the Professor Angstrem's mentality on the space of ideas.

Note that that any state of an epistemic space is itself an epistemic space.

It is natural to consider epistemic spaces that have elements of the same type. For instance, free epistemic spaces that consist of symbolic knowledge units are called *knowledge spaces* or *Mizzaro spaces*. They are studied in [1,4].

**Definition 10.** (a) A *type of structures* is a system of conditions (axioms) that all these structures, *i.e.,* sets with relations, satisfy.
(b) An epistemic space $V$ is called *uniform* if all its elements have the same type.
(c) An epistemic system ($V$, St$V$) in which all states are uniform is called *uniform*.

For instance, the epistemic space of all logical propositions is uniform. Algorithms are knowledge items explaining how to solve a problem or how a system is functioning. Finite automata form a uniform class of algorithms, which is a uniform epistemic space. However, a class of algorithms that consists of finite automata and Turing machines is an epistemic space that is not uniform.

In general, epistemic multispaces spaces have different relations between and operations with their elements and states. Thus, let us assume that $W$ is not simply a set or a multiset but is a structure, *i.e.*, a set with relations and operations [6].

**Definition 11.** (a) A *structured epistemic space*, also called a *structured E-space*, $V$ with the base $W$ is a substructure of the structure $W$.
(b) A substructure of V is called a state of the structured epistemic space (multispace) V.

For instance, stratified M-spaces [3] give an example of structured epistemic spaces.

Let us consider other examples of epistemic spaces. Usually these epistemic spaces are structured.

**Example 1.** A propositional epistemic space *EPS*, in which propositions are epistemic elements, is a knowledge space as propositions constitute one of the basic forms of knowledge representation. There are many relations in this space, *i.e.*, it is a structured epistemic space. One of the main relations is implication denoted by the symbol →, where $p \to q$ means that whenever the proposition $p$ is true, the proposition $q$ is also true. Other important relations in the propositional epistemic space *EPS*:

The *deducibility relation* means "a proposition $p$ is deduced from a proposition $q$", for example, the proposition $q$ is deducible from the proposition $q$ & $p$. This is the most popular logical relation between propositions from a propositional language. The deducibility relation is usually denoted by the symbol ⊢, e.g., $A \vdash \varphi$.

The *generality relation* means that one proposition is more general than another one, for example, the proposition "stars give light" is more general than the proposition "stars from our galaxy give light".

Another important logical relation between propositions or statements is "a proposition (statement) $p$ directly implies a proposition (statement) $q$", *i.e.*, there is an inference rule, e.g., *modus ponens* or *modus tollens*, such that $p$ is the argument and $q$ is the conclusion.

One more important logical relation between propositions or statements is "a proposition (statement) $p$ *entails* a proposition (statement) $q$ (in a theory $T$)", meaning that in every model (of $T$ ) where $p$ is true, $q$ is also true. The *entailment relation* is usually denoted by the symbol $\vDash$, e.g., $p \vDash q$.

There are also operations in the propositional epistemic space *EPS*, for example, classical logical operations—conjunction, disjunction, and negation.

It is also possible to induce topology in the propositional epistemic space *EPS*.

**Example 2.** We can build a *conceptual epistemic space* based on formal concept analysis [22]. A conceptual epistemic space consists of formal contexts, concepts, concept intents, and concept extents, which are defined as rigorous mathematical structures.

A *formal context* **C** is a triad (named set) **C** = ($G$, $I$, $M$), where

- $G$ is a set of objects,
- $M$ is a set of attributes,
- $I$ is a relation between $G$ and $M$, in which the relation $(g, m) \in I$ means, the object $g$ has the attribute $m$, *i.e.*, $I$ is the connection between objects and attributes.

Thus, we can see that formal contexts are named sets and it is possible to apply to them different named set operations [2]. There are also various relations between formal contexts that come from the named set theory. For instance, a formal context **C** = ($G$, $I$, $M$) is a *subcontext* of a formal context **D** = ($H$, $J$, $L$) if $G \subseteq H$, $M \subseteq L$ and $I$ is the restriction of $J$ on $G$ and $M$, *i.e.*, $I = J|_{(G, M)}$.

It is also possible to build an epistemic space from informal concepts. Taking concepts used in some society as symbolic epistemic elements, we obtain an epistemic space with different relations, *i.e.*, a structured epistemic space. For instance, it is natural to use the relation "to be a subconcept", e.g., the concept *dog* is a subconcept of the concept *animal*. Another relation between informal concepts is the *foundational relation*, which shows when one concept is used in a definition of another concept.

Semantic networks are examples of structured epistemic spaces built from informal concepts. *Semantic network* or *semantic net* is a knowledge representation formalism (graphic notation) that describes objects and their relationships in the form of a network consisting of labelled (named) nodes and (usually directed) links in the form of arcs or arrows. The nodes represent objects or concepts by their names, while the links represent relations between nodes also by their names.

Note that in a general case, epistemic spaces are graphs. So, conceptual epistemic spaces of Ganter and Wille [22] are special cases of general epistemic spaces.

**Example 3.** The epistemic scenarios together constitute an *epistemic space* in the sense of Chalmers [23]. The most natural way of scenario interpretations, at least initially, are possible worlds. More exactly, an epistemic scenario describes possible (in some sense) ways a world might be. Defining this, Chalmers uses the notion of possibility that is different from the notion of possibility usually associated with possible worlds. Here, possibility is a sort of epistemic possibility, whereas possible worlds are usually understood to be associated with a sort of "metaphysical" possibility.

Another way to describe epistemic scenarios is to identify them with equivalence classes of thoughts or with maximal classes of thoughts with equivalence defined as mutual implication. In this context, it is possible to assume that thoughts are composed of concepts.

Having defined a space of scenarios for each subject at a time, it is possible to form a common space of scenarios for all subjects. To do this, Chalmers uses the principle of *translation* between the maximal thoughts (or the complexes) of one subject, and those of another. This principle is based on the basic notion of translatability represented by the *translation relation*, which is an equivalence relation on the set of thoughts. As a result, we obtain a structured epistemic space (cf. Definition 11).

**Example 4.** A *knowledge information space M* is an important kind of epistemic information spaces. Knowledge information spaces are studied in [3].

**Example 5.** It is possible to represent a logical variety or a prevariety ***M*** [24] as a structured epistemic space $V$. In it, the components of ***M*** are treated as states of the space $V$, while the mappings $f_i$: $A_i \to L$ and $g_i : T_i \to L$ ($i \in I$), which form connections between components of the variety (prevariety), constitute relations between elements in this space. Thus, ***M*** is a structured epistemic space.

**Example 6.** Taking words from a natural language, such as English, Spanish, or German, as symbolic epistemic elements, it is natural to treat two words as linked when they express similar concepts. In such a way, Motter, *et al.* [25] built topology of the conceptual network of a language. This network gives one more example of structured epistemic spaces.

Given a set $X$ with a binary relation $R$, it is possible to introduce a metric in this space [26].

An *R-path* in $X$ between elements $x$ and $y$ is a sequence $p(x, y) = (x_1, x_2, x_3, \ldots, x_n)$ of elements from $X$ such that all pairs $(x_{i-1}, x_i)$ belong to $R$, $x_1 = x$ and $x_n = x$. The number $n$ is called the length of the path $p(x, y)$, *i.e.*, $l(p(x, y)) = n$. Then we define the distance by the following rule:

$$\mathrm{d_R}(x,y) = \begin{cases} \min\{l(p(x,y)); p(x,y) \text{ is path in } X \text{ between elements } x \text{ and } y\} & \text{if, at least, one path exists} \\ \infty & \text{if there are no paths in } X \text{ between elements } x \text{ and } y \\ 0 & \text{when } x = y \end{cases}$$

**Lemma 1.** The distance $\mathrm{d}_R(x, y)$ defines metric in the space $X$.

Indeed, by definition, $\mathrm{d}_R(x, y) = 0$ if and only if $x = y$. The relation $\mathrm{d}_R$ is symmetric, *i.e.*, $\mathrm{d}_R(x, y) = \mathrm{d}_R(y, x)$. In addition, if $(x_1, x_2, x_3, \ldots, x_n)$ is path in $X$ between elements $x$ and $z$ and $(y_1, y_2, y_3, \ldots, y_m)$ is path in $X$ between elements $y$ and $y$, then $(x_1, x_2, x_3, \ldots, x_n, y_1, y_2, y_3, \ldots, y_m)$ is path in $X$ between elements $x$ and $y$. Thus, $\mathrm{d}_R(x, y) \leq \mathrm{d}_R(x, z) + \mathrm{d}_R(z, y)$.

We will call this metric by the name the *relational metric* in the set $X$.

Note that in the relational metric, $X$ is a discrete topological space.

Lemma 1 shows that $W_{es}$ and $W_{ses}$ are metric spaces. Structures in the spaces $W_{es}$ and $W_{ses}$ are inherited by epistemic spaces and their states. Thus, as any subspace of a metric space is a metric space [26], an epistemic space $E$ and all its states of are metric spaces.

## 4. Weighted Epistemic Spaces

Epistemic structures and epistemic units have many properties. Basic characteristics of epistemic structures (epistemic units) in general and of knowledge in particular are called *dimensions*. Usually dimensions are gradual compound properties or attributes.

It is possible to discern the following dimensions of epistemic structures (of knowledge):

(1) The *relevance dimension* reflects relevance of an epistemic structure, e.g., of knowledge unit, to its domain.
(2) The *confidence* or *certainty dimension* reflects the relevance estimation of an epistemic structure, e.g., of knowledge unit, to its domain.
(3) The *justification dimension* reflects justification of the relevance estimation of an epistemic structure, e.g., of knowledge unit, to its domain.
(4) The *complexity dimension* reflects complexity of an epistemic structure, e.g., of knowledge unit, in utilization.
(5) The *significance dimension* reflects significance of an epistemic structure, e.g., of knowledge unit.
(6) The *efficiency dimension* reflects the role of an epistemic structure, e.g., of knowledge unit, in achieving some goals.
(7) The *reliability dimension* reflects reliability of an epistemic structure, e.g., of knowledge unit.
(8) The *abstractness dimension* reflects the level of abstraction of an epistemic structure, e.g., of knowledge unit.
(9) The *generality dimension* reflects degree of generalization achieved by an epistemic structure, e.g., of knowledge unit.
(10) The *meaning dimension* reflects meaning of an epistemic structure, e.g., of knowledge unit.

The first three dimensions are the *separation dimensions* as these traits are often used to separate knowledge from other epistemic structures, e.g., from beliefs.

The next six dimensions are the *feature dimensions*.

The tenth dimension is the *integration dimension* as all other dimensions are projected into it.

Each dimension is a composite attribute comprising several basic epistemic attributes. For instance, the efficiency dimension of an epistemic structure *e* includes the efficiency of *e* for reaching some goal, e.g., for reaching the Mars, the efficiency of *e* for understanding *e*, the efficiency of *e* for understanding people, the efficiency of *e* for building some object *A* and the efficiency of *e* for obtaining knowledge about some object *D*.

Another example of efficiency is given by such an epistemic structure *int* as knowledge of mathematical integration and its efficiency for a student *C*. In this case, efficiency of *int* for getting a high grade in the class is rather high, while efficiency of *int* for getting from home to the college is rather low (usually it is zero).

Complexity comprises such properties as compression, understandability and hardness.

Taking the justification dimension, we see that there are different ways and strategies of epistemic structure (knowledge) justification. It is possible to treat each kind of justification as an attribute component of the justification dimension. At the same time, there are three basic approaches to justification, which are similar to the approaches to knowledge acquisition: by practice/experience, by reasoning/thinking and by authority/opinion.

*Justification by practice/experience* means that an epistemic structure (knowledge) is justified if it works well in practice, e.g., it allows better achieving some goals, and our experience gives evidence for this.

*Justification by reasoning/thinking* is performed in the mentality of the justifier and is explicit justification. However, the brain has three basic components-the *System of Rational Intelligence* (also called *System of Reasoning*), the *System of Emotions* (*Affective States*) and the *System of Will and Instinct* [1]. Consequently, there are two other kinds of mental justification—by emotions and by instructions/assertions.

*Justification by authority/opinion* means that an epistemic structure (knowledge) is justified if there is the corresponding opinion, which is usually held as an authoritative one. Note that it may be an opinion of an individual, of a social group taken from some source, such as a book, magazine, or the Internet.

In addition to attributes that constitute dimensions, there are other properties/attributes of epistemic structures and epistemic units. For instance, an important epistemic attribute is *novelty* with respect to the infological system of an intelligent agent (cognitive system). This attribute is a (fuzzy) function of another attribute that shows the time of attribution of epistemic structure to the given infological system.

Epistemic structures (knowledge units) described in two previous sections are pure. Properties/attributes of epistemic units (structures) and characteristics of objects they reflect (represent) induce weights of these epistemic units. For instance, taking such an epistemic unit as a statement $P$, we can consider its properties (attributes): (1) time when this statement $P$ was made; (2) person(s) who made this statement; (3) people who supported this statement; (4) time needed to prove validity (truthfulness) of $P$; (5) truth value of $P$; and so on. The value of the first attribute is the first weight $w_1$ of $P$. It is a numerical value. The value of the second attribute is the second weight $w_2$ of $P$. It is a nominal value, *i.e.*, it is a name or names of people who made statement $P$, or a functional value, *i.e.*, it is the indicator function of the names of people who made statement $P$. The value of the third attribute is the third weight $w_3$ of $P$. It is similar to the second weight. The value of the fourth attribute is the fourth weight $w_4$ of $P$. It is numerical and similar to the first weight. The value of the fifth attribute is the fifth weight $w_5$ of $P$ and it takes two values from the two-element set $\{true, false\}$ if we utilize the classical logic. If we evaluate $P$ by means of fuzzy logic, the fifth weight $w_5$ takes values in the interval [0, 1].

An algorithm is knowledge of how to solve a problem or a class of problems. Thus, taking an algorithm $A$ as a symbolic structure, we have such properties as (cf., [27]): (1) the length $l_A$ (weight $w_1$); (2) its time complexity $T_A$ (weight $w_2$); (3) its space complexity $S_A$ (weight $w_3$). The values of the first weight are positive numbers, while the values of the second and third weights are functions (cf., [27]).

Dimensions, which are basic complex properties, also add weights to knowledge units (epistemic structures). For instance, the knowledge unit $A$ represented by the sentence "Now it is ten o'clock in the morning" is the symbolic part of pure knowledge. However, it can be true or false depending on current time. This estimate defines the weight the knowledge unit in the relevance dimension. Namely, if the estimate "true" is represented by 1 and the estimate "false" is represented by 0, then the weight of $A$ is 1 when it is really ten o'clock in the morning and the weight of $A$ is 0 when this is wrong.

As a result, dimensions and other properties/attributes bring us from *pure epistemic structures* (knowledge units) to *weighted epistemic structures* (*weighted knowledge units*). To determine weights, we fix a vector of attributes $(A_1, \ldots, A_k)$. Then we change a pure epistemic structure (pure knowledge

unit) $e$, to the weighted epistemic structure (weighted knowledge unit) $B = (e; w_1, \ldots, w_k)$, where $w_i$ is the weight of $e$ with respect to the attributes $A_i$ (the dimension $i$). The value of the weight $w_i$ of the epistemic structure $e$ with respect to the attribute $A_i$ reflects to what extent $e$ has the attribute $A_i$. When the attributes $A_i$ is an abstract property in the sense of [2], then $w_i$ is the value of this property for the epistemic structure $e$.

It is possible to consider the system of weights $w_1, \ldots, w_k$ of a weighted epistemic structure (knowledge unit) $B = (e; w_1, \ldots, w_k)$ as the state of $e$.

It is interesting to know that weights can describe arbitrary structures in epistemic spaces. In particular, weights in an epistemic space can turn it into an epistemic multispace by inducing an indistinguishability relation between elements of this space. Indeed, in a multiset (cf. Section 3), elements are indistinguishable if and only if they have the same type. Thus, we can introduce the weight $w_t$ the values of which are types of elements (epistemic structures) from an epistemic space. Making elements that have the same value of the weight $w_t$ indistinguishable, we obtain an epistemic multispace.

An epistemic space $E$ is called *weighted* if all its elements are weighted epistemic structures. By construction, there is a natural projection $\pi$ of a weighted epistemic space $E$ onto a pure epistemic space $E_0$ where $\pi(e; w_1, \ldots, w_k) = e$ for any pure epistemic structure $e$.

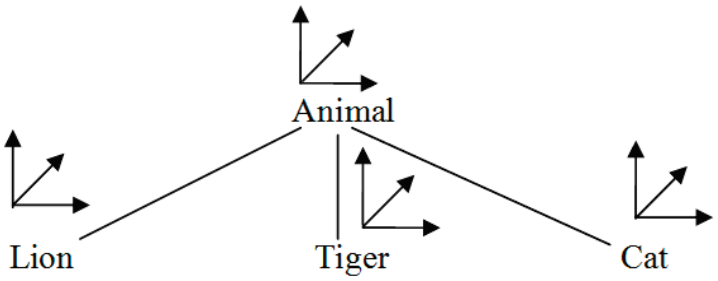
**Example 7.** Gardenfors offers his theory of conceptual representations as a bridge between the symbolic and connectionist approaches [28,29]. Symbolic representation is particularly weak at modeling concept learning, which is paramount for understanding many cognitive phenomena. Concept learning is closely tied to the notion of similarity, which is also poorly served by the symbolic approach. Gardenfors's theory of conceptual spaces presents a framework for representing information on the conceptual level. A conceptual space is built up from geometrical structures based on a number of quality dimensions. The main applications of the theory are on the constructive side of cognitive science: as a constructive model the theory can be applied to the development of artificial systems capable of solving cognitive tasks. Gardenfors also shows how conceptual spaces can serve as an explanatory framework for a number of empirical theories, in particular those concerning concept formation, induction, and semantics. His aim is to present a coherent research program that can be used as a basis for more detailed investigations.

**Example 8.** Osgood, Suci, and Tannenbaum [30] use semantic spaces for building their theory of meaning and its measurement. A *semantic space* is a set of concepts with their meaning. The *meaning of a concept to an individual subject* is defined as the set of the factor scores based on the data from this individual. The *meaning of a concept in the culture* is defined as the set of the averaged factor scores [30]. This shows that a semantic space is a special kind of weighted epistemic spaces. Although very often the factor scores are integers, it is possible to conjecture that using real numbers as the factor scores allows better evaluation of meaning turning the meaning of a concept into a real vector.

The mathematical structure used for representing weighted epistemic spaces in the formal context is called a generalized vector bundle [2]. Informally, it consists of epistemic elements connected by relations and a vector space attached to each of these elements. Below we give explicit examples of such spaces.

**Example 9.** We can also build a weighted conceptual epistemic space *WECS* in which concepts are epistemic elements. Below we give a sample of weighted conceptual epistemic spaces, which consists of four concepts, to each of which a three-dimensional weight space is attached. For instance, it is possible to construct each of these weight spaces using such properties as the level of abstraction, fuzziness and connectedness, *i.e.*, the number of other concepts to which this concept is connected.

**Figure 3.** A simple graphical example of a weighted conceptual epistemic space *WECS*.



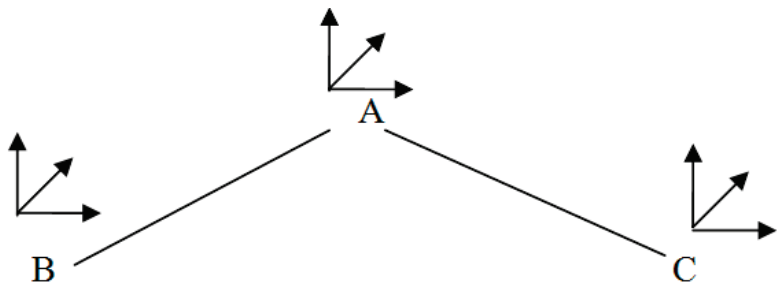

**Example 10.** A weighted propositional epistemic space *WEPS*, in which propositions are epistemic elements, is a knowledge space as propositions constitute one of the basic forms of knowledge representation. There are many relations in this space, *i.e.*, it is a structured epistemic space. One of the main relations is implication denoted by the symbol →, where $r$, $p \rightarrow q$ means that whenever the propositions $p$ and $r$ are true, the proposition $q$ is also true.

Below we give a sample of weighted conceptual epistemic spaces, which consists of three propositions:

A: ABCD is a square.
B: ABCD is a rectangle.
C: ABCD is a rhombus.

To each of these propositions, a three-dimensional weight space is attached. For instance, it is possible to construct each of these weight spaces using such properties as the level of abstraction, fuzziness and connectedness, *i.e.*, the number of other concepts to which this concept is connected.

**Figure 4.** A simple graphical example of a weighted conceptual epistemic space *WECS*.




In what follows, we assume that all weighted epistemic elements of the form $(a; w_1, \ldots, w_k)$ with the fixed $a$ is a real vector space—the space of weights, which is denoted by $L_a$, or even a topological vector space [31]. Note that in general, not all weights are numbers. For instance, there are functional weights. However, it is possible to immerse any domain of weight values into an appropriate vector

space and assume that the whole weight space is the vector space equal to the Cartesian product of weight spaces of individual weights $w$.

In addition, it is possible to assume that all vector spaces $L_a$ of weights have the same dimension. If it is not so, when dimensions of all $L_a$ are bounded, we can take the space $L_{a_0}$ of weights with the maximal dimension as the common space for all weights denoting it by $L_e$. In the case when dimensions of all $L_a$ are unbounded, we come to the necessity to use an infinite-dimensional vector space as the common space of all weights $L_e$. That is why, exploring the general situation, we acknowledge that the common vector spaces $L_e$ of weights is not necessarily finite dimensional.

In this context, the space $W_{esw}$ of weighted epistemic structures from $W_{es}$ has the structure of a vector bundle with the base $W_{es}$, *i.e.*, $\boldsymbol{W}_{esw} = (W_{esw}, \pi_{es}, W_{es})$ where $\pi_{es} : W_{esw} \to W_{es}$ is a projection, while the space $\boldsymbol{W}_{sesw}$ of weighted symbolic epistemic structures from $W_{ses}$ has the structure of a vector bundle with the base $W_{ses}$, *i.e.*, $\boldsymbol{W}_{sesw} = (W_{sesw}, \pi_{ses}, W_{ses})$ where $\pi_{ses} : W_{sesw} \to W_{ses}$ is a projection.

We remind [32] that a vector bundle **E** is a triad (named set) $\mathbf{E} = (E, p, B)$ where the topological space $E$ is called the *total space* or simply, *space* of the vector bundle **E**; the topological space $B$ is called the *base space* or simply, *base* of the vector bundle **E;** and $p$ is the topological projection of $E$ onto $B$ such that there is a vector space $F$ is called the *fiber* of the vector bundle **E**, for all points $b$ from $B$, $p^{-1}(b) = F_b \simeq F$ and every point in the base space has a neighborhood $U$ for which the space $p^{-1}(U)$ is homeomorphic to the direct product $U \times F$. In the case of the epistemic spaces $W_{esw}$ and $W_{sesw}$, $F$ is a vector space isomorphic the common vector space $L_e$ of weights.

Consequently, taking a weighted epistemic space $E \subseteq W_{esw}$, we obtain the vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ in which $p_E$ is the restriction of $\pi_{ses}$ on $E$ and $E_e = \pi_{ses}(E)$.

In general, we have the set $W_w$ of weighted epistemic structures and the vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ where $E \subseteq W_w$ and $E_e \subseteq W$.

Assuming that $E_e$ and the fiber $F$ of the vector bundle $\boldsymbol{E}$ are metric spaces with distances (metrics) d and $d_v$, correspondingly, we are able to define a distance **d** between elements $(e; w_1, \ldots, w_n)$ and $(l; u_1, \ldots, u_m)$ from the space $E$ in the following way

$$\mathbf{d}((e; w_1, \cdots, w_n), (l; u_1, \cdots, u_m)) = d_v((w_1, \cdots, w_n), (u_1, \cdots, u_m)) + d(e, l) \tag{1}$$

when $n = m$;

$$\mathbf{d}((e; w_1, \cdots, w_n), (l; u_1, \cdots, u_m)) = d_v((w_1, \cdots, w_n), (u_1, \cdots, u_m, 0, \cdots, 0)) + d(e, l) \tag{2}$$

when $n > m$; and

$$\mathbf{d}((e; w_1, \cdots, w_n), (l; u_1, \cdots, u_m)) = d_v((w_1, \cdots, w_n, 0, \cdots, 0), (u_1, \cdots, u_m)) + d(e, l) \tag{3}$$

when $n < m$.

In finite-dimensional vector spaces, we can take the Euclidean metric as $d_v$, defining the distance $d_v((x_1, \ldots, x_n), (y_1, \ldots, y_n))$.

However, as we discussed before, it is natural to assume that the fiber $F$ is an infinite-dimensional vector space. In this case, we simply postulate existence of a metric in it. Usually, metrics in vector spaces are defined by norms [33,34]. Note that in this case, we use only formula (1) because all fibers $F_a$ have the same dimension.

**Proposition 1.** The distance $d((e; w_1, \ldots, w_n), (l; u_1, \ldots, u_m))$ defines a metric in the space $W_w$.

**Proof.** By definition, $\mathbf{d}((e; w_1,\dots, w_n), (e; w_1,\dots, w_n)) = 0$. When $\mathbf{d}((e; w_1,\dots, w_n), (l; u_1,\dots, u_m)) = 0$, then $\mathrm{d}(e, l) = 0$ and, thus, $e = l$ because d is a metric in $W$. Besides, $\mathrm{d_v}((w_1, \dots, w_n), (u_1, \dots, u_m)) = 0$ and thus, $(w_1,\dots, w_n) = (u_1,\dots, u_m)$ because $\mathrm{d_v}$ is a metric in a vector space. Consequently, $\mathrm{d}((e; w_1,\dots, w_n), (l; u_1,\dots, u_m)) = 0$ if and only if $(e; w_1,\dots, w_n) = (l; u_1,\dots, u_m)$.

The function **d** is symmetric because the function d is symmetric in $W$, while the function $\mathrm{d_v}$ is symmetric in a vector space.

In addition, let us take arbitrary weighted (symbolic) epistemic structures $(e; w_1,\dots, w_n)$, $(l; u_1,\dots, u_m)$ and $(h; v_1,\dots, v_p)$ from $W$ and denote $\mathrm{d}(e, l) = a$, $\mathrm{d}(l, h) = b$, $\mathrm{d}(e, h) = c$, $\mathrm{d_v}((w_1,\dots, w_n), (u_1,\dots, u_m)) = d$, $\mathrm{d_v}((u_1,\dots, u_m), (v_1,\dots, v_p)) = k$ and $\mathrm{d_v}((w_1,\dots, w_n), (v_1,\dots, v_p)) = r$. Then we have:

$$c \le a + b \text{ because d is a metric in } W$$

$$r \le d + k \text{ because } \mathrm{d_v} \text{ is a metric in a vector space.}$$

Consequently,
$\mathbf{d}((e; w_1,\dots, w_n), (h; v_1,\dots, v_p)) \le \mathbf{d}((e; w_1,\dots, w_n), (l; u_1,\dots, u_m)) + \mathbf{d}((l; u_1,\dots, u_m), (h; v_1,\dots, v_p))$
*i.e.*, the third axiom of metric spaces is true. □

**Corollary 1.** $\mathbf{d}((e; w_1,\dots, w_n),(h; v_1,\dots, v_n)) \ge \mathrm{d}(e, h)$ and $\mathbf{d}((e; w_1,\dots, w_n), (h; v_1,\dots, v_n)) \ge \mathrm{d_v}((w_1,\dots, w_n), (v_1,\dots, v_n))$

**Corollary 2.** $\mathbf{d}((e; w_1,\dots, w_n), (h; v_1,\dots, v_n)) < k$, then $\mathrm{d}(e, h) < k$.

There are other ways to define metrics in the spaces $W_{\mathrm{esw}}$ and $W_{\mathrm{sesw}}$ based on metrics in the base and fiber of the corresponding vector bundle. For instance, it is possible to use the following formulas:

$$\mathbf{d}((e; w_1, \cdots, w_\mathrm{n}), (l; u_1, \cdots, u_\mathrm{m})) = \sqrt{\mathrm{d_v}((e; w_1, \cdots, w_\mathrm{n}), (l; u_1, \cdots, u_\mathrm{m}))^2 + \mathrm{d}(e, l)^2} \tag{4}$$

when $n = m$;

$$\mathbf{d}((e; w_1, \cdots, w_\mathrm{n}), (l; u_1, \cdots, u_\mathrm{m})) = \sqrt{\mathrm{d_v}((w_1, \cdots, w_\mathrm{n}), (u_1, \cdots, u_\mathrm{m}, 0, \cdots, 0))^2 + \mathrm{d}(e, l)^2} \tag{5}$$

when $n > m$; and

$$\mathbf{d}((e; w_1, \cdots, w_\mathrm{n}), (l; u_1, \cdots, u_\mathrm{m})) = \sqrt{\mathrm{d_v}((w_1, \cdots, w_\mathrm{n}, 0, \cdots, 0), (u_1, \cdots, u_\mathrm{m}))^2 + \mathrm{d}(e, l)^2} \tag{6}$$

when $n < m$.

Structures in the spaces $W_{\mathrm{es}}$, $W_{\mathrm{ses}}$, $W_{\mathrm{esw}}$ and $W_{\mathrm{sesw}}$ are inherited by epistemic spaces and their states. In particular, a weighted epistemic space $E$ and each its state is a vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ with the metric **d**.

We remind that a set $X$ in a metric space $E$ with a metric d is called *bounded* if there is a number $k$ such that for any points $a$ and $b$ from $X$, $\mathrm{d}(a, b) < k$.

To study bounded sets in metric spaces that are spaces of vector bundles, we need additional concepts.

**Example 11.** Osgood, Suci and Tannenbaum [30] define *distance* in semantic spaces (cf. Example 8) by the formula from the $m$-dimensional Euclidean spaces:

$$\mathbf{d}(e, l) = \sqrt{\sum_{j=1}^{m} \mathrm{d}_{elj}^2}$$

In this formula, $m$ is the number of factors and $d_{elj}$ is the difference between the coordinates of the elements $e$ and $l$ with respect to the same factor (dimension) $j$. In the most refined models, the number $m$ is equal to 3 [30].

Let us consider a vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ with the fiber $F$.

**Definition 12.** A set $X \subseteq E$ is called *rectangular* in $E$ if $X = \{(b, u) \mid b \in X_e, u \in F$ and for any $a \in X_e$ and $v \in F$ $((a, v) \in X \Rightarrow (b, v) \in X)\}$.

**Example 12.** Let us consider a trivial vector bundle $\boldsymbol{H} = (H = \{a, b, c\} \times \boldsymbol{R}, p_h, H_e = \{a, b, c\})$. Then the set $X = \{(a, 1), (a, 2), (b, 1), (b, 2), (c, 1), (c, 2)\}$ is rectangular in $H$, while the set $Z = \{(a, 1), (a, 3), (b, 1), (b, 5), (c, 1), (c, 3)\}$ is not rectangular in $H$.

**Definition 13.** If $X \subseteq E$, then the minimal rectangular in $E$ set $\mathrm{R}(X)$ that contains $X$ is called the *rectangular closure* of $X$ in $E$.

**Lemma 2.** The rectangular closure of a set in $E$ always exists and is unique.

**Lemma 3.** The operation of taking the rectangular closure of a set in $E$ is a closure operation in the sense of ( ) on sets in metric spaces.

In particular, the operation of taking the rectangular closure of a set in $E$ is idempotent, *i.e.*, $\mathrm{R}(\mathrm{R}(X)) = \mathrm{R}(X)$ for any $X \subseteq E$.

**Lemma 4.** A set $X$ in $E$ is rectangular if and only if $X = \mathrm{R}(X)$.

**Definition 13.** If $X \subseteq E$, then the *fiber projection* $\sigma(X)$ of $X$ is defined as follows

$$\sigma(X) = \{ u \mid \exists b \in X_e ((b, u) \in X \}$$

For instance, taking sets $X$ and $Z$ from Example 8, we see that:

$$\sigma(X) = \{1, 2\}$$

and

$$\sigma(Z) = \{1, 3, 5\}$$

Definitions imply the following result.

**Lemma 5.** A set $X$ in $E$ is rectangular if and only if $X = X_e \times \sigma(X)$.

Some properties of sets and their rectangular closures are the same.

**Proposition 2.** A subset $X$ of the space $E$ of the vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ is bounded if and only if its rectangular closure is bounded.

**Proof.** *Sufficiency*. By definition, any subset of a bounded set is bounded.

*Necessity*. Let us assume that $X$ is bounded. It means that there is a positive number $k$ such that $\mathbf{d}(x, z) < k$ for any two points $x = (a, u)$ and $z = (b, v)$ from $X$ where $a, b \in X_e$.

Let us take two points $p$ and $q$ from R($X$). Then $p = (c, w)$ and $q = (d, y)$ with $c, d \in X_e$. By the definition of the rectangular closure R($X$), there are points $x$ and $z$ from $X$ such $x = (a, w)$ and $z = (b, y)$ with $a, b \in X_e$. By the properties of metric,

$$\mathbf{d}(p, q) \leq \mathbf{d}(p, x) + \mathbf{d}(x, z) + \mathbf{d}(z, q)$$

By initial conditions, $\mathbf{d}(x, z) < k$. At the same time, by the definition of the metric **d** and Corollary 2, we have:

$$\mathbf{d}(p, x) = \mathbf{d}((c, w), (a, w)) = \mathbf{d}(c, a) < k$$

and

$$\mathbf{d}(z, q) = \mathbf{d}((b, y), (d, y)) = \mathbf{d}(b, d) < k$$

Consequently,

$$\mathbf{d}(p, q) < 3k$$

Proposition is proved because $p$ and $q$ are arbitrary points from R($X$). □

Reducing the problem of boundedness to rectangular sets, now we find conditions of boundedness for rectangular sets.

**Proposition 3.** A rectangular subset $X$ of the space $E$ of the vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ is bounded if and only if the projection $X_e = p_E(X)$ of $X$ and the fiber projection $\sigma(X)$ of $X$ are uniformly bounded.

**Proof.** *Necessity*. Let us assume that the projection $X_e = p_E(X)$ of $X$ is unbounded. It means that for any positive number $k$, there are two points $a$ and $b$ in $X_e$ such that $\mathbf{d}(a, b) > k$. As $X_e$ is the projection of $X$, there are two points $x$ and $z$ in $X$ such that $a = p_E(x)$ and $b = p_E(z)$. By the definition of the metric in the space $E$, $\mathbf{d}(x, z) \geq \mathbf{d}(a, b) > k$. Consequently, $X$ is also unbounded.

Now, let us suppose that the fiber projection $\sigma(X)$ of $X$ is not uniformly bounded. It means that for any positive number $k$, there are two points $u$ and $v$ in $\sigma(X)$ such that $d_v(u, v) > k$. As $\sigma(X)$ is a projection of $X$, there are points $x = (a, u)$ and $z = (b, v)$ from the space $X$. By Corollary 2, $\mathbf{d}(x, z) \geq k$ as by choice of the points $u$ and $v$, $d_v(u, v) > k$. Thus, the space $X$ is not bounded.

Then by the Law of Contraposition, if the space $X$ is bounded, then the projection $X_e = p_E(X)$ of $X$ and the fiber projection $\sigma(X)$ of $X$ are bounded.

*Sufficiency*. Let us suppose that the projection $X_e = p_E(X)$ of $X$ and the fiber projection $\sigma(X)$ of $X$ are bounded. It means that there is a positive number $k$, such that for any two points $a$ and $b$ in $X_e$, we have $d(a, b) < k$ and there is a positive number $h$, such that for any two points $u$ and $v$ from $\sigma(X)$, we have $\mathbf{d}(x, z) < h$

Let us take two points $x$ and $z$ from $X$. Then $x = (a, u)$ and $z = (b, v)$ where $a, b \in X_e$, while $u$ and $v$ belong to the fiber $F$ of the bundle **E**. As $X$ is a rectangular set, points $(a, u)$ and $(b, v)$ belong to $X$ and by definition,

$$\mathbf{d}(x, y) = \mathbf{d}(a, b) + \mathbf{d}(u, v) < k + h$$

Consequently, the set $X$ is bounded. □

Propositions 2 and 3 imply the following result.

**Corollary 3.** A subset $X$ of the space $E$ of the vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ is bounded if and only if the projection $X_e = p_E(X)$ of $X$ and the fiber projection $\sigma(X)$ of $X$ are uniformly bounded.

Note that here we study weighted epistemic structures with real number weights and weighted epistemic space in which weights form real vector spaces. However, using the same technique, it is possible to obtain similar results for weighted epistemic structures with complex number weights or vector weights and for weighted epistemic space in which weights form complex vector spaces.

## 5. Information Operators and Dynamics in Epistemic Spaces

Epistemic information operators act in (weighted) epistemic spaces, transforming these spaces and describing dynamics of infological systems and epistemic spaces, which model infological systems.

Let us consider two (weighted) epistemic spaces $E$ and $H$.

**Definition 14.** (a) A (partial) mapping $A$: $E \to H$ is called an *epistemic information operator*.
(b) If both epistemic spaces $E$ and $H$ have the same structure and an information operator $A$: $E \to H$ preserves this structure, then $A$ is called a *structured information operator* or an *information homomorphism*.
(c) When $E = H$, the operator $A$ is called an *inner epistemic information operator*.

An inner epistemic information operator changes elements or states or elements of (weighted) epistemic spaces and multispaces.

There are three basic types of inner epistemic information operators: content, bond and weight operators.

A content epistemic information operator acts on symbolic epistemic items.

For instance, all information operators studied in [1,3,4] are content epistemic information operators.

There are three key content inner epistemic information operators:

An *addition*/*deletion content operator* AD (DL) adds a symbolic epistemic item (knowledge item) to the state of the (weighted) epistemic space.

A *transformation*/*substitution content operator* TR (ST) transforms or substitutes a symbolic epistemic item in the state of the (weighted) epistemic space.

A *substantiation content operator* switches on or off an existing symbolic epistemic item in the state of the (weighted) epistemic space.

**Example 13.** When an intelligent agent learns, it usually adds knowledge items to its knowledge base changing in such a way the state of this base. The knowledge base is naturally represented by an appropriate epistemic space. Thus, growth of knowledge in the process of learning is modeled by application of adding content information operators.

**Definition 15.** (a) A *replica* of an epistemic (knowledge) item is another knowledge item equivalent to the initial one.
(b) A *replication epistemic information operator REPL* makes a replica of an epistemic (knowledge) item and adds it to the current epistemic (knowledge) state.

Note that copies of an epistemic (knowledge) item always are its replicas but a replica of an epistemic (knowledge) item is not always its copy.

**Example 14.** Let us consider logical knowledge representation in which knowledge items are propositions. Then according to laws of logic there are equivalent propositions. For instance, taking the proposition (1) "*B* implies *A*", we have equivalent propositions (2) "*A* follows from *B*", (3) "If *B*, then *A*", and (4) "*A* is a consequence of *B*". All of them are replicas of one another although they are not copies.

Addition of symbolic epistemic items can be performed by five operations:

- by generation of a new item inside the current state of the (weighted) epistemic space;
- by generation of a new item outside (the current state of) the epistemic space and its transition into the current state of the (weighted) epistemic space;
- by transition of an existing item from the epistemic space into the current state of the (weighted) epistemic space;
- by replication of an item from the current state of the (weighted) epistemic space;
- by replication of an item outside (the current state of) the epistemic space and transition of this replica into the current state of the (weighted) epistemic space.

Consequently, substitution of symbolic epistemic items can be performed by five operations because substitution is the sequential composition of elimination and addition.

In the case of stratified epistemic spaces, there is one more type of key content epistemic operators, namely, a moving operators MV, which moves epistemic items from one strata to another [3].

A *transformation epistemic information operator TR* takes a group of epistemic (knowledge) items (may be, one item) from the current epistemic (knowledge) state and transforms it into another group of epistemic (knowledge) items (may be, into one item).

A *generation epistemic information operator TR* takes a group of epistemic (knowledge) items (may be, one item) from the current epistemic (knowledge) state and generates another group of epistemic (knowledge) items (may be, one item).

The difference between transformation and generation is that in generation, the initial group of epistemic items is preserved, while in transformation, it is not preserved.

A *bond epistemic information operator* acts on connections (bonds or relations) between symbolic epistemic items.

Such operators as interpretation and reinterpretation of information/knowledge items [2] are bond epistemic information operators.

There are three key bond epistemic information operators:

(1) An *addition/deletion bond operator* adds or deletes a connection (bond or relation) between symbolic epistemic items in the current epistemic (knowledge) state of the (weighted) epistemic space.
(2) A *substitution bond operator* changes a connection (bond or relation) between symbolic epistemic items in the current epistemic (knowledge) state of the (weighted) epistemic space to another connection (bond or relation).
(3) A *substantiation bond operator* switches on or off an existing connection (bond or relation) between symbolic epistemic items in the current epistemic (knowledge) state of the (weighted)

epistemic space.

A *weight epistemic information operator* acts on weights of symbolic epistemic items.

There are three basic weight epistemic information operators:

(1) An *addition/deletion weight operator* adds or deletes a weight to symbolic epistemic items in the weighted epistemic space.
For instance, epistemic items had one weight *constructible*, which indicates whether the structure is constructible or not. Then the weight *complexity* was added by the adding operator. The new weight reflects complexity of the structure construction.

(2) A *transformation/substitution weight operator* substitutes one weight of symbolic epistemic items in the weighted epistemic space by another weight.
For instance, epistemic items had the weight *justification*, which is then substituted by the weight *provability*. In a different situation, a transformation operator changes the weight *complexity* for the weight *hardship*. Such situation happens in software engineering [35].

(3) A *value changing weight operator* changes weights of symbolic epistemic items in the weighted epistemic space.

For instance, let us assume the value of the weight *complexity* for texts was estimated based on recursive algorithms, such as Turing machines. Later the complexity estimate was obtained by means of super- recursive algorithms, such as inductive Turing machines. As it is proved that super-recursive algorithms decrease complexity [27], the value-changing operator has to be applied to give correct complexity of the texts.

There also mixed epistemic information operators. A *mixed epistemic information operator* acts on symbolic epistemic items in an epistemic state, their weights and their connections (bonds or relations).

For instance, a mixed epistemic information operator can act on knowledge items in a knowledge state, their weights and their connections (bonds or relations).

Operators of logical inference, such as rules of deduction, are mixed epistemic information operators act because they add new knowledge items in the form of propositions or/and predicates and establish relations of provability/deducibility between propositions or/and predicates.

Subspaces of knowledge spaces represent subsystems of knowledge systems. For instance, in large knowledge systems, such as a scientific theory, it is possible to separate the subsystem of denotational knowledge and the subsystem of operational knowledge.

It looks like it might be sufficient to consider only finite or at least, locally finite agents. However, if knowledge is represented by logical statements and it is assumed (as it is done, for example, in the theory of semantic information developed by Bar-Hillel and Carnap [12]) that any knowledge system contains all logical consequences of all its elements, then an agent with such knowledge system is infinite. In information algebras, portions of information are represented by close subsets of sentences from a logical language $L$ [36].

However, in conventional logics closed with respect to such information operators as deduction, sets are infinite because any sentence $p$ implies $p \vee q$ for any sentence $q$ from $L$, which is, as a rule, infinite (cf., for example, [37]). Thus, in the context of classical logic and information algebras any portion of information has infinitely many representations. Consequently, such a portion generates a system with the infinite number of knowledge items.

Let us consider epistemic information operators from a weighted epistemic space $E$ into a weighted epistemic space $H$.

**Definition 16.** An epistemic information operator $A$: $E \to H$ is called:
(a) *stationary* if for any epistemic structures $e$ and $l$, the equality $A(e; w_1, \ldots, w_k) = (l; v_1, \ldots, v_h)$ implies the equality $A(e; u_1, \ldots, u_k) = (l; q_1, \ldots, q_h)$ for any $(e; u_1, \ldots, u_k)$.
(b) *permanent* if for any weighted epistemic structure $(e; w_1,\ldots, w_k)$, we have $A(e; w_1,\ldots, w_k) = (e; v_1,\ldots, v_k)$.
(c) *semipermanent* if for any epistemic structure $e$ and any number $k$, there is a number $h$ such that for any system of weights $(w_1, \ldots, w_k)$ of $e$, we have $A(e; w_1,\ldots, w_k) = (e; v_1,\ldots, v_h)$.

Definitions imply the following result.

**Lemma 6.** Any permanent epistemic information operator $A$ is semipermanent, while any semipermanent epistemic information operator $B$ is stationary.

**Lemma 7.** (a) Any weight epistemic information operator is semipermanent.
(b) Operators of adding weights and of deleting weights are semipermanent but not permanent.
(c) Operators of substituting weights and of changing values of weights are permanent.

Stationary epistemic information operator are related to morphisms of epistemic vector bundles.

We remind [32] that a morphism of a vector bundle $\mathbf{E} = (E, p, B)$ into a vector bundle $\mathbf{H} = (H, r, D)$ is a pair of continuous mappings $f$: $E \to H$ and $g$: $B \to D$ such that the following Diagram (3) is commutative.

$$\begin{tikzcd} E \arrow[r, "f"] \arrow[d, "p"'] & H \arrow[d, "r"] \\ B \arrow[r, "g"'] & D \end{tikzcd}$$

(3)

Note that morphisms of a vector bundles are mappings (morphisms) of named sets, which are studied in the theory of named sets [2]. Thus, weighted epistemic spaces form a category with vector bundle morphisms as its morphisms. It makes possible application of results for categorical information modeling [38,39] to epistemic information operators in weighted epistemic spaces.

Let us consider two weighted epistemic spaces $E$ and $H$.

**Proposition 4.** An epistemic information operator $A$: $E \to H$ is stationary if and only if it induces a morphism of the vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ into the vector bundle $\boldsymbol{H} = (H, p_H, H_e)$.

**Proof.** *Necessity*. Let us consider a stationary epistemic information operator $A$: $E \to H$. Then by Definition 16, each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p_E, E_e)$ is mapped by $A$ into a single fiber $G_b$ of the vector bundle $\boldsymbol{H} = (H, p_H, H_e)$. That is why, we can build the mapping $C$: $E_e \to H_e$ defining $C(a) = b$. By construction, we have $C(p_E(x)) = p_H(A(x))$ for all elements $x$ from $E$. This gives us the commutative Diagram (4).

$$\begin{tikzcd} E \arrow[r, "A"] \arrow[d, "p_E"] & H \arrow[d, "p_H"'] \\ E_E \arrow[r, "C"'] & H_E \end{tikzcd}$$

(4)

It means that the pair $(A, C)$ is a morphism of the vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ into the vector bundle $\boldsymbol{H} = (H, p_H, H_e)$.

*Sufficiency*. If the pair $(A, C)$ is a morphism of the vector bundle $\boldsymbol{E} = (E, p_E, E_e)$ into the vector bundle $\boldsymbol{H} = (H, p_H, H_e)$, *i.e.*, the Diagram (4) is commutative. Consequently, each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p_E, E_e)$ is mapped by $A$ into a single fiber $G_b$ of the vector bundle $\boldsymbol{H} = (H, p_H, H_e)$, *i.e.*, $A$ is a stationary epistemic information operator. □

**Definition 17.** A stationary epistemic information operator $A: E \to H$ is called:
(a) *uniform* if for any real number $a$ and any weighted epistemic structures $(e; w_1, \ldots, w_k)$ and $(l; v_1, \ldots, v_h)$, the equality $A(e; w_1, \ldots, w_k) = (l; v_1, \ldots, v_h)$ implies the equality $A(e; au_1, \ldots, au_k) = (l; av_1, \ldots, av_h)$.
(b) *additive* if $A(e; w_1 + u_1, \ldots, w_k + u_k) = A(e; w_1, \ldots, w_k) + A(e; u_1, \ldots, u_k)$.
(c) *linear* if it is uniform and additive.

**Example 15.** Let us consider a weighted epistemic space $E$, in which there are $n$ epistemic structures $e_1, e_2, e_3, \ldots, e_n$, the distance between any two of them is 1 and each of them has one weight $w$ the range of which is the real line $\boldsymbol{R}$. Taking a real number $t$, we define the epistemic information operator $A$ by the following rule:

$$A(e_k, w) = (e_k, tw)$$

By definition, this operator is linear and thus, uniform and additive.

At the same time, the epistemic information operator $B$ with $B(e_k, w) = (e_k, k)$ is neither uniform nor additive nor linear.

To study linear epistemic information operators, we need some topological constructions.

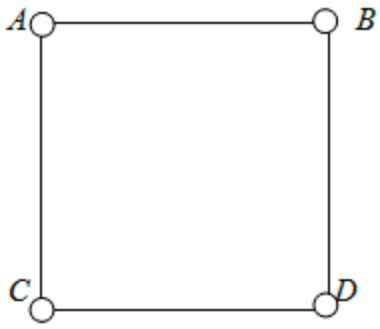
In [40], the concept of path Q-connectedness is introduced and studied. To find relations between linearity and boundedness of information operators, we need to further develop this concept for metric spaces. Path Q-connectedness is important for epistemic spaces because in many cases, epistemic spaces have the structure of a graph or network of epistemic items connected by various relations. For instance, Motter, *et al*. [25] build in such a way the conceptual network of a language.

Let $C$ be a subspace of a metric space $U$ with the distance function $\mathbf{d}$.

**Definition 18.** The space $C$ is called *path* $(q, r)$*-connected* in $U$ if for any two points $a$ and $b$ in $C$, there exists a sequence $a_1, a_2, a_3, \ldots, a_n$ of points in $C$ such that $\mathbf{d}(a, a_1) \leq r$, $\mathbf{d}(a_n, b) \leq r$, $\mathbf{d}(a_i, a_{i+1}) \leq r$ for all $i = 1, 2, 3, \ldots, n - 1$ and $[\mathbf{d}(a, a_1) + \mathbf{d}(a_1, a_2) + \mathbf{d}(a_2, a_3) + \ldots + \mathbf{d}(a_{n-1}, a_n) + \mathbf{d}(a_n, b)] < q \cdot \mathbf{d}(a, b)$.

**Example 16.** Taking a square in which the length of the side is equal and which is situated in an Euclidean plane (cf. Figure 3) and the space of its vertices $C = \{A, B, C, D\}$, we see that it is path (2, 1)-connected but it is not path (1, 1)-connected and not path (2, ½)-connected. Indeed, $\mathbf{d}(A, C) = \sqrt{2}$, the lengths of both paths $(A, B, C)$ and $(A, D, C)$ is equal to 2 with $\mathbf{d}(A, B) = \mathbf{d}(B, C) = \mathbf{d}(A, D) = \mathbf{d}(D, C) = 1$. Consequently, $\mathbf{d}(A, B) + \mathbf{d}(B, C) < 2\mathbf{d}(A, C)$ but $\mathbf{d}(A, B) + \mathbf{d}(B, C) > 1 \cdot \mathbf{d}(A, C)$. Thus, the space $C$ is path (2, 1)-connected but it is not path (1, 1)-connected. In addition, it is not path $(q, p)$-connected when $p < 1$ because there no paths between $A$ and $C$ such that the distance between two consecutive points is less than 1.

**Figure 5.** A topological space that is path (2, ½)-connected but it is not path (1, 1)-connected.




However, not all sets in metric spaces are path $(q, r)$-connected.

**Example 17.** The parabola $y = x^2$ as the space $C$, we see that it is not path $(q, r)$-connected for any numbers $q$ and $r$. Indeed, taking points $u = (x, x^2)$ and $w = (-x, x^2)$, we see that the distance between these points $\mathrm{d}(u, w) = 2x$. Now let us suppose that this parabola $C$ is path $(q, r)$-connected for some numbers $q$ and $r$. It means that there is a sequence $a_1, a_2, a_3, \ldots, a_n$ of points in $C$ such that $\mathrm{d}(u, a_1) \leq r$, $\mathbf{d}(a_n, w) \leq r$, $\mathbf{d}(a_i, a_{i+1}) \leq r$ for all $i = 1, 2, 3, \ldots, n - 1$. As all points $a_1, a_2, a_3, \ldots, a_n$ belong to $C$, the sum $[\mathbf{d}(u, a_1) + \mathbf{d}(a_1, a_2) + \mathbf{d}(a_2, a_3) + \ldots + \mathbf{d}(a_{n-1}, a_n) + \mathbf{d}(a_n, w)]$ is larger than $x^2 - (½\, r)^2$. At the same time, we have $[\mathbf{d}(u, a_1) + \mathbf{d}(a_1, a_2) + \mathbf{d}(a_2, a_3) + \ldots + \mathbf{d}(a_{n-1}, a_n) + \mathbf{d}(a_n, w)] < q \cdot \mathbf{d}(u, w)$. Thus, $x^2 - (½\, r)^2 < 2qx$ because $\mathrm{d}(u, w) = 2x$. Transforming this inequality, we obtain

$$x^2 - 2qx < (½\, r)^2$$

$$x(x^2 - 2q) < (½\, r)^2$$

For sufficiently big $x$, this inequality cannot be valid because $r$ is a fixed number. Thus, our assumption is not true and the parabola $y = x^2$ is not path $(q, r)$-connected for any numbers $q$ and $r$.

Here we are mostly interested in content epistemic information operators, which we simply call epistemic information operators. Using classical concepts of continuity and boundedness [26,41], as well as the concept of $(p, q)$-continuity from neoclassical analysis [42], we explicate important classes of epistemic information operators.

Let us consider epistemic information operators from a weighted epistemic space $E$ with a metric $\mathbf{d}$ into a weighted epistemic space $H$ with a metric $\mathbf{d}$. In this case, it is possible to define the diameter $\mathbf{d}$ of sets in $E$ and in $H$. Namely, if $X \subseteq E$, then $\mathbf{d}(X) = \sup \{\mathbf{d}(x, z); x, z \in X\}$ when this supremum exists and undefined otherwise.

**Definition 19.** An epistemic information operator $A$: $E \to H$ is called:
(a) *bounded* if given a state $X$ of $E$, for any number $r$, there is a number $t$ such that the condition $\mathbf{d}(X) \leq r$, implies the condition $\mathbf{d}(A(X)) \leq t$.
(b) *uniformly bounded* if for any number $r$, there is a number $t$ such that for any state $X$ of $E$, the condition $\mathrm{d}(X) \leq r$, implies the condition $\mathbf{d}(A(X)) \leq t$.
(c) *continuous* if $A$ is a continuous mapping.
(d) $(p, q)$-*continuous* if $A$ is a $(p, q)$-continuous mapping.

Boundedness of an epistemic information operator $A$ means that when the distances between epistemic staructures in a set $X$ are bounded, then the distances between epistemic staructures in the image $A(X)$ of the set $X$ are bounded.

$(p, q)$-continuity of an epistemic information operator $A$ informally means that when the distances between epistemic staructures in a set $X$ are not larger than $p$, then the distances between epistemic staructures in the image $A(X)$ of the set $X$ are not larger than $q$.

**Lemma 8.** Any uniformly bounded epistemic information operator $A$: $E \to H$ is bounded.

In a discrete metric space, any point is an open and a closed set [26]. Thus, any epistemic information operator in an epistemic space $E$ is continuous, open and closed.

Results from [42] give us the following property of epistemic information operators.

**Lemma 9.** An epistemic information operator $A$: $E \to H$ continuous if and only if it is (0, 0)-continuous.

In addition, we need some constructions from neoclassical analysis, such as fuzzy limits and fuzzy continuity [42].

Let $r \in \boldsymbol{R}^{+}$.

**Definition 20.** (a) An element $a$ from $E$ is called an *r-limit of a sequence l* (it is denoted by $a = r\text{-}\lim_{i\to\infty} a_i$ or $a = r\text{-}\lim l$ ) if for any $\varepsilon \in \boldsymbol{R}^{++}$ the inequality $\mathbf{d}(a, a_i) < r + \varepsilon$ is valid for almost all $a_i$, *i.e.*, there is such $n$ that for any $i > n$, we have $\mathbf{d}(a, a_i) < r + \varepsilon$.
(b) A sequence $l$ that has an $r$-limit is called *r-convergent* and it is said that $l$ *r-converges* to its $r$-limit $a$.

Informally, $a$ is an $r$-limit of a sequence $l$ if for an arbitrarily small $\varepsilon$, the distance between $a$ and all but a finite number of elements from $l$ is smaller than $r + \varepsilon$. In other words, an element $a$ is an $r$-limit of a sequence $l$ if for any $\varepsilon \in \boldsymbol{R}^{++}$ almost all $a_i$ belong to the interval $(a - r - \varepsilon, a + r + \varepsilon)$.

It is a natural generalization of the classical concept of a limit as the following result demonstrates.

**Lemma 10** [42]. A point $a$ a limit of a sequence $l$ if and only if it is a 0-limit of the sequence $l$.

We also need fuzzy continuity.
**Definition 22.** (a) A partial function $f$: $\boldsymbol{R} \to \boldsymbol{R}$ is called $(q, r)$-*continuous at a point* $a \in \boldsymbol{R}$ if for any sequence $l = \{ a_i \in \boldsymbol{R};\ i = 1, 2, 3, \ldots\}$, for which $a$ is an $q$-limit, the point $f(a)$ is an $r$-limit of the sequence $\{ f(a_i) \in \boldsymbol{R};\ i = 1, 2, 3, \ldots\}$.
(b) A function $f$: $\boldsymbol{R} \to \boldsymbol{R}$ is called $(q, r)$-*continuous in* (*inside*) set $X \subseteq \boldsymbol{R}$ if $f(x)$ (the restriction of $f(x)$ on $X$) is $(q, r)$-continuous at each point $a$ from $X \cap \mathrm{Dom}\, f$.

Fuzzy continuity is a natural generalization of the classical concept of continuity as the following result demonstrates.

**Lemma 11** [42]**.** A function $f(x)$ is continuous at a point $a \in \boldsymbol{R}$ if and only if it is (0,0)-continuous at the point $a$.

These results show that the concept of $(q, r)$-continuity is a natural extension of the concept of conventional continuity.

**Lemma 12** [42]**.** If $t > r$, and $p < q$, then any $(q, r)$-continuous at $a$ function $f(x)$ is also $(p, t)$-continuous at $a$.

Note that if $q < p$, then it is possible that a $(q, r)$-continuous at $a$ function is not $(p, r)$-continuous at $a$. For instance, the function $f(x) = x$ is (0, 0)-continuous at the point 0, but for any $p > 0$, it is not $(p, 0)$-continuous at 0.

Let us consider two (weighted) epistemic spaces $E$ and $H$, assuming that the space $E$ is a path $(q, 1)$-connected metric space with the metric $\mathbf{d}$ and the space $H$ is a metric space with the metric $\mathbf{d}$. Denoting metrics in different spaces by the same letter $\mathbf{d}$ follows the mathematical tradition and does not cause confusion.

**Theorem 1.** A epistemic information operator $A$: $E \to H$ is $(1, k)$-continuous for some positive number $k$ if and only if it is uniformly bounded.

**Proof.** *Necessity*. Let us consider a $(1, k)$-continuous epistemic information operator $A$: $E \to H$ and two points $a$ and $b$ from a set $X$ in $E$. By Definition 17, there exists a path (sequence of points) $l = \{a_1, a_2, a_3, \ldots, a_n\}$ in $E$ such that $\mathbf{d}(a, a_1) \leq 1$, $\mathbf{d}(a_n, b) \leq 1$, $\mathrm{d}(a_i, a_{i+1}) \leq 1$ for all $i = 1, 2, 3, \ldots, n-1$ and $[\mathbf{d}(a, a_1) + \mathbf{d}(a_1, a_2) + \mathbf{d}(a_2, a_3) + \ldots + \mathbf{d}(a_{n-1}, a_n) + \mathrm{d}(a_n, b)] < q \cdot \mathbf{d}(a, b)$. If for some $i$, $\mathbf{d}(a_i, a_{i+1}) < ½$ and $\mathbf{d}(a_{i+1}, a_{i+2}) < ½$, then it is possible to eliminate the point $a_{i+1}$ from the path because by properties of metric, $\mathbf{d}(a_i, a_{i+2}) \leq \mathbf{d}(a_{i+1}, a_{i+2}) + \mathbf{d}(a_i, a_{i+1}) < ½ + ½ = 1$ and the new path is not longer than the previous one. We can reduce the initial path in such a way and assume that we have an irreducible path $l = \{a_1, a_2, a_3, \ldots, a_n\}$ between $a$ and $b$.

As $A$ is a $(1, k)$-continuous epistemic information operator, $\mathbf{d}(A(a), A(a_1)) < k$, $\mathbf{d}(A(a_n), A(b)) < k$, $\mathbf{d}(A(a_i), A(a_{i+1})) < k$ for all $i = 1, 2, 3, \ldots, n - 1$. Thus,

$$\mathbf{d}(A(a), A(b)) \leq \mathbf{d}(A(a), A(a_1)) + \mathbf{d}(A(a_1), A(a_2)) + \ldots + \mathbf{d}(A(a_{i-1}), A(a_n)) + \mathbf{d}(A(a_n), A(b)) < k(n + 1)$$

Now let us estimate the number $n$. Taking an irreducible path $l = \{a_1, a_2, a_3, \ldots, a_n\}$, we know that between any two pairs $(a_i, a_{i+1})$ and $(a_{i+2}, a_{i+3})$ with the distance less than ½, there is at least, one pair $(a_{i+1}, a_{i+2})$ with the distance larger than ½. Thus, the number of pairs $(a_i, a_{i+1})$ the distance between which larger than ½ is more than $(¼)n$. Consequently, the length of the path $l = \{a_1, a_2, a_3, \ldots, a_n\}$ is larger than $½ \cdot (¼)n = (⅛)n$, *i.e.*, $(⅛)n < [\mathbf{d}(a, a_1) + \mathbf{d}(a_1, a_2) + \mathbf{d}(a_2, a_3) + \ldots + \mathbf{d}(a_{n-1}, a_n) + \mathbf{d}(a_n, b)] < q \cdot \mathbf{d}(a, b) = q \cdot d$ where the distance $\mathbf{d}(a, b)$ is equal to $d$. Thus, $n < 8q \cdot d$.

Let us assume that $X$ is a bounded set. It means that there is a positive number $h$ such that for two points $a$ and $b$ from $X$, the distance $\mathbf{d}(a, b)$ is less than $h$. It is possible to assume that $h > 1$. Then $n < 8q \cdot h$ and $\mathbf{d}(A(a), A(b)) < k(n + 1) < k(8q \cdot h + 1) = t < (8qk + k) \cdot h$. It means that the operator $A$ is uniformly bounded because numbers $k$ and $q$ are constants and $t$ depends only on one variable $h$.

Necessity is proved.

*Sufficiency*. Let us consider a uniformly bounded epistemic information operator $A$: $E \to H$. Then (cf. Definition 18) for any number $r$, there is a number $t$ such that for any state $X$ of $E$, the condition $\mathbf{d}(X) \leq r$, implies the condition $\mathbf{d}(A(X)) \leq t$. In particular, for the number 1, there is a number $k$ such that for any points $a$ and $b$ from $E$, the condition $\mathbf{d}(a, b) \leq 1$, implies the condition $\mathbf{d}(A(a), A(b)) \leq k$. It means that the operator $A$ is (1, $k$)-continuous. □

**Remark 1.** The proof of Theorem 1 is sufficiently general. So, this result remains true for general metric spaces that satisfy the necessary conditions.

One of the basic results of functional analysis is the theorem stating that a linear operator in Banach space is continuous if and only if it is uniformly bounded [34,43]. It is demonstrated that (1, 0)-continuity is stronger than continuity in metric spaces [42]. Thus, it is possible to ask a question whether it would be possible to change the condition of (1, $k$)-continuity to the condition of continuity in Theorem 1. The following example shows that it is impossible as there are linear epistemic information operators that are continuous but not uniformly bounded.

**Example 18.** Let us consider a weighted epistemic space $E$, in which there is a countable number of epistemic structures $e_1, e_2, e_3, \ldots, e_n, \ldots$, the distance between any two of them is 1 and each of them has one weight $w$ the range of which is the real line $\boldsymbol{R}$. We define the epistemic information operator $A$ by the following rule:

$$A(e_n, w) = (e_n, nw)$$

By definitions, this operator is linear and continuous but it is not uniformly bounded. Indeed, $\mathbf{d}((e_1, 1), (e_n, 1)) \leq 2$, while $\mathbf{d}(A(e_1, 1)\, A(e_n, 1)) > n$ for any $n$.

Even more, there are linear epistemic information operators that are continuous but not bounded.

**Example 19.** Let us consider a weighted epistemic space $E$, in which there is a countable number of epistemic items $e_1, e_2, e_3, \ldots, e_n, \ldots$ such that $\mathbf{d}(e_n, e_{n+1}) = (1/2^n)$ and $\mathbf{d}(e_n, e_{n+k}) = \sum_{i=n}^{n+k-1} (1/2^i)$. Each of these epistemic items has one weight $w$ the range of which is the real line $\boldsymbol{R}$. We define the epistemic information operator $A$ by the following rule

$$A(e_n, w) = (e_n, nw)$$

By definitions, the set $U = \{(e_1, 1), (e_2, 1), (e_3, 1), \ldots, (e_n, 1), \ldots\}$ is bounded as $\mathbf{d}(U) = \sup \mathbf{d}(e_n, e_{n+k}) < 3$. The operator $A$ is linear and continuous. However, it is not bounded because in the set $A(U)$, there are pairs of points with the arbitrary big distance between them, e.g., $\mathbf{d}(A(e_1, 1), A(e_n, 1)) > n$.

Although continuity is insufficient for boundedness of a linear epistemic information operator in a general case, there are situations when boundedness is still equivalent to continuity.

Let us consider two weighted epistemic spaces $E$ and $H$ such that both spaces $E_e$ and $H_e$ are metric spaces with the metric $\mathbf{d}$, the base epistemic space $E_e$ is finite and all fibers $F_a$ of the vector bundle $\mathbf{E} = (E, p, E_e)$ and fibers $G_a$ of the vector bundle $\mathbf{H} = (H, p, H_e)$ are hyperseminormed vector spaces.

**Theorem 2.** A linear epistemic information operator $A$: $E \to H$ is continuous in each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p, E_e)$ if and only if $A$ is bounded.

**Proof.** *Necessity*. Let us consider an epistemic information operator $A$: $E \to H$ continuous in each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p, E_e)$ and a bounded set $X$ in $E$. Then each intersection $X_a = X \cap F_a$ is a bounded set because a subset of a bounded set is also bounded. As the operator $A$ is continuous in the fiber $F_a$, the image $A(X_a)$ of $X_a$ is bounded [33]. As the union of a finite number of bounded sets is bounded, $X = \bigcup_{a \in E_e} X_a$ is a bounded set.

*Sufficiency*. Let us consider a bounded epistemic information operator $A$: $E \to H$. Then it is bounded on any subset of $E$, in particular, on each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p, E_e)$. As each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p, E_e)$ is a hyperseminormed vector space, the results from [33] show that the epistemic information operator $A$ is continuous in each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p, E_e)$. □

In many applications, the base epistemic space $E_e$ is finite. For instance, a popular type of the base epistemic space $E_e$ is a semantic network (cf. Example 2) and all known semantic networks are finite. Thus, the conditions from Theorem 2 are almost always satisfied.

As normed vector spaces are an important special case of hyperseminormed vector spaces [33], the following result is implied by Theorem 2.

**Corollary 4.** If all fibers $F_a$ and $G_a$ are normed vector spaces, then a linear epistemic information operator $A$: $E \to H$ is continuous in each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p, E_e)$ if and only if $A$ is bounded.

For linear operators in vector spaces uniform boundedness coincides with boundedness [43,34]. This gives us the following result.

**Corollary 5.** If all fibers $F_a$ and $G_a$ are seminormed, e.g., normed, vector spaces, then a linear epistemic information operator $A$: $E \to H$ is continuous in each fiber $F_a$ of the vector bundle $\mathbf{E} = (E, p, E_e)$ if and only if $A$ is uniformly bounded.

When $E_e$ consists of single element, Theorem 2 gives us the classical result of functional analysis.

**Corollary 6** [34,43]. A linear operator in Banach space is continuous if and only if it is uniformly bounded.

## 6. Conclusions

Here, we studied weighted epistemic information operators in weighted epistemic spaces with weights forming real vector spaces. However, using the same technique, it is possible to obtain similar results for epistemic information operators in weighted epistemic space with weights forming complex vector spaces.

An interesting problem is to study properties of the sequential composition of epistemic information operators. Sequential composition allows one to build new operators from the given ones. Besides, the categorical approach to information studies is based on sequential composition [39]. It would be also appealing to introduce and study other operations in the space of epistemic information operators, for example, the ones that mathematically represent operations with information studied in [44].

**Acknowledgment**

The author is grateful to unknown reviewers for their useful comments.

**Conflicts of Interest**

The author declares no conflict of interest.

**References**

1. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: Singapore, Singapore, 2010.
2. Burgin, M. *Theory of Named Sets*; Nova Science Publishers: New York, NY, USA, 2011.
3. Burgin, M. Epistemic information in stratified M-spaces. *Information* **2011**, *2*, 697–726.
4. Mizzaro, S. Towards a Theory of Epistemic Information. In *Information Modelling and Knowledge Bases*; IOS Press: Amsterdam, The Netherlands, 2001; pp. 1–20.
5. Burgin, M. Information and transformation. *Transformation* **1998/1999**, *10*, 48–53. (in Polish)
6. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
7. Frieden, R.B. *Physics from Fisher Information*; Cambridge University Press: Cambridge, UK, 1998.
8. Von Weizsäcker, C.F. Die Quantentheorie der einfachen Alternative (Komplementarität und Logik, II). *Z. Naturforsch.* **1958**, *13*, 245–253. (in German)
9. Von Weizsäcker, C.F.; Scheibe, E.; Süssmann, G. Komplementarität und Logik, III (Mehrfache Quantelung). *Z. Naturforsch.* **1958**, *13*, 705–721. (in German)
10. Wheeler, J.A. Information, Physics, Quantum: The Search for Links. In *Complexity*, *Entropy*, *and the Physics of Information*; Zurek, W., Ed.; Addison-Wesley: Redwood City, CA, USA, 1990; pp. 3–28.
11. Burgin, M. Information in the context of education. *J. Interdiscip. Stud.* **2001**, *14*, 155–166.
12. Bar-Hillel, Y.; Carnap, R. Semantic information. *Br. J. Philos. Sci.* **1958**, *4*, 147–157.
13. Hintikka, J. The Varieties of Information and Scientific Explanation. In *Inquiry as Inquiry: A Logic of Scientific Discovery*; Jaakko Hintikka Selected Papers; Springer: New York, NY, USA, 1999; Volume 5, pp 205–225.
14. Hintikka, J. Surface Information and Depth Information. In *Information and Inference*; Synthese Library, Volume 28; Springer: Dordrecht, The Netherlands, 1970; pp. 263–297.
15. Shreider, Y.A. On the semantic characteristics of information. *Inf. Storage Retr.* **1965**, *2*, 221–233.
16. Anderson, R.C. The Notion of Schemata and the Educational Enterprise. In *Schooling and the Acquisition of Knowledge*; Anderson, R.C., Spiro, R.J., Montague, W.E., Eds.; Lawrence Erlbaum: Hillsdale, NJ, USA, 1977.
17. Arbib, M. Schema Theory. In *The Encyclopedia of AI*; Wiley-Interscience: New York, NY, USA, 1992; pp. 1427–1443.
18. Armbruster, B. Schema Theory and the design of content-area textbooks. *Educ. Psychol.* **1996**, *21*, 253–276.

19. Barwise, J.; Perry, J. *Situations and Attitudes*; MIT Press: Cambridge, MA, USA, 1983.
20. Burgin, M.; Bratalskii, E.A. The Principle of Asymptotic Uniformity in Complex System Modeling. In *Operation Research and Automated Control Systems*; Naukova Dumka: Kiev, Ukraine, 1986; pp. 115–122. (in Russian)
21. Burgin, M.S.; Bratalskii, E.A.; Belkov, M.S. The language PDL for the large system design automation. *Progr. Comput. Softw.* **1979**, *5*, 80–90.
22. Ganter, B.; Wille, R. *Formal Concept Analysis—Mathematical Foundations*; Springer: Berlin/Heidelberg, Germany, 1999.
23. Chalmers, D.J. *Epistemic Modality*; Oxford University Press: Oxford, UK, 2011.
24. Burgin, M. Logical tools for program integration and interoperability. In Proceedings of the IASTED International Conference on Software Engineering and Applications, MIT, Cambridge, MA, USA, 9–11 November 2004; pp. 743–748
25. Motter, A.E.; de Moura, A.P.S.; Lai, Y.-C.; Dasgupta, P. Topology of the conceptual network of language. *Phys. Rev. E* **2002**, *65*, 065102.
26. Kuratowski, K. *Topology*; Academic Press: Waszawa, Poland, 1966; Volume 1.
27. Burgin, M. *Super-Recursive Algorithms*; Springer: Berlin/Heidelberg, Germany, 2005.
28. Gärdenfors, P. *Conceptual Spaces*: *On the Geometry of Thought*; MIT Press: Cambridge, MA, USA, 2000.
29. Gärdenfors, P. Conceptual spaces as a framework for knowledge representation. *Mind Matter* **2004**, *2*, 9–27.
30. Osgood, C.E.; Suci, G.J.; Tannenbaum, P.N. *The Measurement of Meaning*; University of Illinois Press: Chicago, IL, USA, 1967.
31. Bourbaki, N. *Topological Vector Spaces*, *Elements of Mathematics*; Springer: Berlin/Heidelberg, Germany, 1987.
32. Le Potier, J. *Lectures on Vector Bundles*; Cambridge Studies in Advanced Mathematics, Volume 54; Cambridge University Press: Cambridge, UK, 1997.
33. Burgin, M. Semitopological vector spaces and hyperseminorms. *Theory Appl. Math. Comput. Sci.* **2013**, *3*, 1–35.
34. Rudin, W. *Functional Analysis*; McGrow-Hill: New York, NY, USA, 1991.
35. Burgin, M.; Debnath, N. Hardship of program utilization and user-friendly software. In Proceedings of the ISCA 16th International Conference on Computer Applications in Industry and Engineering (CAINE-2003), International Society for Computers and their Applications, Las Vegas, NV, USA, 11–13 November 2003; pp. 314–317.
36. Kohlas, J.; Stärk, R.F. Information Algebras and consequence operators. *Log. Univers.* **2007**, *1*, 139–165.
37. Shoenfield, J.R. *Mathematical Logic*; Addison-Wesley: Reading, MA, USA, 1967.
38. Burgin, M. Information operators in categorical information spaces. *Information* **2010**, *1*, 119–152.
39. Burgin, M. *Information Dynamics in a Categorical Setting*, *in Information and Computation*; World Scientific: Singapore, Singapore, 2011; pp. 35–78.
40. Burgin, M. Discontinuity structures in topological spaces. *Int. J. Pure Appl. Math.* **2004**, *16*, 485–513.

41. Alexandroff, P. *Elementary Concepts of Topology*; Dover Publications, Inc.: New York, NY, USA, 1961.
42. Burgin, M. *Neoclassical Analysis*: *Calculus Closer to the Real World*; Nova Science Publishers: New York, NY, USA, 2008.
43. Dunford, N.; Schwartz, J. *Linear Operators*; Interscience Publishers: New York, NY, USA, 1958.
44. Burgin, M. Information Algebras. *Control Syst. Mach.* **1997**, *6*, 5–16. (in Russian)

© 2014 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).

proceedings

MDPI

*Proceedings*

# Prolegomena to Information Taxonomy †

**Mark Burgin [1,*] and Gordana Dodig-Crnkovic [2]**

[1] Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA

[2] Department of Applied IT, Division of Cognitive Science and Communication, University of Technology and University of Gothenburg, 41296 Gothenburg, Sweden; gordana.dodig-crnkovic@chalmers.se

* Correspondence: mburgin@math.ucla.edu

† Presented at the IS4SI 2017 Summit DIGITALISATION FOR A SUSTAINABLE SOCIETY, Gothenburg, Sweden, 12–16 June 2017.

Published: 9 June 2017

**Abstract:** We present a part of the multiscale taxonomy of information constructed by the authors. This taxonomy is a unified system of aspect taxonomies of information.

**Keywords:** information; taxonomy; classification; world; law; science; brain; emotions; will

## 1. Introduction

There is a diversity of different types and kinds of information. To organize this huge collection into a system, it is necessary to classify information with respect to various criteria developing a multiscale (multiaspect) information taxonomy, in which each dimension is an aspect information taxonomy. We construct such a multiscale (multiaspect) information taxonomy based on the general theory of information [1–3] and making use of its principles and technical tools.

It is important to understand that taxonomies are not auxiliary edifices in science but they are also laws of science when scientifically grounded and validated. For instance, the biological taxonomy of Carolous Linnaeus is a law of biology in the same way as Newton's laws are laws of physics.

Here we follow taxonomic traditions of Linnaeus Carolous Linnaeus [4] and Charles Saunders Peirce [5] in the direction of information science. On the one hand, the results of our research connect new information science and technology with classical science demonstrating intrinsic links between information theory and profound results of Linnaeus. On the other hand, these results show unity in achievements of scientists working in different countries and on different continents such as biological classification of Linnaeus, chemical classification of Mendeleev, semiotic classifications of Peirce, classifications of subatomic particles in contemporary physics and classifications in information science developed here. We begin with a brief exposition of methodological principles of taxonomy construction and then apply these principles to the development of basic information taxonomies. Here we describe only some of them due to the space restrictions.

## 2. Principles of Taxonomy Construction

Having a multiplicity of objects, it is necessary to induce organization because it can help to study, understand and utilize this multiplicity. Organization is achieved by structuration of the multiplicity. An efficient technique of structuration is construction of taxonomies, classifications, typologies and categorizations. Let us consider the process and basic principles of taxonomy construction.

Taking a multiplicity of objects $M$, a researcher explicates objects' properties molding aspects or amalgamated features of $M$. Then the researcher elucidates a criterion for each aspect. This allows us to form a scale for measuring/evaluating each aspect. Such a scale together with the corresponding

criterion allows the researcher to build an aspect taxonomy. Combining together all aspect taxonomies, the researcher obtains a multiscale taxonomy of the multiplicity $M$.

It is important to understand that according to the contemporary methodology of science, there are three types of scientific laws: classificational, equational and implicational laws. Scientists traditionally consider only two latter types as laws of nature although the first type also reflects important regularities in nature and society.

An *equational law* has the form of an equation, for example, of a differential equation as many laws in physics, e.g., $E = mc^2$, chemistry or economics.

An *implicational law* has the form of an implication "If $A$, then $B$". For instance, if $\Delta ABC$ is a right triangle, then its sides satisfy the equation $c^2 = a^2 + b^2$. It is a mathematical law called the Pythagorean theorem.

A *classificational law* has the form of a classification, typology or taxonomy. The biological taxonomy of the great biologist Linnaeus Carolous Linnaeus (1707–1778) and triadic typologies of the great logician Charles Saunders Peirce (1839–1914) are examples of classificational laws.

In addition, scientific laws can be qualitative and quantitative.

A *quantitative law* describes relations between quantitative characteristics of definite phenomena. For instance, Newton's law of motion $ma = F$ is a quantitative law of physics.

A *qualitative law* describes relations between qualitative characteristics of definite phenomena. For instance, Galilean law of motion "Everybody continues its state of rest or of uniform motion in a straight line unless it is compelled to change that state by forces impressed upon it" is a qualitative law of physics. Classificational laws are usually also qualitative laws.

These methodological findings determine a higher scientific status and importance of the groundbreaking Linnaeus' classification, as well as of the taxonomies constructed in this paper. Namely, this new understanding of scientific laws shows these taxonomies are qualitative laws of information science.

## 3. Three Basic Taxonomies of Information

We begin with the uppermost level of the taxonomic arrangement, which includes a huge diversity of types, kinds, sorts, categories and classes of information. On this level, we build the existential taxonomy

As information is an omnipresent phenomenon [6], it is crucial to start its classification on the global level of the whole world. This thesis implies the conjecture that the structure of the world affects existence of specific forms of information, which correspond to this structure. The large-scale structure of the world is represented by the Existential Triad of the World [7].

- Physical World
- Mental World
- World of Structures

In the Existential Triad, the Physical (material) World is conceived as the physical reality studied by natural sciences, the Mental World encompasses different levels of mentality, and the World of Structures consists of various forms and types of structures.

The existential stratification of the World continues the tradition of Plato with his World of Ideas [8] and the tradition of Charles Sanders Peirce with his extensive triadic classifications [5].

This stratification brings us to the phenomenon studied by the general theory of information and called information in a broad sense [3]. According to this approach, information in a broad sense is represented in each of the three worlds. In the Physical (material) World, it is called energy supporting in such a way the conjecture of von Weizsäcker that energy might in the end turn out to be information [9]. Situated at the first level of the Mental World, individual mental energy includes psychic energy studied by such psychologists as Ernst Wilhelm von Brücke (1819–1892), Sigmund Freud (1856–1939) and Carl Gustav Jung (1875–1961). Information in a broad sense, which is situated in the World of Structures, is called information in a strict sense.

### *3.1. The Global Existential Taxonomy*

As a result, we have three types of information in the global existential taxonomy:

- Physical-world information or energy
- Mental-world information and its particular case, mental energy
- Structural-world information or information per se defined as information in a strict sense

We will not analyze here the first two kinds of information in a broad sense as the first of them belongs to the scope of physics, while the second one is in the domain of psychology. Our concern is information in a strict sense or simply information.

### *3.2. The Developmental Taxonomy*

Time has three modalities—past, present and future. Information is functioning and acting in time similar to all physical processes. Therefore, it is natural to distinguish types of information related to the modalities of time.

The developmental taxonomy is brought on by the temporal aspect of information:

- Potential information is measured and evaluated by its potential actions (potential impact) on the infological system.
- Actualized information is measured and evaluated by its present impact on the infological system, i.e., changes that occurred until now, that is, in the past, in the infological system.
- Emerging information is measured and evaluated by its actions that cause changes (transformations) going in the infological system at the time of measurement/evaluation, i.e., going on in the present.

Let us consider some examples.

**Example 1.** *Information in a book before somebody reads it is potential.*

It is possible to measure potential information by its potential to make changes in the corresponding infological system. For instance, measuring potential epistemic information, we estimate (measure) potential changes in the knowledge system [10].

**Example 2.** *Information that already gave knowledge about something, e.g., information about observation of a positron obtained by Carl Anderson in 1932, is actualized.*

It is possible to measure actual information by changes it made in the corresponding infological system. For instance, measuring actualized epistemic information, we determine (measure) changes in the knowledge system made by reception of this information [10].

**Example 3.** *Information in a computer, which processes this information or in the head of a person who thinks about it, is emerging.*

It is possible to estimate emerging information by its potential to make changes, by transformations it made in the corresponding infological system and by the rate of ongoing transformations. For instance, measuring emerging epistemic information, we estimate (measure) what changes in the knowledge system have already been made and reckon the rate of ongoing changes.

### *3.3. Models of the Brain*

One more information taxonomy is based on the extended triune model of the brain elaborated in [3] as the further development of triune brain model of the Paul MacLean [11]. The main conception of this approach is existence of three levels of perception and action that are controlled by three

corresponding centers in the human brain. These three centers together form the triune brain and have the structure of a triad.

Anatomically, it is possible to structure the brain into three principal parts: hindbrain (including the spinal cord), midbrain, and forebrain. According to the theory of MacLean, the neural basis, or framework, of the brain has three parts: the spinal cord, hindbrain, and midbrain. In addition to it, centuries of evolution have endowed people with three distinct cerebral systems. The oldest of them is called the reptilian brain or R-complex, which comes from reptiles. It programs behavior that is primarily related to instinctual actions based on ancestral learning and memories.

Through evolution, living beings have developed a second cerebral system, the limbic system, which MacLean calls the paleomammalian brain and which contains hippocampus, amygdala, hypothalamus, pituitary gland, and thalamus. This system is situated around the R-complex, is shared by humans with other mammals, and plays an important role in human emotional behavior.

The most recent development of the cerebral hierarchy is called the neomammalian brain, or the neocortex. It constitutes 85% of the whole human brain mass and receives its information from the external environment through the eyes, ears, and other organs of senses. The neocortex, which consists of two hemispheres, governs people creative and intellectual functions such as social interaction and advance planning. The left hemisphere works with symbolic information, applying step-by-step reasoning, while the right hemisphere handles images processed by massively parallel (gestalt) algorithms.

The extended triune model of the brain has the following structure representing three basic systems of the brain each of which is distributed through different anatomical parts of the brain:

- the System (Center) of Rational Intelligence (also called the System of Reasoning)
- the System (Center) of Emotions (or more generally, the System of Affective States)
- the System (Center) of Will and Instinct.

### *3.4. The Formation/Action Taxonomy*

The extended triune model of the brain induces the following bifocal formation/action taxonomy/typology of information:

- Epistemic (form) or cognitive (action) information
- Instructional (form) or effective (action) information
- Emotional (form) or affective (action) information.

In this taxonomy, the first facet (term/name of each class) represents the form of the corresponding infological system, while the second facet (term/name of each class) represents action of information. It means that the first nominal attribute of the taxonomic classes characterizes formation aspects of information while the second operational attribute of the taxonomic classes characterizes procedural aspects of information.

Epistemic (cognitive) information is related to and processed by the System of Rational Intelligence or Reasoning.

Instructional (effective) information is related to and processed by the System of Will and Instinct.

Emotional (affective) information is related to and processed by the System of Affective States (Emotions).

To conclude, it is necessary to understand that here only a small part of the hierarchical multiscale (multiaspect) information taxonomy developed by the authors is represented. In essence, all described taxonomies together with other taxonomies constructed by the authors of this work form a hierarchical multiscale (multiaspect) information taxonomy, which gives a systematic picture of information.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References


1. Burgin, M. Information: Problems, Paradoxes, and Solutions. *J. Glob. Sustain. Inf. Soc.* **2003**, *1*, 53–70.
2. Burgin, M. Data, Information, and Knowledge. *Information* **2004**, *7*, 47–57.
3. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
4. Linnaeus, C. *Systema Naturae, Sive Regna Tria Naturae Systematice proPosita per Classes, Ordines, Genera & Species*; Johann Wilhelm de Groot for Theodor Haak: Leiden, The Netherlands, 1735.
5. Peirce C.S. *Collected Papers*; Cambridge University Press: Cambridge, UK, 1931–1935; Volume 1–6.
6. Burgin, M.; Dodig-Crnkovic, G. Information and Computation—Omnipresent and Pervasive. In *Information and Computation*; World Scientific: New York, NY, USA; London, UK; Singapore, 2011; pp. 7–32.
7. Burgin, M. *Structural Reality*; Nova Science Publishers: Hauppauge, NY, USA, 2012.
8. Plato. *The Collected Dialogues of Plato*; Princeton University Press: Princeton, NJ, USA, 1961.
9. Von Weizsäcker, C.F. *Die Einheit der Natur*; Deutscher Taschenbuch Verlag: Munich, Germany, 1974
10. Burgin, M. Epistemic Information in Stratified M-Spaces. *Information* **2011**, *2*, 697–726.
11. MacLean, P.D. *A Triune Concept of the Brain and Behavior*; University of Toronto Press: Toronto, ON, USA, 1973.


© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

*Proceedings*

# The General Theory of Information as a Unifying Factor for Information Studies: The Noble Eight-Fold Path †

**Mark Burgin**

Department of Mathematics, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu; Tel.: +1-818-817-3170

† Presented at the IS4SI 2017 Summit DIGITALISATION FOR A SUSTAINABLE SOCIETY, Gothenburg, Sweden, 12–16 June 2017.

Published: 9 June 2017

**Abstract:** We analyze advantages and new opportunities, which the general theory of information (GTI) provides for information studies.

**Keywords:** information; theory; axiom; principle; energy; psychic energy; ontology

## 1. Introduction

The general theory of information (GTI) is a novel approach, which offers powerful tools for all areas of information studies. The theory has three components:

- The axiomatic foundations
- The mathematical core
- The functional hull

In Section 2, we give a very brief exposition of the axiomatic foundations of the general theory of information. The mathematical core is presented in [1–7] and some other publications. In Section 3, we demonstrate advantages and new opportunities, which the general theory of information (GTI) provides for science in general and information studies, in particular.

## 2. Axiomatic Foundations of the General Theory of Information

The axiomatic foundation consists of principles, postulates and axioms of the general theory of information.

- *Principles* describe and explain the essence and main regularities of the information terrain.
- *Postulates* are formalized representations of principles.
- *Axioms* describe mathematical and operational structures used in the general theory of information.

There are two classes of principles:

- *Ontological principles* explain the essence of information as a natural and artificial phenomenon.
- *Axiological principles* explain how to evaluate information and what measures of information are necessary.

At first, we consider ontological principles.
There are three groups of ontological principles:

◊ *Substantial ontological principles* (O1, O2 and its modifications O2g, O2a, O2c) define information.
◊ *Existential ontological principles* (O3, O4, O7) describe how information exists in the physical world [8].
◊ *Dynamical ontological principles* (O5, O6) show how information functions.

**Ontological Principle O1** (the *Locality Principle*). It is necessary to separate information in general from information (or a portion of information) for a system *R*.

In other words, empirically, it is possible to speak only about information (or a portion of information) for a system. This principle separates local and global approaches to information definition, i.e., in what context information is defined.

The Locality Principle explicates an important property of information, but says nothing what information is. The essence of information is described by the second ontological principle, which has several forms.

**Ontological Principle O2** (the *General Transformation Principle*). In a broad sense, *information* for a system *R* is a capacity to cause changes in the system *R*.

Thus, we may understand information in a broad sense as a capacity (ability or potency) of things, both material and abstract, to change other things. Information exists in the form of *portions of information*.

The Ontological Principle O2 is fundamental as it intimately links information with time. Changes to *R*, when they occur by reception of information, are defined here to be the result of a causal process. Causality necessarily implies that the related effect happens after its cause. The Ontological Principle O2 leaves open the question whether the potential causal changes may or must be irreversible.

The Ontological Principle O2 unifies dynamic aspects of reality because information in a broad sense projected onto three primal components of reality—*physical reality, mental reality* and *structural reality*—amalgamates the conceptions of *information*, *physical energy* and *mental energy* with its special form, *psychic energy*, in one comprehensive concept.

Being extremely wide-ranging, this definition supplies meaning and explanation to the conjecture of von Weizsäcker that *energy might in the end turn out to be information* [9], as well as to the aphorism of Wheeler *It from Bit* [10] and to the statement of Smolin that the three-dimensional energetic world is the flow of information [11].

*Mental energy* is considered as a mood, ability or willingness to engage in some mental work and is often related to the activation level of the mind. The concept stems from an "energy of the soul" introduced by Henry More in his 1642 *Psychodia platonica*.

*Psychic energy* has become an essential component of several psychological theories. At first, the concept of *psychic energy*, also called *psychological energy*, was developed in the field of psychodynamics by German scientist Ernst Wilhelm von Brücke (1819–1892). Then it was further developed by his student Sigmund Freud (1856–1939) in psychoanalysis. Next step in its development was done by his student Carl Gustav Jung (1875–1961).

Mental energy is innate for any mentality, while psychic energy is related only to the human psyche.

**Ontological Principle O2g** (the *Relativized Transformation Principle*). *Information* for a system *R relative to the infological system* IF(*R*) is a capacity to cause changes in the system IF(*R*).

The concept of infological system plays the role of a free parameter in the general theory of information, providing for representation of different kinds and types of information in this theory. That is why the concept of *infological system*, in general, should not be limited by boundaries of exact definitions. A free parameter must really be free. Identifying an infological system IF(*R*) of a system *R*, we can define different kinds and types of information.

Here are examples from popular information theories:

In Shannon's information theory (or more exactly, a theory of communication), information is treated as elimination of uncertainty [12], i.e., as a definite change in the knowledge system of the receptor of information. In the semantic information theory of Bar-Hillel and Carnap, information

causes change in knowledge about the real state of a system under consideration [13]. In algorithmic information theory [1,14], information about a constructive object, e.g., a string of symbols, is characterized by construction of this object, while information in one object about another one reflects changes in the systems of construction algorithms.

Taking a physical system *D* as the infological system and allow only for physical changes, we see that information with respect to *D* coincides with (physical) energy.

Taking a mental system *B* as the infological system and considering only mental changes, information with respect to *B* coincides with mental energy.

Taking a cognitive system *C* as the infological system and considering only structural changes, information with respect to *B* coincides with information *per se*.

As a model example of an infological system IF(*R*) of an intelligent system *R*, we take the system of knowledge of *R*. In cybernetics, it is called the *thesaurus* Th(*R*) of the system *R*. Another example of an infological system is the memory of a computer. Such a memory is a place in which data and programs are stored and is a complex system of diverse components and processes.

The concept of an infological system shows that not only living beings receive and process information. For instance, it is natural to treat the memory of a computer as an infological system. Then what changes this memory is information for the computer.

**Ontological Principle O2a** (the *Special Transformation Principle*). *Information in the strict sense* or *proper information* or, simply, *information* for a system *R*, is a capacity to change structural infological elements from an infological system IF(*R*) of the system *R*.

There is no exact definition of infological elements although there are various entities that are naturally considered as infological elements as they allow one to build theories of information that inherit conventional meanings of the word *information*. For instance, knowledge, data, images, algorithms, procedures, scenarios, ideas, values, goals, ideals, fantasies, abstractions, beliefs, and similar objects are standard examples of infological elements. Note that all these elements are structures and not physical things. That is why, we use structural infological elements *per se* for identifying information in the strict sense.

This allows giving an esthetically eye-catching description of information:

**Information is energy in the Platonic World of Ideas.**

**Ontological Principle O2c** (the *Cognitive Transformation Principle*). *Cognitive information* for a system *R*, is a capacity to cause changes in the cognitive infological system IFC(*R*) of the system *R*.

An infological system IF(*R*) of the system *R* is called *cognitive* if IF(*R*) contains (stores) elements or constituents of cognition, such as knowledge, data, ideas, fantasies, abstractions, beliefs, etc. A cognitive infological system of a system *R* is denoted by CIF(*R*) and is related to cognitive information.

After we outlined (defined) the concept *information*, let us consider how information exists in the physical world.

**Ontological Principle O3** (the *Embodiment Principle*). For any portion of information *I*, there is always a *carrier C* of this portion of information for a system *R*.

The substance *C* that is a carrier of the portion of information *I* is called the *physical*, or *material*, *carrier* of *I*.

**Ontological Principle O4** (the *Representability Principle*). For any portion of information *I*, there is always a *representation C* of this portion of information for a system *R*.

**Ontological Principle O5** (the *Interaction Principle*). A transaction/transition/transmission of information goes on only in some interaction *of C* with *R*.

**Ontological Principle O6** (the *Actuality Principle*). A system *R* accepts a portion of information *I* only if the transaction/transition/transmission causes corresponding transformations in *R*.

**Ontological Principle O7** (the *Multiplicity Principle*). One and the same carrier *C* can contain different portions of information for one and the same system *R*.

Now we give a list of axiological principles.

**Axiological Principle A1.** A measure of information *I* for a system *R* is some measure of changes caused by *I* in *R* (for information in the strict sense, in IF(*R*)).

Note that it is possible to take the quantity of resources used for inflicting changes caused by information *I* in a system *R* as a measure of these changes and consequently, as a measure of information *I*.

**Axiological Principle A2.** One carrier *C* can contain different portions of information for a given system *R*.

**Axiological Principle A3.** According to time orientation, there are three types of measures of information: (1) potential or perspective; (2) existential or synchronic; (3) actual or retrospective.

**Axiological Principle A4.** According to the scale of measurement, there are two groups, each of which contains three types of measures of information: (1) qualitative measures, which are divided into descriptive, operational and representational measures, and (2) quantitative measures, which are divided into numerical, comparative and splitting measures.

**Axiological Principle A5.** According to spatial orientation, there are three types of measures of information: external, intermediate, and internal.

**Axiological Principle A6.** Information *I*, which is transmitted from a carrier *C* to a system *R*, depends on interaction between *C* and *R*.

**Axiological Principle A7.** Measure of information transmission from a carrier *C* to a system *R* reflects a relation (like ratio, difference etc.) between measures of information that is admitted by the system *R* in the process of transmission and information that is presented by *C* in the same process.

## 3. The General Theory of Information as a Unifying Factor for Information Studies

**First**, the general theory of information gives a flexible, efficient and all-encompassing definition of information [1,15]. In contrast to other definitions and descriptions used before, this definition is parametric allowing specification of information in general, as well as information in any domain of nature, society and technology.

Even more, the new definition taken in broad context make it possible to unite the conceptions of information, physical energy and psychic energy in one comprehensive concept. Being extremely wide-ranging, this definition supplies meaning and explanation to the conjecture of von Weizsäcker [9] that *energy might in the end turn out to be information* as well as to the aphorism of Wheeler [10] *It from Bit*.

This shows that the general theory of information provides means for a synthesis of physics, psychology and information science playing the role of a metatheory for these scientific areas.

At the same time, the new definition characterizes proper information when the general concept is specified by additional principles. The construction of an infological system allows researchers to exactly delineate information in the area of their studies.

**Second**, the general theory of information explains and makes available constructive tools for discerning information, measures of information, information representations and carriers of information. For instance, taking a letter written on a piece of paper, we see that the paper is the carrier of information, the text on it is the representation of the information contained in this text and it is possible to measure the quantity of this information using Shannon entropy or algorithmic complexity.

**Third**, the general theory of information provides efficient mathematical models. There are models of three types: information algebras, operator models based on functional analysis and operator models based on category theory. Functional representations of information dynamics preserve internal structures of information spaces associated with infological systems as their state or phase spaces. Categorical representations of information dynamics display external structures of information spaces associated with infological systems. Algebraic representations of information

dynamics maintain intermediate structures of information spaces. These models allow researchers to discover intrinsic properties of information.

**Fourth**, the general theory of information supplies methodological and theoretical tools for the development of measurement and evaluation technologies in information studies and information technology. Moreover, any science needs theoretical and practical means for making grounded observations and measurements. Different researchers in information theory have developed many methods and measures. The most popular of them are Shannon's entropy [13] and algorithmic complexity [14]. The general theory of information unifies all these approaches opening new possibilities for building efficient methods and measures in areas where the currently used methods and measures are not applicable.

**Fifth**, the general theory of information offers organization and structuration of the system of all existing information theories.

However, it is important to understand that this unifying feature and all advantages of the general theory of information do not exclude necessity in special theories of information, which being more specific, can go deeper in their investigation of properties of information and information processes in various areas. For instance, syntactic information theories, such as Shannon's theory, are very useful in the area of communication [12]. Algorithmic information theories [1,14], such as the theory of Kolmogorov complexity, are very useful in the area of automata, computation and algorithms. There are also semantic [13], economic [16], evolutionary [17], pragmatic, semiotic and other special information theories, each of which is directed at investigation of specific properties of information, information processes and systems.

**Sixth**, the general theory of information explicates the relevant relations between information, knowledge and data demonstrating that while knowledge and data are objects of the same type with knowledge being more advanced than data, information has a different type. These relations are expressed by the Knowledge-Information-Matter-Energy Square in Figure 1.

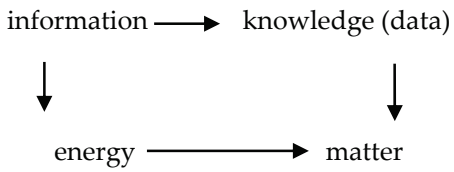



**Figure 1.** The Knowledge-Information-Matter-Energy Square.

In other words,

**information** is related to **knowledge (data)** as **energy** is related to **matter**.

In particular, it is possible to transform knowledge or data into information as we can transform matter into energy.

**Seventh**, the general theory of information rigorously represents static, dynamic and functional aspects and features of information. These features are modeled and explored by algebraic, topological and analytical structures of operators in functional spaces and functors in the categorical setting forming information algebras, calculi and topological spaces.

**Eighth**, the general theory of information explicates and elucidates the role of information in nature, cognition, society and technology clarifying important ontological, epistemological and sociological issues. For instance, this theory explains why popular but not exact and sometimes incorrect publications contain more information for people in general than advanced scientific works with outstanding results.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References


1. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
2. Burgin, M. Information Algebras. *Control Syst. Mach.* **1997**, *6*, 5–16. (In Russian)
3. Burgin, M. Data, Information, and Knowledge. *Information* **2004**, *7*, 47–57.
4. Burgin, M. Information Operators in Categorical Information Spaces. *Information* **2010**, *1*, 119–152.
5. Burgin, M. Epistemic Information in Stratified M-Spaces. *Information* **2011**, *2*, 697–726.
6. Burgin, M. Information Dynamics in a Categorical Setting. In *Information and Computation*; World Scientific: New York, NY, USA; London, UK; Singapore, 2011; pp. 35–78.
7. Burgin, M. Weighted E-Spaces and Epistemic Information Operators. *Information* **2014**, *5*, 357–388.
8. Burgin, M. Information in the Structure of the World. *Inf. Theor. Appl.* **2011**, *18*, 16–32.
9. Von Weizsäcker, C.F. *Die Einheit der Natur*; Deutscher Taschenbuch Verlag: Munich, Germany, 1974.
10. Wheeler, J.A. Information, Physics, Quantum: The Search for Links. In *Complexity, Entropy, and the Physics of Information*; Addison-Wesley: Redwood City, CA, USA, 1990; pp. 3–28.
11. Smolin, L. *The Life of the Cosmos*; Oxford University Press: Oxford, UK; New York, NY, USA, 1999.
12. Shannon, C.E. The Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423.
13. Bar-Hillel, Y.; Carnap, R. Semantic Information. *Br. J. Philos. Sci.* **1958**, *4*, 147–157.
14. Chaitin, G.J. Algorithmic information theory. *IBM J. Res. Dev.* **1977**, *21*, 350–359.
15. Burgin, M. Information: Concept Clarification and Theoretical Representation. *TripleC* **2011**, *9*, 347–357.
16. Arrow, K.J.; Marschak, J.; Harris, T. Economics of information and organization: Optimal inventory policy. *Econometrica* **1951**, *19*, 250–272.
17. Burgin, M. Evolutionary Information Theory. *Information* **2013**, *4*, 224–268.


© 2017 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

information

MDPI

Article

# Structural and Symbolic Information in the Context of the General Theory of Information

**Mark Burgin [1,*] and Rainer Feistel [2,*]**

[1] Department of Mathematics, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA
[2] Leibniz Institute for Baltic Sea Research, D-18119 Warnemünde, Germany
* Correspondence: mburgin@math.ucla.edu (M.B.); rainer.feistel@io-warnemuende.de (R.F.)

Received: 26 September 2017; Accepted: 1 November 2017; Published: 6 November 2017

**Abstract:** The general theory of information, which includes syntactic, semantic, pragmatic, and many other special theories of information, provides theoretical and practical tools for discerning a very large diversity of different kinds, types, and classes of information. Some of these kinds, types, and classes are more important and some are less important. Two basic classes are formed by structural and symbolic information. While structural information is intrinsically imbedded in the structure of the corresponding object or domain, symbolic information is represented by symbols, the meaning of which is subject to arbitrary conventions between people. As a result, symbolic information exists only in the context of life, including technical and theoretical constructs created by humans. Structural information is related to any objects, systems, and processes regardless of the existence or presence of life. In this paper, properties of structural and symbolic information are explored in the formal framework of the general theory of information developed by Burgin because this theory offers more powerful instruments for this inquiry. Structural information is further differentiated into inherent, descriptive, and constructive types. Properties of correctness and uniqueness of these types are investigated. In addition, predictive power of symbolic information accumulated in the course of natural evolution is considered. The phenomenon of ritualization is described as a general transition process from structural to symbolic information.

**Keywords:** general theory of information; structural information; symbolic information; correctness; uniqueness; information conservation; prophecy; evolution; ritualization

## 1. Introduction

Everything has its structure, and according to the contemporary approach in methodology of science, scientists discover structure, rather than the essence and nature of the studied phenomena (cf., for example, [1–3]). As a result of their research, scientists gain and publish information obtained from the structure under investigation. For instance, astronomers, such as Tycho Brahe and Johannes Kepler, watched the changing positions of visible stars and, eventually, in 1627, published related tables with the numbers that described the structure of our planetary system. Extracted structural information was symbolically described by numbers making possible the discovery of dynamic regularities of the solar system made by Kepler. Formulated in terms of mathematical symbols, Kepler's laws permitted, for example, successful predictions of the solar transits of Mercury in 1631 and of Venus in 1639, and eventually the launch of satellites and space probes.

To achieve better understanding of information processes, it is useful to study relations between *structural* and *symbolic information* [4–8], or, synonymously, between *bounded* and *free information* [9], respectively. Symbolic information relies on a convention between the transmitter and receiver. In contrast to this, intrinsic structural information, on the contrary, is not based on conventions. For instance, when astronomers watch the changing brightness of distant stars, they receive structural

information about possible planets of those suns. At the same time, when sailors at night watch the changing brightness of a distant lighthouse at the shore, they receive both structural and symbolic information in the form of the optical code as a unique identifier of the beacon.

It is possible to attribute structural information to all physical processes and systems. It may be often quantified in terms of physical entropy [5,8,10–12]. Physical structures, such as moving planets and their shining light, exist independently of life. On the other hand, the prediction of future events, such as solar transits, derived from observations made in the past, is a typical characteristic of life and, in particular, of humans. In order to facilitate future survival, genetic information also acts this way, as "genomes do not predict the future but recall the past: they reflect the exigencies of history" [13]. Information constitutes a bridge between lifeless physics and living beings struggling for survival. It is obtained by "measurement in which a rate-dependent dynamical state is coded into quiescent symbols" [14], and created by trial and error in the course of Darwinian evolution. The emergence of symbolic information may "distinguish the living from the lifeless", as the problem was raised already by Pearson ([15], p. 341). In this context, symbolic information has a purpose, while structural information does not. "Evolution is based on function and—what is really new in physics—the 'purposefulness' of the function involved" [16].

In many cases, portions of symbolic information may be true or false. A lighthouse on the shore, as the transmitter of information, may send a wrong code either as a result of a technical failure or intentional deception by the operators on land. Additionally, the sailor, as the receiver of information, may fail in properly reading or interpreting the code. In the case of structural information, however, if an alien astronomer observes the 11-year brightness cycle of our sun, he may falsely conclude that there is a big planet with an 11-year orbital period, similar to Jupiter. In that case, information extracted by the scientist from the measured signal may be false, but one may not reasonably blame the changing sunlight itself to be wrong, as in the case of the lighthouse code. There may also be "fuzzy" cases of unclear right or wrong, such as "Tabby's star", where the physical structure causing the received light signal remains elusive [17].

The examples above demonstrate, as in the case of light received from a distant source, that the truth of information carried by a light beam is related to the structure of the source, to the structure of the light itself as the transmission medium, as well as to the structure or structural change of the receiver. All these elements represent certain forms of structural information that are physically converted from one into another during the information transfer process.

Before looking in more detail at the roles of the different parts of an information transfer process, we introduce (Section 2) the relations between structures, symbols, information and prediction in the context of the general theory of information (GTI) (cf. for example, [18–21]) and the general theory of structures (GTS) developed in [1]. Different types of information and its comprehension are carefully differentiated in our exploration of structural information presented in Sections 3 and 4. Section 3 deals with descriptive structural information while Section 4 examines inherent structural information. In Sections 5 and 6, we study symbolic information comparing it to structural information.

## 2. Basic Postulates of the General Theory of Information (GTI)

The general theory of information (GTI) has been constructed by Mark Burgin utilizing the basic postulates of information theory and information technology [18,22,23]. These postulates are presented in the form of principles. They allow to achieve the comprehensive definition of information as a foremost phenomenon in the world, to discern information, information representation and information carrier and to efficiently study information measures, processes, and systems by means of mathematical models and experiments (cf. for example, [19–21]).

**Ontological Principle O1 (the *Locality Principle*).** It is necessary to separate information, in general, from information (or a portion of information) for a system $R$.

In other words, empirically, it is possible to speak only about information (or a portion of information) for a system. This principle separates local and global approaches to information definition, i.e., in what context information is defined.

Here a portion of information means information that can be constructively or analytically separated from other information although relations and connections to usually exist.

The Locality Principle explicates an important property of information, but says nothing of what information is. The essence of information is described by the second ontological principle, which has several forms.

Ontological Principle O1 does not necessarily imply that the "system" under consideration is a physical system, somehow localized in space, exchanging information with other systems "out there". The "system" may as well be a mathematical or philosophical construct, such as an abstract automaton or a learning matrix, that exists "beyond space and time" in a sense that for its functioning, physical coordinates of the system's elements are irrelevant, unspecified aspects. However, in the interplay between structural and symbolic information, we shall focus here on systems that physically exist in space and time, at least as thought experiments.

**Ontological Principle O2 (the *General Transformation Principle*).** In a broad sense, *information* for a system $R$ is a capacity to cause changes in the system $R$.

Thus, we may understand information in a broad sense as a capacity (ability or potency) of things, both material and abstract, to change other things. Information exists in the form of *portions of information*.

Ontological Principle O2 is fundamental as it intimately links information with time. Changes to $R$, when they occur by reception of information, are defined here to be the result of a causal process. Causality necessarily implies that the related effect happens strictly after its cause, if observed from any physically possible reference frame in the sense of Einstein's special theory of relativity. Ontological Principle O2 leaves open the question of whether the potential causal changes may or must be irreversible. For example, revealing a secret to somebody $R$ has the capacity to change $R$, and if so, this change is irreversible for the impossibility to take back the secret.

However, the common usage of the word *information* does not imply such wide generalizations as Ontological Principle O2 implies. Thus, we need a more restricted theoretical meaning because an adequate theory, whether of the information or of anything else, must be in significant accord with our common ways of thinking and talking about what the theory is about, else there is the danger that theory is not about what it purports to be about. To achieve this goal, we use the concept of an *infological system* IF($R$) of the system $R$ for the information definition. Elements from IF($R$) are called *infological elements*.

The exact definition consists of two steps. At first, we make the concept of information relative and then we choose a specific class of infological systems to specify information in the strict sense. That is why it is impossible and, as well as, counterproductive to give an exact and thus, too rigid and restricted definition of an infological system. Indeed, information is a very rich and widespread phenomenon to be reflected by a restricted rigid definition (cf. for example, [24,25]).

The concept of the infological system plays the role of a free parameter in the general theory of information, providing for representation of different kinds and types of information in this theory. This is why the concept of *infological system*, in general, should not be limited by boundaries of exact definitions. A free parameter must really be free. Examples of infological systems are a dictionary, system of knowledge, thesaurus, database, scientific theory, and encyclopedia.

Identifying an infological system IF($R$) of a system $R$, we can define information relative to this system. This definition is expressed in the following principle.

**Ontological Principle O2g (the *Relativized Transformation Principle*).** *Information* for a system $R$ *relative to the infological system* IF($R$) is a capacity to cause changes in the system IF($R$).

When we take a physical system *D* as the infological system and allow only for physical changes, we see that information with respect to *D* coincides with (physical) energy. This kind of information is structural information as physical changes are usually changes of structures.

Taking a mental system *B* as the infological system and considering only mental changes, information with respect to *B* coincides with mental energy. This kind of information is symbolic information if we assume that mental processes operate with symbolic representations of information such as nerve-pulse frequencies or synaptic transmission of symbolic signal molecules. However, it is necessary to keep in mind that there are theories that describe mental processes as subsymbolic operations.

As a model example of an infological system IF(*R*) of an intelligent system *R*, we take the system of knowledge of *R*. In cybernetics, it is called the *thesaurus* Th(*R*) of the system *R*. Another example of an infological system is the memory of a computer. Such a memory is a place in which data and programs are stored and is a complex system of diverse components and processes.

The concept of an infological system allows showing that not only living beings receive and process information. For instance, it is natural to treat the memory of a computer as an infological system. Then, what changes this memory is information for a computer.

There is no exact definition of infological elements although there are various entities that are naturally considered as infological elements as they allow one to build theories of information that inherit conventional meanings of the word *information*. For instance, knowledge, data, images, algorithms, procedures, scenarios, ideas, values, goals, ideals, fantasies, abstractions, beliefs, and similar objects are standard examples of infological elements. Note that all these elements are structures and not physical things. Thus, we can consider structural infological elements per se and use them for identifying information in the strict sense.

**Ontological Principle O2a (the *Special Transformation Principle*).** *Information in the strict sense* or *proper information* or, simply, *information* for a system *R*, is a capacity to change structural infological elements from an infological system IF(*R*) of the system *R*.

An infological system IF(*R*) of the system *R* is called *cognitive* if IF(*R*) contains (stores) elements or constituents of cognition, such as knowledge, data, ideas, fantasies, abstractions, beliefs, etc. A cognitive infological system of a system *R* is denoted by CIF(*R*) and is related to cognitive information.

**Ontological Principle O2c (the *Cognitive Transformation Principle*).** *Cognitive information* for a system *R*, is a capacity to cause changes in the cognitive infological system IFC(*R*) of the system *R*.

A special case of cognitive information is epistemic information, which causes changes in the system of knowledge treated as the infological system.

After we outlined (defined) the concept *information*, let us consider how information exists in the physical world.

**Ontological Principle O3 (the *Embodiment Principle*).** For any portion of information *I*, there is always a *carrier* *C* of this portion of information for a system *R*.

The substance *C* that is a carrier of the portion of information *I* is called the *physical*, or *material*, *carrier* of *I*.

**Ontological Principle O4 (the *Representability Principle*).** For any portion of information *I*, there is always a *representation* *Q* of this portion of information for a system *R*.

Note that any representation of information is its carrier, but not any carrier is its representation. For instance, a text printed on a piece of paper is both carrier and representation of information, but the piece of paper where this text is written is only a carrier but a representation.

**Ontological Principle O5 (the *Interaction Principle*).** A transaction/transition/transmission of information goes on only in some interaction of the carrier *C* with the system *R*.

Note that interaction can be not only physical.

**Ontological Principle O6 (the *Actuality Principle*).** A system *R* accepts a portion of information *I* only if the transaction/transition/transmission causes corresponding transformations in *R* or in its infological system IF(*R*).

For instance, only people with sufficiently high level of knowledge and intelligence will be able to obtain information from it when they will read this paper.

**Ontological Principle O7 (the *Multiplicity Principle*).** One and the same carrier *C* can contain different portions of information for one and the same system *R*.

## 3. Structural Information in the Context of the General Theory of Information

It is possible to comprehend structural information in different ways. For instance, Bates [26] treats structural information as "the pattern of organization of matter and energy", while Reading [27] defines it as "the way the various particles, atoms, molecules, and objects in the universe are organized and arranged".

At first, we consider the approach developed in the general theory of information. The general theory of information discerns information in the broad sense (Ontological Principle O2) and information in the strict sense (Ontological Principle O2a).

Structural information is a kind of information in the strict sense being defined as a capacity to change the subsystem of knowledge in an intelligent system.

It means that structural information is a kind of epistemic information, which, in turn, is a type of cognitive information (cf. Section 2).

It is possible to differentiate three types of structural information.

1. Inherent structural information is information in structures.
2. Descriptive structural information is information about structures.
3. Constructive structural information is information that allows building knowledge about structures.

In this section, we describe properties of descriptive structural information while inherent structural information is studied in the next section.

These definitions allow obtaining key properties of structural information. Let us consider some of them.

1. Structural information can be more correct or less correct.

Correctness of structural information about a system depends on correctness of knowledge produced by this information [18]. As we know, some knowledge can be more correct, better representing the essence of the system, while other knowledge is less correct, providing a worse representation of the fundamental nature of the system.

Here are two examples:

**Example 1.** *For a long time, people believed that the Earth was flat, i.e., it had the structure of a plane.*

*Then scientists found that the Earth had the structure of a ball.*

*Then scientists assumed that the Earth had the structure of a geoid.*

**Example 2.** *For a long time, people believed that in the structure of the solar system, the Sun rotated around the Earth.*

*Then scientists found that the Earth rotated around the Sun and the orbit had the structure of a circle.*

*Then scientists assumed that the Earth rotates around the Sun and the orbit had the structure of an ellipse.*

2. As a rule, structural information about a system is not unique.

Many researchers believe that each (e.g., a natural) system has a unique structure. At the same time, according to the general theory of structures [1], any system has several structures. For instance, the structure of a table on the level of molecules is essentially different from the structure of this table on the level of its parts, such as legs. In essence, material systems, which people can see with their eyes and touch with their hands, have structural information on different levels.

3. Structural information about a system can be inherent to this system, built into the interaction with the system or innate for an image of the system.

Indeed, as it is stated above, structure makes things such as they are. Naturally, structural nformation reflects this identity of things although structural information about different systems and objects can be similar.

4. Processes in a system can change structural information about this system.

Indeed, the evolution (development) of a system can produce an essentially new structure when the system is changed, even becoming another system. For instance, butterflies have the four-stage life cycle. In it, winged adults lay eggs, which later become caterpillars, which later pupate in a chrysalis, while at the end of the metamorphosis, the pupal skin splits and a butterfly flies off.

5. Structural information about a system describes this system to a definite extent of precision, i.e., structural information can be more precise and less precise.

For instance, the Copernican model (structure) of the solar system is more precise than the Ptolemaic model (structure) of the solar system. Another example comes from mathematics where mathematicians are striving to find the decimal structure of the number $\pi$ with higher and higher precision.

6. For complex systems, it is possible to consider structural information on different levels and various scales.

For instance, it is possible to treat the structure of a human being on the level of visible parts, on the level of its functional systems, on the level of inner organs, on the level of cells, on the level of chemical compounds or on the level of molecules.

7. Structural information about a subsystem of a system is not always a part of the structural information about this system.

For instance, when we consider an organism as a system of its visible parts, the structure of its nervous system is not a part of this structure.

8. The process of conversion of structural information about a system into knowledge about this system is, in essence, structuration of this system.

Indeed, people get information about different objects in the form of raw data. Only after reception of this information, the brain converts these data into knowledge and this knowledge is often about the structure of studied objects. It is natural to treat data as symbolic representation of structural information, where correct structural information may be represented by symbolic information only within finite uncertainty, or even incorrectly. This may happen in the measuring device, or in the transmission or storage, or in the brain, etc.

Note that the general theory of information provides other possibilities for defining structural information. For instance, it can be information that changes the system of beliefs of an intelligent system.

## 4. Structural Information as an Intrinsic Property

At the same time, Feistel and Ebeling suggest the vision of structural information, in which structural information may no longer be restricted to changing just "knowledge in an intelligent system", and may more generally be defined as the capacity of a physical system, the "carrier of structural information", to cause changes in a second physical system, the "receiver of structural information" [4,5,7,8].

If in particular, the receiver is the same system as the carrier but at some later point of time, reversible microscopic dynamics described by the Liouville equation is universally understood as "conserving [microscopic] [structural] information" [28–30]. In contrast to this, irreversible macroscopic dynamics is commonly associated with a loss of [macroscopic] [structural] information, directly related to the growth of thermodynamic entropy (cf. [7,8]). In the sense of Planck [31], who wrote that "a macroscopic state always comprises a large number of microscopic states that combine to an average value", macroscopic structural information represents a portion of the microscopic structural information of a given system. This approach is consistent with the approach from Section 3. Namely, a system may have different structures, such as a microscopic structure and a macroscopic structure. In each of those structures, there exists related structural information. While the microscopic information may be conserved, the macroscopic information may decay according to the 2nd law of thermodynamics. That is, the different structures of the same system may possess different symmetries and may follow different laws. Irreducibilities of this kind indicate *emergent* properties or structures (see [8] and Section 5).

In his famous article "The RNA world" about the origin of life, Gilbert [32] wrote about "the useful distinction between information and function," that "information storage needs to be one-dimensional, for ease of copying, but molecules with enzymatic functions tend to be tight three-dimensional structures". This quotation demonstrates that the term "information" is in common scientific use in situations in which "knowledge in an intelligent system" does not exist at all, such as before life emerged on Earth. Gilbert refers to the structural information given by the sequence of building blocks in a chain molecule. This information is received later by the same molecule when it folds into a functioning enzyme, and by other reactants then catalyzed by it. Similarly, the information scientist Fuchs-Kittowski [33] speaks about "information" that was generated in evolution for the first time when life began, and has allowed living structures to survive to the present day. Without retaining this structural information, living beings would be as astronomically rare among random chemical mixtures [34], as are sensible books among the random texts in Jorge Luis Borges' "Library of Babel" [35].

Structural information available from a carrier depends on the receiver determining what portion of this information is actually received. If, for example, the receiver is a thermometer and the carrier is liquid, then all information received is the temperature of the liquid displayed by this thermometer. In essence, structural information can be extracted from a given system by "measurement" when, for example, a sensor is used as a receiver. Structural information also can be quantified when it is comparable to the structural information of a reference system and is reflected in a scale, such as the length scale of a mercury thermometer.

A numerical value being the result of a comparison between the same kinds of structural information available from two different systems, such as by counting their parts, is a "measurement result". Numbers represent information in the symbolic form, or as "symbolic information", see Section 5. The meaning of symbolic information is subject to convention (such as what "reference" system is used) and is no longer a portion of the structural information of the carrier, such as printed symbols on a sheet of paper. Very different structural information carriers can carry the same symbolic information. Symbolic information is restricted to the realm of life [4,7,8], such as having the form of genetic DNA molecules or human knowledge, and emerged from structural information in the course of evolution by a transition process regarded as ritualization.

The origin of life is a famous example for the ritualization transition from structural to symbolic information. Organic chain molecules, see the quotation of Gilbert [32] above, carried structural information in the form of certain sequences of their elements. Some of those resulted in catalytic properties that supported the copying of the chain molecule, such as in an RNA-replicase cycle [7,16,32]. In the course of evolution, the initial chain molecules developed to carriers of symbolic information in a complex multi-step process [7]. Similar features of the ritualization transition are observed, for example, when humans began to speak, or when digital computers replaced mechanical calculators.

The codons of the modern genetic code are the symbols used for representation of genetic information, and the phenotypic traits of the developed organism represent the meaning of that information. The symbols also carry structural information, but due to the arbitrariness of symbols in a mature coding system, this structural information is irrelevant for the meaning they express. The properties of the ink used by Mozart for writing down his music are irrelevant for the beauty of its acoustic performance.

The main distinction between structural and symbolic information is that the meaning of structural information is inseparably bound to its physical structure, while symbolic information is free insofar as arbitrary symbols may take over one and the same role in the information processing system. In this sense, symbolic information possesses the code symmetry, or coding invariance, that emerges during the ritualization transition [6,36]. The structural information of symbols that preserves traces of the evolution of the symbolic information-processing system is related to the code symmetry. Therefore, it is known that many spoken words have onomatopoetic roots. The similarity of words used in the various languages provides structural information about the migration histories of the populations. In the same way, the physical and chemical properties of the genetic apparatus carry structural information about the beginning of life in the ancient past.

## 5. Symbolic Information

Another important type of epistemic information is symbolic information.

**Definition 1.** *Symbolic information is information that is represented by systems of symbols.*

Note that it is possible that such a system may consist of one symbol.

For instance, texts form an important type of symbolic information. A printed or written text is typically a physical structure consisting of dark and light dots on the paper. The information carried by the text is in no way reducible to the physical properties of the given spatial distribution of dye or brightness. In this sense, symbolic information is an emergent property of texts.

Usually, the term *symbol* has two main meanings. In semiotics, a symbol is treated as a cognitive structure, which is a special case of signs. On the other hand, a symbol can mean a physical object written on a piece of paper, printed in a book, or displayed on the screen of a computer.

An *emergent property* is a property that exists as a process of its exposition. In this context, *emergence* conveys the empirical fact that "the whole is more than the sum of its parts" [37]. Now, many metaphorically express this phenomenon by the expression 1 + 1 = 3 (cf. for example, [38–40]). As a logical category, an emergent property can be defined as a property that is "novel and robust relative to some natural comparison class" [41]. Less rigorously stated, the term "emergence is broadly used to assign certain properties to features we observe in nature that have certain dependence on more basic phenomena (and/or elements), but are in some way independent from them and ultimately cannot be reduced to those other basic interactions between the basic elements" [42].

Consequently, as a result of the fundamental irreducibility of symbolic information to the structural information of its carrier symbols, which exists in alphabetic languages, such as English or German, it is a mere agreement which particular symbol is in use to represent a certain meaning. If everybody understood under the word "green" the color of the sky and under "blue" the color of leaves, all communication would work quite the same [43]. This explicates invariance as the new

fundamental code symmetry of symbolic information with respect to substitution. If we change knowledge of those who communicate, then symbols may be substituted by physically different symbols without affecting the meaning of the message. This invariance is in striking contrast to structural information, where a different physical structure always implies different information.

In contrast to this, in pictographic and logographic languages, such as ancient Egyptian, Chinese, or Japanese, structure of symbols essentially influences their meaning. Such languages represent intermediate stages of the ritualization transition, from original iconic, "bound" structural descriptions, which do not require any special reading skills, to abstract, "free" symbolic descriptions such as computer bits, which cannot be deciphered without knowing the coding conventions [6]. There exist various other incompletely ritualized information systems, for example, human emotional tears.

The code symmetry has several important and general implications which elucidate the reasons for the self-organized emergence of symbolic information, repeatedly and in various forms, in natural evolution history. Symbols may alternatively appear as sequences, such as in languages, or as communication signals, such as neuronal transmitter substances or human gestures. Here are some of those key features:

(i) Discrete symbols are robust against small perturbations, i.e., symbols may be replaced by similar imitations. In simple information-processing systems, the receiver may be a dynamical system to which an incoming symbol appears as an imposed boundary or initial condition. Then, the system will approach an associated attractor state, which physically represents the meaning of the symbol. Often, the *attraction basin*, i.e., the set of conditions leading to the same attractor, is a compact set, and slightly modified symbols within that basin will cause the same attractor to be reached. As an example, written letters are recognized as being equal even if their symbols are displayed in different fonts, sizes, or colors. However, irregular handwriting, distortion or damage can essentially change interpretation of symbols by the receiver.

(ii) Reading a symbol can refresh it, permitting largely *lossless copies* to be produced if the refreshment happens within the physical lifetime of the symbol(s). Multiplying cells and organisms, but also computer memories, implement the refreshment technique for safe long-term data storage.

(iii) Robustness against small symbol perturbations permits *dialects* to evolve which increasingly use modified symbols that are similar in the sense that upon reading, they produce the same results as their originals. In turn, this process permits gradual deformation of the attraction basin, or even spawning of new basins, that is, *drift and diversification* of symbols; "If signals are under selection for efficient transmission between signaler and receiver, then populations inhabiting environments that differ in their effect on signal transmission or detection are expected to evolve different signals—a process known as 'sensory drive'" [44].

(iv) Symbolic information is *conventional*. A system of symbols may be replaced by a completely different set of symbols if this transformation is simultaneously applied to the message, the transmitter and the receiver. On a Chinese tablet computer, Chinese letters, their Latin transcription, and the related binary machine code are permanently exchanged by one another while a tablet is used. Genetic DNA or RNA bases, together with their complementary strains, represent the same information. Symbolic information is invariant against such arbitrary symbol transformations (substitutions).

(v) The replacement of a symbol by a physically different one, either with the same or with a different meaning, is energetically practically neutral. Symbols are "energy-degenerate" [14]. Any forces driving a modified message back to some fictitious distinguished "equilibrium message" are virtually absent. Physically formulated, so-called *Goldstone modes* with vanishing Lyapunov exponents appear along with the emergence of symbols (a process termed the *ritualization transition* [7,8], and permit exceptionally large fluctuations. Thermodynamically, particular messages appear as alternative "microstates" that populate a "Boltzmann shell" of an information processing system; "In principle, a sequence of symbols—such as the text of a given novel—represents a microstate" [16]. In fact the famous Boltzmann formula for the

thermal entropy, $S = k \log W$, of an equilibrium system with $W$ microstates equals Shannon's formula for the information capacity if converted to the unit "bit" [5].

(vi) As a result of the coincidence of structural and symbolic information immediately at the ritualization transition, in the Goldstone modes the structural information of the symbols keep a *trace of the evolution history* of the symbolic information system, until this trace may gradually be eroded by fluctuations and neutral drift. The physical form of symbols expresses and reveals their *historicity*.

(vii) Looking at symbols from the symbolic side, the code symmetry *impedes conclusions* to be drawn from the meaning of the information on the physical properties of symbols. Running a computer program does not permit deciding whether the memory bits are stored by, say, charging electrical capacitors or swapping magnetic fields. Introspection of our mind while thinking does not offer any clues on which transmitter substances may be released between synapses, or on the nature of nerve pulse propagation; "The faculty with which we ponder the world has no ability to peer inside itself or our other faculties to see what makes them tick" ([45] p. 4).

(viii) Looking at symbols from the structural side, the code symmetry *impedes conclusions* to be drawn from the structure of the symbols on the meaning of the symbolic message. "Information and computation reside in patterns of data and in relations of logic that are independent of the physical medium that carries them" ([45] p. 24). This means, *symbolic information is an emergent property*. The same message may be expressed by different symbols, tokens or languages; a sequence of symbols may be reversibly compressed or redundantly inflated without affecting the meaning of the message. In order to produce a cup of coffee, a single on/off bit may be sent to a coffee machine, or a long instruction may be given to an unexperienced cook to prepare an equivalent result. The same mathematical problem may be solved by very different program codes whose mutual equivalence remains elusive without knowledge about the receiver, namely the rules how to compile and execute the code and to convert the message back into structural information. This position differs from opinions like that of the sharp thinker Pearson ([46] p. 50) that "we may say ... without any dogmatic assumption that psychical effects can all be *reduced* to physical motion".

(ix) Added redundancy, such as partial repetition or embedded grammatical rules combined with orthographic vocabularies, leaves the meaning of symbolic information immediately unaffected but allows additional information-protection tools to evolve for *error-detection* and *-correction* of random perturbations. During later stages after the ritualization, such tools partially counteract the neutral drift of symbols and constrict the set of available Goldstone modes. About half of written English text represents syntactic redundancy [47].

(x) Information processing of discrete symbols is performed by digital computers of any physical kind. Although Turing asserted, "This special property of digital computers, that they can mimic any discrete state machine, is described by saying that they are *universal* machines" ([48] p. 441), this is not true because digital computers cannot simulate (mimic) discrete state machines with an infinite number of states. Some think that computational universality suggests the possibility of simulating the human brain on an electronic computer [49]. However, this is also incorrect because nobody proved that the human brain is a discrete state machine. Moreover, it was demonstrated that even universal Turing machines are not *universal* in the realm of algorithms and automata because there are super-recursive algorithms, which are more powerful than any Turing machine [50,51].

Code symmetry, or coding invariance, is a key property of symbolic information that can be established after a complete ritualization transition. The physical carriers of symbols possess structural information apart from their symbolic meaning. Their physical structures maintain percussions of the symbols' evolution history over a certain characteristic relaxation time until random noise in the language development erases those traces.

Symbolic information has a number of general properties (Feistel and Ebeling, 2011):

(i) Symbolic information systems possess a kind a partial symmetry, the carrier invariance In many situations, it is possible to copy information without loss to other carriers or to multiply it in the form of an unlimited number of physical instances. The information content seems independent of the physical carrier used. However, many linguists doubt a possibility of exact translation of the text from one natural language to another one (cf. for example, [52,53]). This problem is also apparent as the absence of automatic translators between high-level programming languages, such as Fortran, Pascal, or C, demonstrates.
(ii) Symbolic information systems possess a new symmetry, the coding invariance. The functionality of the processing system is unaffected by substitution of symbols by other symbols as long as unambiguous bidirectional conversion remains possible. In particular, the stock of symbols can be extended by the addition of new symbols or the differentiation of existing symbols. At higher functional levels, code invariance applies similarly also to the substitution of groups of symbols, synonymous words or of equivalent languages.
(iii) Within the physical relaxation time of the carrier structure, discrete symbols represent quanta of information that do not degrade and can be refreshed unlimitedly.
(iv) Imperfect functioning or external interference may destroy symbolic information but only life-based processing systems can generate new or recover lost information.
(v) Symbolic information systems consist of complementary physical components that are capable of producing the structures of each of the symbols in an arbitrary sequence upon writing, of keeping the structures intact over the duration of transmission or storage, and of detecting each of those structures upon reading the message. If the stock of symbols is subject to evolutionary change, a consistent co-evolution of all components is required.
(vi) Symbolic information is an emergent property; its governing laws are beyond the framework of physics, even though the supporting structures and processes do not violate physical laws.
(vii) Symbolic information often has a meaning or purpose beyond the scope of physics.
(viii) In their structural information, the constituents of the symbolic information system preserve a frozen history ("fossils") of their evolution pathway.
(ix) Symbolic information processing is an irreversible, non-equilibrium processes that produces entropy and requires free-energy supply.
(x) Symbolic information is encoded in the form of structural information of its carrier system. Source, transmitter and destination represent and transform physical structures.
(xi) Symbolic information exists only in the context of life although this life can be natural or artificial.

Symbolic information is indispensable in the symbolosphere, which is situated as the highest level of existence finalizing the hierarchy *biosphere-sociosphere-noosphere-ideosphere-symbolosphere* [54–56].

Emergence of the *symbolosphere* is related to the development of language. The first oral or signed languages probably changed their form rapidly, leading to a multitude of language systems; "The diversity of human languages is a great example of the arbitrariness of human cultural history" ([57] p. 256). Then, about 5000 years ago, writing developed from iconic pictures in a ritualization process, essentially as a technology that amplified and made persistent the oral, nonmaterial, and invisible language component of the symbolosphere. The symbolosphere also includes, of course, mathematics, painting, music, sculpture, and photography, etc. In general, we can define symbolosphere as a component of our world in which symbols emerge, symbolic systems develop, function, and interact, and where symbolic interaction of people goes on. The symbolosphere includes all living beings that use symbols, as well as their technical and abstract creations.

Mathematics gives the most advanced example of a symbolosphere domain as in it symbolism is made explicit, achieving very high levels of abstraction. Formalism is the most extreme approach going in this direction. The main thesis of formalism is that mathematical statements are not about anything material, but are rather to be regarded as meaningless marks. The formalists are interested in the rules

that govern how these marks are manipulated. Mathematics, in other words, is the manipulation of symbols. The fact that (a + b) + c = a + (b + c) is simply a rule of the system. The principle protagonist of this philosophy was David Hilbert.

However, many mathematicians disputed this approach. For instance, Gödel in 1961 wrote that the certainty of mathematics is to be secured not by proving certain properties by a projection onto material systems—namely, the manipulation of physical symbols but rather by cultivating (deepening) knowledge of the abstract concepts themselves, which lead to the setting up of these mechanical systems, and further by seeking, according to the same procedures, to gain insights into the solvability and the actual methods for the solution of all meaningful mathematical problems. Being a Platonist, Gödel represents another extremity in philosophy of mathematics, postulating independent existence of abstract mathematical objects in the sense of Plato Ideas. However, reality of mathematical objects and other structures was supported by philosophical and scientific analysis of reality as a whole [1,58,59].

In a more recent time, electronic technologies have been developed that further amplify the symbolosphere: the telephone, the telegraph, radio, television, fax, the Internet. A storm in the symbolosphere can have the same personal consequences as a storm in physiosphere. This world has a life of its own and cannot be controlled by "operationalizing our definitions", "using language carefully", or attempting to wall off language from "dangerous outside influences". The symbolosphere is subject to manipulation, but all attempts to control it eventually fail.

This realm of our existence must be viewed as part of an ecology that also includes the biological and physical world. Language is but one part of the symbolosphere, and grammar is an even smaller part. In the future, we will explore these ideas in detail, via radio, television, the fax, satellite-enhanced communication, and most recently, the Internet. All these technologies amplify the symbolosphere and maintain it as an open system in far-from-equilibrium states.

Humans inhabit the symbolosphere as much as the biosphere. These spheres of human existence are not separate: they intersect and interact. We must know how to deal with the vagaries of the symbolosphere, just as we deal with the vagaries of the physiosphere (i.e., weather, climate, radiation, tornado, typhoons, earthquakes, etc.).

It is necessary to discern symbolic information from *symbolical* information.

**Definition 2.** *Symbolical information is information that provides and/or changes knowledge about symbols.*

In other words, symbolical information acts on (impacts) the infological systems that contain knowledge about symbols. For instance, information on how to write letters from the English alphabet is symbolical information.

If we consider words as symbols, then information about meaning of words is symbolical. In particular, definitions of terms contain symbolical information. In essence, understanding of symbolic information is essentially based on symbolical information. Namely, understanding information coded by symbols demands understanding the meaning of these symbols.

## 6. Relations and Interactions between Symbolic Information and Structural Information

There are intrinsic relations and operational interaction between symbolic information and structural information. For instance, exploring the structure of the surrounding world by trial and error using mutation and selection in the course of Darwinian evolution, biological species accumulate in their gene pools symbolic information as successful recipes for their survival; "The theory of life is a theory for the generation of information" [16]. At the same time, symbolic information in genes is also structural information as it predetermines the inner structure of the evolving organism.

Similarly, scientists observe and investigate the structures encountered in nature and in laboratory experiments. Structural information is extracted and converted into symbolic information, in the form of articles and books, data tables, or oral lectures given to students and colleagues. Although we do not possess a suitable general theory for the formulation of information conservation laws valid for the

research process, we feel intuitively that the amount of symbolic information produced cannot exceed the amount of structural information that is embodied by the research target. Formal approaches to such a putative conservation of information are often discussed as "no-free-lunch theorems" [60]. If there were a perpetuum mobile of the fourth kind that does nothing but generate useful symbolic information without input of structural information, a great deal of money could be saved which is currently spent on satellites, particle accelerators, or geological drilling. The invention of such a machine of universal knowledge is very unlikely even though its existence is not forbidden by any natural law we know of. Similar to the overwhelming empirical evidence for the validity of the Second Law, some people believe in the physical impossibility of a perpetuum mobile of the fourth kind based the experience that systematic prophecy and fortune telling often failed in the past. Winners of lotteries are not counted here as prophets. Humans tend to believe in divine beings, oracles, or superstition equipped with the gift of prophecy, but this wishful thinking is not supported by scientific observational evidence. Contemporary science was not able to observe information transfer against the arrow of time. However, science is always in the process of development and growth. As a result, it has been demonstrated that utilization of oracles can be very useful for people [59,61,62].

The assumed conservation of information during the transfer of structural information to symbolic information is also valid for the reverse process. Symbolic information has the purpose of being transformed back to structural information, in the form of actions taken based on the information, or of structures generated by symbolic instructions. The phenotype of an organism is an incarnation of its symbolic genetic information. Similarly, a technical product is an incarnation of symbolic recipes or construction plans. The final product is a structural representation of the former symbolic information—new phenotypic traits do not appear unless the genetic code includes the instructions for their ontogenetic development.

Conservation of information during the conversion processes between structural and symbolic information, as described before, is rather different from conservation of information during transmission processes from structure to structure, and from one symbolic information processor to another.

Structural information is transferred according to physical interaction laws. A quantum state, say, described by a wave function, transfers its information to the subsequent quantum state, as described by the time evolution of the Schrödinger equation. Physical interaction is materialized by the transfer of matter, energy, and entropy, and is governed by their respective conservation laws. The loss of energy of one of the interacting systems is accompanied by the equivalent gain of energy by another system, and so, information gained by such an energy transfer is accompanied by an equivalent energy loss at the other side.

Transfer of symbolic information, however, does not require that the gain of information by a receiver is necessarily at the cost of information loss at the information source. Newspapers, TV, and the Internet demonstrate that symbolic information, once available, may be "shared" by numerous receivers without degrading the original information. Sharing or broadcasting information is possible in ways that conserve the information. However, this process may increase or decrease the value of that information [8]. In this paper, we do not study changes of value in the processes of information transmission.

There are different ways to define the meaning of conservation of information. Two such options are measure-related conservation and transformation-related conservation.

(i) *Measure-related conservation* of information is realized if, in a process of conversion of a portion of information $I$ to a different form $I'$, a certain measure of information, e.g., $S$, remains the same, i.e., $S(I) = S(I')$. For instance, if $I$ is the structural information of a macroscopic system at some time $t$, and $I'$ is the structural information of that system at some later time $t'$, and entropy $S$ is the chosen measure of information, then information is conserved with respect to $S$ in reversible processes. Similarly, it is possible to understand conservation of mass, spin, and electric and magnetic charge as the conservation of structural information [63].

(ii) *Transformation-related conservation* of information is realized if in a process of conversion of the form $F$ of a portion of information $I$ to a different form $F'$, while both $F$ and $F'$ have the capacity to cause the same changes in a system $R$. According to the general theory of information both $F$ and $F'$ represent the same information. For instance, if $I$ is the symbolic information in a textbook $F$ written in a language $L$ and $F'$ is a contextual translation of the textbook $F$ to another language, then for a student $R$ the information is transformation-relatedly conserved if the translation is correct and $R$ knows both languages to the same extent. The translation may be formally written as the operation $T$: $F \rightarrow F'$. It is possible to interpret conservation as existence of the inverse operator $T^{-1}$, a backward translation, such that $F = T^{-1}(F')$ and all translations $T$ form a mathematical group. Similarly, structural information in quantum mechanics is subject to conservation in processes governed by a group of unitary evolution operators [28].

Independently of the particular way by which conservation of information may be defined in detail, we may state that, quite generally, conservation of information is understood as a binary relation between two portions of information, $I$ and $I'$. More precisely, mathematically, this relation is a reflexive *equivalence relation* because it has the properties of *symmetry*, *transitivity*, and *reflexivity*. All portions of information, which pairwise are mutually information-conservative, form an *equivalence class*. If each element of this class can be generated from another, arbitrarily-chosen element by a certain transformation $T$, then this set of transformations forms a mathematical group.

In the context of this article, considering conservation of information is fundamental as it permits discrimination of processes that violate conservation, namely, processes that destroy or produce information. Several information theories, such as the one of Shannon, aim at suitable formal descriptions of non-conservative processes. Structural and symbolic information differ with respect to processes that generate information; typically, structural information is produced in processes of physical self-organization, while symbolic information is generated during evolutionary processes. Here, evolution is understood as a potentially unlimited succession of self-organization steps [7].

## 7. Conclusions

Approaches to structural and symbolic information studied here are related to the basic ontological problem of the role of an observer. Namely, the question is whether properties of physical things and relations between them exist by themselves or their existence is dependent on the presence of an observer. In Section 3 of this work, structural information is treated from the perspective of an observer although such an observer can be not only a human being but also any cognitive system. At the same time (in Section 4), structural information is regarded as an intrinsic attribute of physical things independent of any observer. Symbolic information is explored in Section 5 as information created and utilized exceptionally by an observer while interactions between structural and symbolic information are explicated in Section 6.

**Author Contributions:** Mark Burgin and Rainer Feistel wrote the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
2. Ladyman, J. Structural Realism, Stanford Encyclopedia of Philosophy. 2009. Available online: http://plato.stanford.edu/entries/structural-realism/ (accessed on 1 November 2017).
3. Worrall, J. Structural Realism: The Best of Both Worlds? *Dialectica* **1989**, *43*, 99–124. [CrossRef]
4. Ebeling, W.; Feistel, R. Selforganization of Symbols and Information. In *Chaos, Information Processing and Paradoxical Games: The Legacy of John S Nicolis*; Nicolis, G., Basios, V., Eds.; World Scientific Pub Co.: Singapore, 2015; pp. 141–184.
5. Feistel, R. Self-organisation of Symbolic Information. *Eur. Phys. J. Spec. Top.* **2017**, *226*, 207–228. [CrossRef]

6. Feistel, R. Emergence of Symbolic Information by the Ritualisation Transition. In *Information Studies and the Quest for Transdisciplinarity*; Burgin, M., Hofkirchner, W., Eds.; World Scientific: Singapore, 2017; pp. 115–164.
7. Feistel, R.; Ebeling, W. *Physics of Self-Organization and Evolution*; Wiley-VCH: Weinheim, Germany, 2011; p. 517.
8. Feistel, R.; Ebeling, W. Entropy and the Self-Organization of Information and Value. *Entropy* **2016**, *18*, 193. [CrossRef]
9. Ebeling, W.; Feistel, R. *Chaos und Kosmos: Prinzipien der Evolution*; Spektrum: Heidelberg, Germany, 1994. (In German)
10. Brillouin, L. *Science and Information Theory*; Academic Press: New York, NY, USA, 1956.
11. Klimontovich, Y.L. *Turbulent Motion, the Structure of Chaos*; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1991.
12. Schrödinger, E. *What Is Life—The Physical Aspect of the Living Cell*; Cambridge University Press: Cambridge, UK, 1944.
13. Lane, N. *The Vital Question: Why Is Life the Way It Is*? Profile Books: London, UK, 2015.
14. Pattee, H.H. The Physics of Symbols: Bridging the Epistemic Cut. *Biosystems* **2001**, *60*, 5–21. [CrossRef]
15. Pearson, K. *The Grammar of Science*, 2nd ed.; Adam and Charles Black: London, UK, 1900.
16. Eigen, M. *From Strange Simplicity to Complex Familiarity*; Oxford University Press: Oxford, UK, 2013.
17. Boyajian, T.S.; LaCourse, D.M.; Rappaport, S.A.; Fabrycky, D.; Fischer, D.A.; Gandolfi, D.; Kennedy, G.M.; Korhonen, H.; Liu, M.C.; Olah, A.M.K.; et al. Planet Hunters IX, KIC 8462852: Where's the flux? *Mon. Not. R. Astron. Soc.* **2016**, *457*, 3988–4004. [CrossRef]
18. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: New York, NY, USA, 2010.
19. Burgin, M. Information in the Structure of the World. *Inf. Theor. Appl.* **2011**, *18*, 16–32.
20. Burgin, M. Epistemic Information in Stratified M-Spaces. *Information* **2011**, *2*, 697–726. [CrossRef]
21. Burgin, M. Weighted E-Spaces and Epistemic Information Operators. *Information* **2014**, *5*, 357–388. [CrossRef]
22. Burgin, M. Nonlinear phenomena in spaces of algorithms. *Int. J. Comput. Math.* **2003**, *80*, 1449–1476. [CrossRef]
23. Burgin, M. Data, Information, and Knowledge. *Information* **2004**, *7*, 47–57.
24. Capurro, R.; Fleissner, P.; Hofkirchner, W. Is a Unified Theory of Information Feasible? In *The Quest for a Unified Theory of Information: Proceedings of the Second International Conference on the Foundations of Information Science*; Hofkirchner, W., Ed.; Gordon and Breach Publ.: Amsterdam, The Netherlands, 1999; pp. 9–30.
25. Melik-Gaikazyan, I.V. *Information Processes and Reality*; Nauka: Moscow, Russia, 1997.
26. Bates, M.J. Information and knowledge: An evolutionary framework for information science. *Inf. Res.* **2005**, *10*, 10–14.
27. Reading, A. The Biological Nature of Meaningful Information. *Biol. Theory* **2006**, *1*, 243–249. [CrossRef]
28. Giddings, S.B. Black holes, quantum information, and the foundations of physics. *Phys. Today* **2013**, *66*, 30. [CrossRef]
29. Hawking, S. *The Universe in a Nutshell*; Bantam Books: New York, NY, USA, 2001.
30. Zhang, B.; Cai, Q.-Y.; Zhan, M.-S.; You, L. Information conservation is fundamental: Recovering the lost information in Hawking radiation. *Int. J. Mod. Phys. D* **2013**, *22*, 1341014. [CrossRef]
31. Planck, M. *Theorie der Wärmestrahlung*; Johann Ambrosius Barth: Leipzig, Germany, 1966. (In German)
32. Gilbert, W. Origin of life: The RNA world. *Nature* **1986**, *319*, 618. [CrossRef]
33. Fuchs-Kittowski, K. Information und Biologie. Informationsentstehung als neue Kategorie für eine Theorie der Biologie. In *Biochemie ein Katalysator der Biowissenschaften. Kolloquium der Leibniz-Sozietät am 20. Nov. 1997 zum 85. Geburtstag von Samuel Mitja Rapoport*; Sitzungsberichte der Leibniz-Sozietät, Band 22; trafo-Verlag: Berlin, Germany, 1999. (In German)
34. Monod, J. *Chance and Necessity*; Alfred A. Knopf: New York, NY, USA, 1971.
35. Borges, J.L. *The Library of Babel*; Penguin: New York, NY, USA, 1998.
36. Feistel, R. Ritualisation und die Selbstorganisation der Information. In *Selbstorganisation—Jahrbuch für Komplexität in der Natur-, Sozial- und Geisteswissenschaften*; Duncker & Humblot: Berlin, Germany, 1990; Volume 1, pp. 83–98. (In German)
37. Ebeling, W. Ist Evolution vom Einfachen zum Komplexen gerichtet? *Über Werte und Emergenz. Sitzungsberichte der Leibniz-Sozietät der Wissenschaften zu Berlin* **2016**, *125*, 69–80. (In German)

38. Burgin, M.; Meissner, G. 1 + 1 = 3: Synergy Arithmetic in Economics. *Appl. Math.* **2017**, *8*, 133–144. [CrossRef]
39. Lawrence, C. Making 1 + 1 = 3: Improving sedation through drug synergy. *Gastrointest. Endosc.* **2011**, *73*, 215–217.
40. Trott, D. *One Plus One Equals Three: A Masterclass in Creative Thinking*; Macmillan: London, UK, 2015.
41. Butterfield, J. Laws, causation and dynamics at different levels. *Interface Focus* **2012**, *2*, 101–114. [CrossRef] [PubMed]
42. Fuentes, M.A. Complexity and the Emergence of Physical Properties. *Entropy* **2014**, *16*, 4489–4496. [CrossRef]
43. Born, M. Physics in My Generation. In *Symbol and Reality*; Springer: New York, NY, USA, 1969; Chapter 13; p. 132.
44. Endler, J.A. Some general comments on the evolution and design of animal communication systems. *Philos. Trans. Biol. Sci.* **1993**, *340*, 215–255. [CrossRef] [PubMed]
45. Pinker, S. *How the Mind Works*; Penguin Books: London, UK, 1997.
46. Pearson, K. *The Grammar of Science*, 3rd ed.; Cosimo: New York, NY, USA, 2007.
47. Shannon, C.E. The Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423, 623–656. [CrossRef]
48. Turing, A.M. Computing machinery and intelligence. *Mind* **1950**, *59*, 433–460. [CrossRef]
49. Gallistel, C.R. *The Organization of Learning*; The MIT Press: Cambridge, MA, USA, 1993.
50. Burgin, M. Information Theory: A Multifaceted Model of Information. *Entropy* **2003**, *5*, 146–160. [CrossRef]
51. Burgin, M. *Superrecursive Algorithms*; Springer: New York, NY, USA, 2005.
52. Catford, J.A. *Linguistic Theory of Translation*; Oxford University Press: Oxford, UK, 1965.
53. Weisgerber, L. *Grundzüge der Inhaltsbezogenen Grammatik*; Pädagogischer Verlag Schwann: Düsseldorf, Germany, 1971. (In German)
54. Burgin, M.; Schumann, J. Three Levels of the Symbolosphere. *Semiotica* **2006**, *160*, 185–202. [CrossRef]
55. Logan, R.K.; Schumann, J.H. The Symbolosphere, Conceptualiztion, Language and Neo-Dualism. *Semiotica* **2005**, *155*, 201–214. [CrossRef]
56. Schumann, J.H. The evolution of language: What evolved? In Proceedings of the Society for Pidgin and Creole Linguistics Summer Conference, University of Hawaii, Honolulu, HI, USA, 14–17 August 2003.
57. Prum, R.O. *The Evolution of Beauty: How Darwin's Forgotten Theory of Mate Choice Shapes the Animal World—And Us*; Doubleday: New York, NY, USA, 2017.
58. Burgin, M. *Theory of Knowledge: Structures and Processes*; World Scientific: New York, NY, USA, 2016.
59. Burgin, M. Ideas of Plato in the context of contemporary science and mathematics. *Athens J. Humanit. Arts* **2017**, *4*, 161–182.
60. Wolpert, D.H.; Macready, W.G. Coevolutionary free lunches. *IEEE Trans. Evol. Comput.* **2005**, *9*, 721–735. [CrossRef]
61. Leeson, P.T. Oracles. *Ration. Soc.* **2014**, *26*, 141–169. [CrossRef]
62. Burgin, M. *Inaccessible Information and the Mathematical Theory of Oracles, in Information Studies and the Quest for Transdisciplinarity: Unity through Diversity*; World Scientific: New York, NY, USA, 2017; pp. 59–114.
63. Pappas, N.D. A New Approach to Information Loss (no) Problem for Black Holes. *Int. J. Theor. Math. Phys.* **2012**, *2*, 5–9. [CrossRef]


© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

*Proceedings*

# Structures and Structural Information †

**Mark Burgin [1,*] and Rainer Feistel [2]**

[1] Department of Computer Science, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA
[2] Leibniz Institute for Baltic Sea Research, Warnemünde D-18119, Germany; rainer.feistel@io-warnemuende.de
* Correspondence: mburgin@math.ucla.edu
† Presented at the IS4SI 2017 Summit DIGITALISATION FOR A SUSTAINABLE SOCIETY, Gothenburg, Sweden, 12–16 June 2017.

Published: 9 June 2017

**Abstract:** Structures play a crucial role in the world. They define things and thoughts, processes and functions, images and ideas. The goal of this research is to study structural information, which is intrinsically connected to structures. Here we present two approaches. In one of them, structural information is information in structures. In the other one, structural information is information about structures.

**Keywords:** information; structure; structural information; interaction; correctness; knowledge

## 1. Introduction

Everything has a structure and this structure makes things such as they are. This was declared by Aristotle for material things and demonstrated in the general theory of structures developed in [1,2] for the whole generality of existing and possible objects. This is the core reason of importance of structural information, which provides and/or changes knowledge about structures.

As Brinkley writes, "implicit in the word "structure" is not only the concept of elementary units or parts, but also the interdependence and relationships of those parts to form a whole and it can thus be argued that modern science has adopted a structural approach to understanding the natural world, in which parts are defined and the interactions among them are explored" [3,4].

Structural information is the core of structuralism, the heart of structural realism and the basic essence of structural informatics.

The goal of this work is to study structural information based on the general theory of information [1,5–8], research of Feistel and Ebeling [9–12] and works of other authors in this area. The goal is developing more comprehensive and advanced knowledge about structural information and its relation to symbolic information.

## 2. Structural Information in the Context of the General Theory of Information

It is possible to comprehend structural information in different ways. For instance, Bates [13] treats structural information as "the pattern of organization of matter and energy," while Reading [14] defines it as "the way the various particles, atoms, molecules, and objects in the universe are organized and arranged."

At first, we consider the approach developed in the general theory of information. The general theory of information discerns information in the broad sense (Ontological Principle O2) and information in the strict sense (Ontological Principle O2a).

*Structural information is a kind of information in the strict sense being defined as a capacity to change the subsystem of knowledge in an intelligent system.*

It is possible to differentiate three types of structural information.

1. *Inherent structural information* is information in structures.
2. *Descriptive structural information* is information about structures.
3. *Constructive structural information* is information that allows building knowledge about structures.

These definitions allow getting key properties of structural information. Let us consider some of them.

1. *Structural information can be more correct or less correct.*

Correctness of structural information about a system depends on correctness of knowledge produced by this information [7]. As we know, some knowledge can be more correct, better representing the essence of the system, while other knowledge is less correct, providing a worse representation of the fundamental nature of the system.

Here are two examples.

**Example 1.** *For a long time, people believed that the Earth was flat, i.e., had the structure of a plane. Then scientists found that the Earth had the structure of a ball. Then scientists assume that the Earth had the structure of a geoid.*

**Example 2.** *For a long time, people believed that in the structure of the Solar system, the Sun rotated around the Earth.Then scientists found that the Earth rotated around the Sun and the orbit had the structure of a circle.Then scientists assume that the Earth rotates around the Sun and the orbit had the structure of an ellipse.*

2. As a rule, *structural information about a system is not unique.*

Many researchers believe that each (e.g., a natural) system has a unique structure. At the same time, according to the general theory of structures [1], any system has several structures. For instance, the structure of a table on the level of its parts is essentially different from the structure of this table on the level of molecules as well as from the structure of this table on the level of its parts such as legs. In essence, material systems, which people can see with their eyes and touch with their hands, have structural information on different levels.

3. *Structural information about a system can be inherent to this system, inbuilt in the interaction with the system or innate for an image of the system.*

Indeed, as it is stated above, structure makes things such as they are. Naturally, structural information reflects this identity of things although structural information about different systems and objects can be similar.

4. *Processes in a system can change structural information about this system.*

Indeed, the evolution (development) of a system can produce an essentially new structure when the system is changed, even becoming another system. For instance, butterflies have the four-stage life cycle. In it, winged adults lay eggs, which later become caterpillars, which later pupate in a chrysalis, while at the end of the metamorphosis, the pupal skin splits and a butterfly flies off.

5. *Structural information about a system describes this system to a definite extent of precision, i.e., structural information can be more precise and less precise.*

For instance, the Copernican model (structure) of the Solar System is more precise than the Ptolemaic model (structure) of the Solar System. Another example comes from mathematics where mathematicians are striving to find the decimal structure of the number $\pi$ with higher and higher precision.

6. *For complex systems, it is possible to consider structural information on different levels and various scales.*

For instance, it is possible to treat the structure of a human being on the level of visible parts, on the level of its functional systems, on the level of inner organs, on the level of cells, on the level of chemical compounds or on the level of molecules.

7. *Structural information about a subsystem of a system is not always a part of the structural information about this system.*

For instance, when we consider an organism as a system of its visible parts, the structure of its nervous system is not a part of this structure.

8. *The process of conversion of structural information about a system into knowledge about this system is, in essence, structuration of this system.*

Indeed, people get information about different objects in the form of raw data. Only after reception of this information, the brain converts these data into knowledge and this knowledge is often about the structure of studied objects. It is natural to treat data as symbolic representation of structural information, where correct structural information may be represented incorrectly by symbolic information. This may happen in the measuring device, or in the transmission or storage, or in the brain, etc.

Note that the general theory of information provides other possibilities for defining structural information. For instance, it can be information that changes the system of beliefs of an intelligent system.

## 3. Structural Information as the Intrinsic Property

At the same time, Feistel and Ebeling suggest the vision of structural information, in which structural information may no longer be restricted to changing just "knowledge in an intelligent system", and may more generally be defined as the capacity of a physical system, the "carrier of structural information", to cause changes in a second physical system, the "receiver of structural information" [9–12].

If in particular, the receiver is the same system as the carrier but at some later point of time, reversible microscopic dynamics described by the Liouville equation is universally understood as "conserving [microscopic] [structural] information" [15–17]. In contrast to this, irreversible macroscopic dynamics is commonly associated with a loss of [macroscopic] [structural] information, directly related to the growth of thermodynamic entropy [9,10]. In the sense of Planck [18] who wrote that "a macroscopic state always comprises a large number of microscopic states that combine to an average value", macroscopic structural information represents a portion of the microscopic structural information of a given system. This approach is consistent with the approach from Section 2. Namely, a system may have different structures, such as a microscopic structure and a macroscopic structure. In each of those structures, there exists related structural information. While the microscopic information may be conserved, the macroscopic information may decay according to the 2nd law of thermodynamics. That is, the different structures of the same system may possess different symmetries and may follow different laws. Irreducibilities of this kind indicate emergent properties or structures [10].

In his famous article "The RNA world" about the origin of life, Gilbert [19] wrote about "the useful distinction between information and function", that "information storage needs to be one-dimensional, for ease of copying, but molecules with enzymatic functions tend to be tight three-dimensional structures". This quotation demonstrates that the term "information" is in common scientific use in situations in which "knowledge in an intelligent system" does not exist at all, such as before life emerged on Earth. Gilbert refers to the structural information given by the sequence of building blocks in a chain molecule. This information is received later by the same molecule when it folds into a functioning enzyme, and by other reactants then catalyzed by it. Similarly, the information scientist Klaus Fuchs-Kittowski [20] speaks about "information" that was

generated in evolution for the first time when life began, and has allowed living structures to survive to the present day. Without retaining this structural information, living beings would be as astronomically rare among random chemical mixtures [21] as are sensible books among the random texts in Jorge Luis Borges' "Library of Babel" [22].

Structural information available from a carrier depends on the receiver determining what portion of this information is actually received. If, for example, the receiver is a thermometer and the carrier is liquid, then all information received is the temperature of the liquid. Structural information can be extracted from a given system by "measurement" when e.g., a sensor is used as a receiver. Structural information can be quantified if it is comparable to the structural information of a reference system, such as the length scale of a mercury thermometer.

A numerical value being the result of a comparison between the same kinds of structural information available from two different systems, such as by counting their parts, is a "measurement result". Numbers represent information in the symbolic form, or as "symbolic information". The meaning of symbolic information is subject to convention (such as what "reference" system is used) and is no longer a portion of the structural information of the carrier, such as printed symbols on a sheet of paper. Very different structural information carriers can carry the same symbolic information. Symbolic information is restricted to the realm of life [9–11], such as in the form of genetic DNA molecules or human knowledge, and emerged from structural information in the course of evolution by a transition process regarded as ritualisation.

The origin of life is a famous example for the ritualisation transition from structural to symbolic information. Organic chain molecules, see the quotation of Gilbert [19] above, carried structural information in the form of certain sequences of their elements. Some of those resulted in catalytic properties that supported the copying of the chain molecule, such as in a RNA-replicase cycle [19,23,24]. In the course of evolution, the initial chain molecules developed to carriers of symbolic information in a complex multi-step process [9,23]. Similar features of the ritaulisation transition are observed, for example, when humans began to speak, or when digital computers replaced mechanical calculators.

The codons of the modern genetic code are the symbols of genetic information, and the phenotypic traits of the developed organism represent the meaning of that information. The symbols also carry structural information, but due to the arbitrariness of symbols in a mature coding system, this structural information is irrelevant for the meaning they express. The properties of the ink used by Mozart for writing down his music are irrelevant for the beauty of its acoustic performance [25].

The main distinction between structural and symbolic information is that the meaning of structural information is inseparably bound to its physical structure, while symbolic information is free insofar as arbitrary symbols may take over the same role in the information processing system. Symbolic information possesses the code symmetry, or coding invariance, that emerges during the ritualisation transition [26,27]. Related to this symmetry is the structural information of symbols that preserves traces of the evolution of the symbolic information-processing system. So, it is known from many spoken words that they have onomatopoetic roots. The similarity of words used in the various languages provides structural information about the migration histories of the populations. In the same way, the physical and chemical properties of the genetic apparatus carry structural information about the beginning of life in an ancient past.

## 4. Conclusions

Presented here approaches to structural information reflect the basic ontological problem of the role of an observer. Namely, the question is whether properties of physical things exist by themselves or their existence is dependent on the presence of an observer. In Section 2 of this work, structural information is treated from the point of view of an observer although such an observer can be not only a human being but any cognitive system. In Section 3 of this work, structural information is regarded as an intrinsic property of physical things independent of any observer.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References


1. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
2. Burgin, M. *Theory of Knowledge*: *Structures and Processes*; World Scientific: New York, NY, USA; London, UK; Singapore, 2016.
3. Brinkley, J.F. Structural informatics and its applications in medicine and biology. *Acad. Med.* **1991**, *66*, 589–591.
4. Brinkley, J.F. *What Is Structural Informatics*? 1999. Available online: http://sig.biostr.washington.edu/overview/si.html (accessed on 4 July 2017)
5. Burgin, M. Information: Problems, Paradoxes, and Solutions. *Triple C* **2003**, *1*, 53–70.
6. Burgin, M. Data, Information, and Knowledge. *Information* **2004**, *7*, 47–57.
7. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010
8. Burgin, M. Information Dynamics in a Categorical Setting. In *Information and Computation*; World Scientific: New York, NY, USA; London, UK; Singapore, 2011; pp. 35–78
9. Feistel, R.; Ebeling, W. *Physics of Self-Organization and Evolution*; Wiley-VCH: Weinheim, Germany, 2011.
10. Feistel, R.; Ebeling, W. Entropy and the Self-Organization of Information and Value. *Entropy* **2016**, *18*, 193.
11. Ebeling, W.; Feistel, R. Selforganization of Symbols and Information. In *Chaos, Information Processing and Paradoxical Games*: *The Legacy of John S. Nicolis*; Nicolis, G., Basios, V., Eds.; World Scientific Pub Co.: Singapore, 2015; pp. 141–184
12. Feistel, R. Self-organisation of Symbolic Information. *Eur. Phys. J. Spec. Top.* **2016**, doi:10.1140/epjst/e2016-60170-9.
13. Bates, M.J. Information and knowledge: An evolutionary framework for information science. *Inform. Res*. **2005**, *10*, 239.
14. Reading, A. The Biological Nature of Meaningful Information. *Biol. Theory* **2006**, *1*, 243–249.
15. Hawking, S. *The Universe in a Nutshell*; Bantam Books: New York, NY, USA, 2001
16. Zhang, B.; Cai, Q.-Y.; Zhan, M.-S.; You, L. Information conservation is fundamental: Recovering the lost information in Hawking radiation. *Int. J. Mod. Phys. D* **2013**, *22*, 1341014.
17. Ginnings, S.B. Black holes, quantum information, and the foundations of physics. *Phys. Today* **2013**, *66*, 30–35.
18. Planck, M. *Theorie der Wärmestrahlung, 6. Auflage*; Johann Ambrosius Barth: Leipzig, Germany, 1966
19. Gilbert, W. The RNA world. *Nature* **1986**, *319*, 618.
20. Fuchs-Kittowski, K. Information und Biologie—Informationsentstehung als neue Kategorie für eine Theorie der Biologie. In *Biochemie—ein Katalysator der Biowissenschaften*; Kolloquium der Leibniz-Sozietät am 20. Nov. 1997 zum 85. Geburtstag von Samuel Mitja Rapoport, Sitzungsberichte der Leibniz-Sozietät, Band 22; trafo-Verlag: Berlin,Germany, 1999.
21. Monod, J. *Chance and Necessity*; Alfred A. Knopf: New York, NY, USA, 1971.
22. Borges, J.L. *The Library of Babel*; Penguin: New York, NY, USA, 1998.
23. Ebeling, W.; Feistel, R. On the Evolution of Biological Macromolecules. I. Physico-Chemical Self-Organization. *Stud. Biophys.* **1979**, *75*, 131–146.
24. Eigen, M. *From Strange Simplicity to Complex Familiarity*; Oxford University Press: Oxford, UK, 2013.
25. Eigen, M. *Stufen zum Leben*; Piper: München,Germany, 1992.
26. Feistel, R. Ritualisation und die Selbstorganisation der Information. In *Selbstorganisation und Determination*; Niedersen, U., Pohlmann, L., Eds.; Duncker & Humblot: Berlin, Germany, 1990; pp. 83–98.
27. Feistel, R. Emergence of Symbolic Information by the Ritualisation Transition. In *Information Studies and the Quest for Transdisciplinarity*; Burgin, M., Hofkirchner, W., Eds.; World Scientific: Singapore, 2017; pp. 115–164.


©2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open accessarticle distributed under the terms and conditions of the Creative Commons Attribution(CC BY) license (http://creativecommons.org/licenses/by/4.0/).

information

MDPI

Article

# A Dialogue Concerning the Essence and Role of Information in the World System

**Mark Burgin [1] and Jaime F. Cárdenas-García [2,*]**

[1] Department of Mathematics, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

[2] Department of Mechanical Engineering, University of Maryland–Baltimore County, 1000 Hilltop Circle, Baltimore, MD 21250, USA

* Correspondence: jfcg@umbc.edu

Received: 6 July 2020; Accepted: 15 August 2020; Published: 21 August 2020

**Abstract:** The goal of this paper is to represent two approaches to the phenomenon of information, explicating its nature and essence. In this context, Mark Burgin demonstrates how the general theory of information (GTI) describes and elucidates the phenomenon of information by explaining the axiomatic foundations for information studies and presenting the comprising mathematical theory of information. The perspective promoted by Jaime F. Cárdenas-García is based on Gregory Bateson's description of information as "difference which makes a difference" and involves the process of info-autopoiesis as a sensory commensurable, self-referential feedback process.

**Keywords:** information; info-autopoiesis; principle; axiom; transformation; representation; carrier; feedback process; measure; ontology; axiology

## 1. Introduction

Now, dialogue is defined as a written or spoken conversational exchange between two or more people, as well as the literary and theatrical forms of such an exchange. In the West, this form is predominantly associated with Platonian dialogues, which persuasively demonstrated the multiple advantages of this form of exposition of diverse philosophical ideas.

However, in the Middle East and Asia, dialogue, as a literary genre, dates back to ancient times. For instance, Sumerian disputations are preserved in copies from the late third millennium B.C.E., while Rigvedic dialogue hymns are dated to the second millennium B.C.E.

In the West, Plato (437–347 B.C.E.) perfected dialogue as a literary form, and all his extant writings, except the Apology and Epistles, use this form. Following Plato, the dialogue became a popular literary genre in antiquity, and several important works both in Latin and in Greek were written in this form.

For instance, Protagoras of Abdera also used dialogues to present his ideas and opinions [1]. Plutarch of Chaeronea in Boeotia, who was a Platonist philosopher, wrote his collection of "Moralia" or "Ethical Essays" mostly in dialogue format [2]. Xenophon wrote his dialogue Symposium, and even Aristotle is said to have written several philosophical dialogues.

Dialogue as a literary form once more became very popular in the medieval period [3]. The most prominent dialogue authors at time were Anicius Manlius Severinus Boethius, John Scottus Eriugena, Augustine, Peter Abelard, Gilbert Crispin, Petrus Alphonsi, Raymund Lull, Yehudah HaLevi, and Nicolas of Cusa.

Later famous philosophers, such as David Hume, George Berkeley, Torquato Tasso, and Judah Abrabanel, wrote some of their works in the form of a dialogue.

From literature and philosophy, dialogue came to science. One of the most famous dialogues at the beginning of science was Galileo's *Dialogue Concerning the Two Chief World Systems* (*Dialogo*

*sopra i due massimisistemi del mondo*). In it, Galileo Galilei (1564–1642), who is considered the father of modern science by Albert Einstein and many other scientists, contrasts two different world views: the traditional at that time, the Ptolemaic geocentric system, according to which everything in the Universe circles around the Earth, and the new emerging Copernican system, according to which the Earth is one of the planets, which together with other planets orbits the Sun.

It is possible to give examples of many other prominent authors of dialogues demonstrating the attractiveness of the dialogue as a tool for the exposition of ideas, opinions, beliefs, values, and attitudes. We only mention such renowned figures of human culture as Iris Murdoch and George Santayana.

In our times, dialogue is also used as a functional form for the exposition of general, as well as scientific, ideas, opinions, and theories. For instance, the mathematician and computer scientist Donald Knuth used dialogue for presenting an innovative mathematical theory of surreal numbers [4]. Roth, McRobbie, and Lucas published four dialogues and what they call metalogues about the nature of science [5]. Information scientists Dodig-Crnkovic and Müller wrote a dialogue concerning two possible world systems, info-computational versus mechanistic contrasting the mechanistic approach to computation where computation is restricted to the functioning of such a popular model as Turing machine and info-computationalism, which treats the dynamics of the universe as a process of computation [6]. One more example of a contemporary scientific dialogue is an interview-style conversation of a group of neuroscientists who explore the conversation in science and the role of storytelling in engaging with and educating the public about what they do as scientists [7].

Moreover, actually, any interview is a dialogue, while an interview with a scientist is often a popular exposition of scientific ideas, achievements, and results in the form of a dialogue. Many different researchers have demonstrated the usefulness of the dialogical literary form in this and other areas (cf., for example, [8–10]).

Understanding the power of dialogue for comprehension to a wide audience and exposition of scientific ideas and results, the authors of this paper chose this form to better outline the differences and similarities in their approaches to the focal concept of information. The dialogue presented here is the result of the active exchange of opinions between the authors after they met at the International Society for the Study of Information (IS4SI) 2019 Summit organized at the University of California, Berkeley in June 2019. In spite of having a common interest in information, the authors found that they approached this widely discussed and fundamental topic from paradigmatically different perspectives. Mark Burgin represented the general theory of information as a comprehensive unified information theory providing foundations for all kinds of information studies. At the same time, Jaime F. Cárdenas-García described the process of info-autopoiesis, developing the difference theory of information based on Gregory Bateson's aphoristic expression 'difference which makes a difference'. As a result, they decided to explain their views in the form of a dialogue in treating the phenomenon of information. It is necessary to remark that the exposition of the general theory of information is concise because there are many publications [11–19] on this theory, including its exposition in a book [11]. Thus, there are two participants in this dialogue: J and M. The authors hope that for the readers, this exchange of opinions will be meaningful and revelatory of how information may be perceived from the presented paradigmatically different perspectives.

## 2. The Dialogue

**M**: Let's discuss the problems of the essence of information and how it is possible to define it.

**J**: My interest in the topic of information results from becoming acquainted with the work of Trewavas and Baluska [20]. In turn, it introduced me to the works of Maturana and Varela [21,22]. Eventually, I identified the problem of understanding information as an essential issue for answering fundamental questions that have eluded explanation. One of my explorations led me to publications of Gregory Bateson, where he introduces a definition of information as "a difference which makes a difference" [23,24]. This is certainly very different from the belief that information is a fundamental quantity of the universe, implying that we humans are able to identify it as existing everywhere and

use it to our advantage [25–28]. This point of view appeared with the creation of cybernetics and the construction of computers during World War II, demonstrating that machines can have what before were considered as exclusively human capabilities.

**M**: It looks like you are suggesting taking Bateson's statement that *information is the difference which makes a difference* [23] as an ultimate definition of information.

**J**: Indeed, Bateson's simplicity of definition is certainly attractive as a starting point, and many researchers always seemed to accept this definition, but no one seemed to go beyond this phrase. As I developed my interest in Bateson, I more and more appreciated his musings while making them my own. One interesting aspect of Bateson's statement is its subjective aspect, which is atypical in the world where "true" scientists emphasize the objective aspect of everything. This apparent contradiction is certainly something that needs to be taken seriously as the following questions come into view. How does one become an objective scientist if he or she is a subjective being? Does not Bateson's statement hold the promise of simultaneous subjectivity and objectivity?

**M**: Okay, taking Bateson's statement as a candidate for a definition of information, its thorough analysis shows that it is as deceptive as it is beautiful. First, without a sufficiently exact definition of difference, this is not a definition, as it is wrong to define one unknown term through another unknown term. This is a logical fallacy. Second, even if we take the mundane meaning of the word *difference*, we may come to the conclusion that anything is a difference because anything is different from something else. Besides, being different, anything makes this difference and thus, we have to logically conclude that anything is information. This makes the concept of information void from a scientific point of view. Fortunately, this is not true, and the concept of information is not only meaningful but is becoming one of the pillars of contemporary science.

**J**: I fully understand your concern about logical consistency and general confusion when the word "difference" is used. That is why I said that *information as a difference which makes a difference* is only a starting point. To get a measure of what difference means, we can appeal to its dictionary definition in the Merriam-Webster Online Collegiate Dictionary [29]: (1) the quality or state of being dissimilar or different; and (2) an instance of being unlike or distinct in nature, form, or quality. So, indeed, we are in agreement, though at first glance we seem to be in a logical quandary and great confusion. However, let me remind you that context is very important in any situation. We have to ask, who determines these differences? We are certainly talking about human beings, such as Bateson or ourselves, who are determining these differences. Sometimes, they can be very complex differences such as the ones we might experience in looking at and analyzing an abstract painting that compounds many different levels and guises of differences that are possible to distinguish. Luckily for us, that is not where we begin our process of distinguishing differences. We begin this process from the point of our conception as living beings. A single cell has to become two, then four, and so on until the emergence of the child from the womb to begin an additional gestation period out of the womb. How does the single cell know how to become two cells? What is/are the difference(s) that it detects that allows this certain process to become effective? I am not a biologist, so I do not want to delve into all of the biological complexity that probably plays a role. However, I do want to assert that at some point in this process, our five primary senses (touch, sight, hearing, smell, and taste) come online 24/7. This is the only basis for our access to the world. One thing that can be said with certainty about our senses is that they are continuously detecting spatial and/or temporal differences in our dynamic environment. We do it because the alternative, to not do it, would only lead to our certain demise. We have to be good at detecting differences so that when made to feel a nipple we can suck as though our life depended on it, but unaware that that is why we are doing it.

One important aspect of our primary senses is that they deal with quantities/qualities that are commensurable, i.e., that have a common measure. For example, the sense of touch (though multidimensional as composed of mechanoreceptors, thermoreceptors, nociceptors, proprioceptors) might be simply ascribed as being sensitive to pressure, and in that role, it is able to keep track of all such pressure sensations that come into its sphere of action. As might be imagined, from one

instant of time to the next, pressure sensations are felt by the human in question and become part of her experience. This is how quantitatively and unambiguously "a (pressure) difference" becomes qualitatively "a (pressure) difference which makes a difference". This is the process of information that Bateson discovered and applies to any and all of our primary senses, which not only act individually but in concert. Primary senses provide our only contact with our environment and provide keys to our development. This is why we can say that:

"A key issue in reaching a unified definition of information is the fundamental problem of identifying how a human organism, in a self-referential process, develops from a state in which its knowledge of the human-organism-in-its-environment is almost non-existent to a state in which the human organism not only recognizes the existence of the environment but also sees itself as part of the human-organism-in-its-environment system. This allows a human organism not only to self-referentially engage with the environment and navigate through it, but also to transform it into its own image and likeness. In other words, the Fundamental Problem of the Science of Information concerns the phylogenetic development process, as well as the ontogenetic development process of Homo sapiens from a single cell to our current multicellular selves, all in a changing long-term and short-term environment, respectively."

Summarizing the above quote, we can ask the question: What is the process of how we become what we become? Note that implicit in this conception of information is that all information that human beings have access to is produced by us. In other words, we can conceive it as a process of info-autopoiesis, or of self-produced information, as central to humankind [30,31]. This can be extended, more generally, to all cognitive beings that exist in nature.

**M**: Thus, you are implying that information is produced only by people.

**J**: Yes, the phylogenetic and ontogenetic development of humankind is a long process of information creation, which necessarily impacts our environment and us. However, we can even extend this concept and talk about an infosphere created by all living beings, but I am getting ahead of myself.

**M**: Now you are contradicting yourself, because at first, you once more assert that information is produced only by people, but then you add that all living beings create an infosphere.

**J**: As mentioned previously, we can conceive info-autopoiesis or self-produced information as central to humankind, but this ability may be extended more generally to all cognitive beings that exist in nature. This goes to the heart of whether or not humankind is unique as to its abilities when compared to other living beings. Asserting our uniqueness does not require denying the uniqueness of other living beings, or even constraining their abilities due to our inability to access their world.

**M**: I can agree that people created an infosphere and that all living beings created even a larger infosphere. Nevertheless, your statement causes many other questions.

First, do only living beings produce information, or can information be produced by other systems?

Second, do all living beings create information or only some of them?

Third, is all information created by living beings or only a part of it?

Four, do people only create information, or do they also receive information? Then, what does it mean to receive a difference?

Fifth, do people create or produce information from nothing or from something else?

Six, do only living beings contain/store information, or can other natural and/or artificial systems also contain/store information? In addition, what does it mean to store difference?

Seven, if living beings only create information, then what does information transmission mean? Then, what does it mean to transmit difference?

**J**: I am glad that we are in agreement as to that people created an infosphere and that all living beings created even a larger infosphere. Of course, this cannot but raise many additional questions such as the seven questions you have framed. Let me answer each question, in turn, as thoroughly as possible to allow us to come to a better understanding of our positions.

So, let us consider the first question, asking whether only living beings produce information or information can be produced by other systems.

I want to propose and postulate that all living beings engage in a process of info-autopoiesis, i.e., in a process of self-production of information. This extends the concept of autopoiesis by Maturana and Varela [21,22], who did not consider information as an element of autopoiesis. Indeed, the process of info-autopoiesis is the basis to understanding information in biology. This proposal could be considered as an attempt to do what Varela [32] suggested when he identified the need for the inclusion of information in autopoiesis. In the case of tool-making and machine-making living beings, this implies that the systems that are created by people and other living beings can also have the capacity to produce information.

To better organize my explanation of this issue, I would like to divide the history of information utilization and exploration into two major epochs. In the **first information epoch**, the need to provide a quantitative/objective appraisal of information might have existed but was not a priority. There was an implicit understanding (qualitative/subjective) of information as something that added to our existing knowledge, i.e., "when 'what we know' has changed" [33], with mostly semantic undertones. The major producers and consumers of information in this time period were human beings. Note that our understanding of information as "when 'what we know' has changed" has the connotation of difference, i.e., something has changed. However, we can also view this from another perspective that few would ascribe to information, but it is just as real: the role of human actions and effort (labor) exerted on our environment, which is more directly related to Bateson's "difference which makes a difference" producing changes in our environment. This perspective also falls in line with a historically meaningful perspective on the word information. We find that it derives from the Latin stem *informatio*, which comes from the verb *informare* (to inform) in the sense of the act of giving a form to something material as well as the act of communicating knowledge to another person [34–39]. The first of these meanings is what allows us to see a connection of information to human labor exertion. In other words, the term information may be said to mediate the act of labor between humans and nature, i.e., the act of labor as a metabolic connection between humans and nature is the action of giving form to something material, i.e., labor *in-forms* matter. In addition, matter *in-forms* humans by reacting to the efforts of humans. It is a never-ending interactive process of action–sensing–action. This notion of information correlates with that of Bateson emphasizing the utilization of our five primary senses in our environmental interactions.

Now, let us consider a description by Gregory Bateson, of a laborer wielding an ax:

"Consider a tree and a man and an ax. We observe that the ax flies through the air and makes certain sorts of gashes in a pre-existing cut in the side of the tree. If now we want to explain this set of phenomena, we shall be concerned with differences in the cut face of the tree, differences in the retina of the man, differences in his central nervous system, differences in his efferent neural messages, differences in the behavior of his muscles, differences in how the ax flies, to the differences which the ax then makes on the face of the tree. Our explanation (for certain purposes) will go round and round that circuit. In principle, if you want to explain or understand anything in human behavior, you are always dealing with total circuits, completed circuits. This is the elementary cybernetic thought [23]."

This is a description that evolves from a cybernetic perspective of the world by Gregory Bateson that identifies information/differences that are pertinent, in this case, to the dynamic and evolving labor effort at hand, which is no different from many typical labor tasks, and can be ascribed as a series of material informational efforts involving the use of the human brain, nerves, muscles, and sense organs. In short, we want to argue that labor and information/differences are intimately entwined and that every artifact embodies labor and information/differences. This aspect of all human artifacts goes largely unnoticed, but it lends much credence to the fact that we can easily recognize implements manufactured by humans no matter their anthropological age. Another example is signs of butchery in animal bones that are more than 2 million years old [40,41].

The **second information epoch** begins with Shannon's landmark paper on the Mathematical Theory of Information that was central to the establishment of 'Information Theory' as a discipline [42]. This prompted the need to quantify measurements of different kinds that would allow one "to discover

the method which will give us the maximum amount of information for a given outlay of time or space or other resources" [33] with mostly syntactic constructions that negate semantic content. In this current epoch, producers and consumers of information are not only humans but also machines that are designed and built for that purpose by humans. The production and consumption of information by humans consists in being the producers and users of Information and Communication Technologies (ICTs) such as wireless radios, cybernetic control mechanisms, encryption machines, and television, evolving to the technological levels that make items of common usage today such as cell phones, digital televisions, satellite communications, the internet, social media, etc. These ICTs allow messages to be composed by humans/machines, coded, and optimally transmitted as communication signals, which are received, denoised, decoded, and interpreted by humans/machines. This is the information basis of the information age.

**M**: I don't object to your stratification of the history of information utilization and exploration. At the same time, I would like to suggest that it is possible to distinguish more periods or epochs in this history.

The **first period** (**epoch**) began with the emergence of human beings and is characterized by the situation in which people operated with information but did not have the notion of information.

The **second period** (**epoch**) is characterized by the emergence and utilization of the notion of information.

The **third period** (**epoch**) is characterized by the beginning of information studies.

I would also like to remark that although etymology does always define a scientific concept, often it provides a path to a better understanding.

**J**: Your periodization is consistent with mine. So, let us return to our main problem and illustrate how Bateson's "difference which makes a difference" translates into a mechanism that is capable of generating information. For example, we can consider a thermostat represented in Figure 1. Note that the design, construction, and functioning of such a thermostat is the result of human labor, i.e., of an info-autopoietic process.

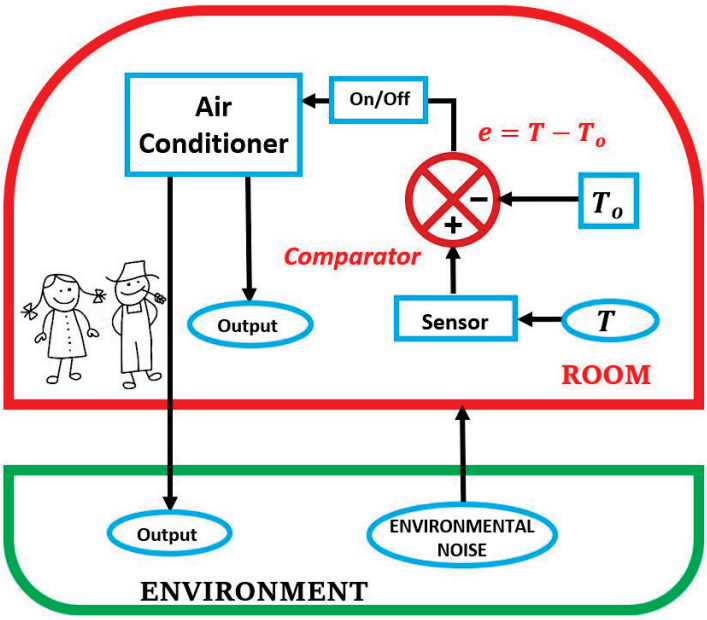



**Figure 1.** A one-parameter feedback loop simulation of a control mechanism (thermostat) associated with a room air conditioning system.

Figure 1 shows the relationship between a room and the surrounding environment. The room incorporates an air conditioner unit that can cool/heat the room and the associated thermostat control unit that controls its operation. The thermostat control unit consists of a one-parameter *Sensor* and

a *Comparator*. The one-parameter *Sensor* measures the room temperature $T$ on a continuous basis. This temperature is an input to the *Comparator* as the *detected parameter*. Another input to the *Comparator* is the temperature setting $T_o$ arbitrarily set by the occupant of the room based on her level of comfort. The temperature setting $T_o$ is regarded as the *reference parameter*. The role of the *Comparator* is to obtain the difference $e = T - T_o$ between the *detected parameter* and the *reference parameter*. This is an example of the Bateson *difference*, which may also be referred to as *error*. This *difference/error* is the parameter used to trigger the *On–Off* switch of the air conditioner unit. In cooling mode, the air conditioner turns *On* when $e > 0$, and turns *Off* when $e < 0$. The opposite is true when the air conditioner is in heating mode.

This configuration shows that the actions of the thermostat depend on the *Comparator* as a function of the *difference/error* obtained from comparing the *detected parameter* to the *reference parameter*. The circuit that comprises this system is a semi-closed-loop cybernetic feedback circuit, not a closed-loop cybernetic feedback circuit, since there is no direct connection between the air conditioner and the *Sensor*. In addition, there is no connection between the *Comparator* output and the temperature setting. This brings about the existence of two semi-closed-loop cybernetic feedback circuits. One feedback circuit comprises the air conditioner that exhibits two outputs: one to the room and the other to the outside environment. The impact of the air conditioner on the *Sensor* is by way of the air currents in the room as well as the environmental noise, which is the impact that the environment might exert on the walls of the room by being more/less windy or sunny/cloudy than the room. This feedback circuit has an eventual effect on the *detected parameter* of the temperature at the one-parameter temperature *Sensor*, leading to a subsequent action affecting the *Comparator* output that leads to an effect on the On/Off switch of the air conditioner. A second semi-closed-loop cybernetic feedback circuit mechanism to change the temperature setting $T_o$ is by way of the room occupant. In either of these two cases, the room occupants can intervene to make the room temperature amenable to their needs.

The definition of information by Bateson as *a difference which makes a difference* may be used in the context of this thermostat example. Let us note that this definition of information implies a quantitative portion (*a difference*) and a qualitative portion (*a difference which makes a difference*) [23]. The quantitative portion is the *difference/error* $e = T - T_o$ that is calculated by the Comparator that results in the On/Off actuation of the air conditioner; the qualitative portion is that given by the comfort level of the person inside the room that controls the setting $T_o$. Furthermore, the person whose temperature comfort is at issue in the use of the air conditioner may have the use of a thermometer to compare its experienced comfort level. This is what drives the temperature setting. Here, we identify, for illustrative reasons that the *difference/error*, $e$, represents Bateson *information*. This means that the terms *difference/error/information* here are treated, although not reducible to each other, as equivalent. We further note that in this example, the *error* is the term that acts as a cybernetic correction factor in pursuing the goal of achieving temperature $T_o$.

In short, the four-step conceptualization of how this semi-closed-loop cybernetic feedback system consisting of a one-parameter *Sensor* and *Comparator* works is as follows:

1. A *Comparator* (thermostat) is set to a *reference parameter* (room temperature setting);
2. The *sensor* (room temperature sensor) distinguishes the value of the *detected parameter*;
3. The *Comparator* obtains the difference/error between the detected parameter and the reference parameter;
4. The detected *difference/error* is the *information* needed to send a signal to turn-on/turn-off the system governing the level of temperature in the room.

In summary, all living beings engage in a process of info-autopoiesis, i.e., in a process of self-production of information, which includes the capacity of information production by all artifacts that incorporate the designed means for such production.

**M**: Your example of a thermostat is a good illustration of a possible interpretation of Bateson's statement. However, what system creates information $e$? Is it a human being or the thermostat?

Indeed, the human being only introduces $T_o$, while the thermostat constructs the *difference/error e* $= T - T_o$, and you interpret this difference/error as information.

However, this example shows that even such a simple system as a thermostat creates information. Besides, when you stated before that "matter *in-forms* humans by reacting to the efforts of humans," it implied that matter created information. So, how to balance this with your previous assertion that only living beings create information?

**J**: Human beings in the process of interacting with their environment conceive, design, and create devices such as the thermostat. Implicit in the conception, design, and creation of these devices is the inclusion of information in the devices by human beings. Furthermore, the implication is that if the devices serve a specific purpose for humans, the devices are designed so as to process specific information for humans and then to display that information so that humans can take advantage of it. So, your appraisal is correct: the thermostat does create information that a human can interpret and use, but at the behest of humans.

This is the same as when I stated that "matter *in-forms* humans by reacting to the efforts of humans." Consider any mundane activity that engages a human being with matter, such as the cutting down of a tree. With each cut of the tree, the wood is acted on and transformed. This transformation, due to human actions, allows the human the opportunity to assess her work and decide what to do next. The in-forming of matter by the human has a purpose, which the human continuously assesses. In each and every case, it is the human that initiates action on matter, i.e., in-forms matter, with a specific purpose. Of course, matter, as a reactive object, yields a response that the human uses to gain information about the behavior, mechanical or otherwise, of the matter at hand for present or future use. This is at the heart of info-autopoietic behavior by humans, i.e., humans self-produce information from their interactions with the environment. So, it is not matter that creates information; rather, it is matter, as a function of the actions and needs of humans, that reflects desired information by humans. The thermostat is also an example of this, since it processes and produces information at the behest of humans. There is no example of information that does not reflect this dynamic: the process of info-autopoiesis is at the heart of information creation.

**M**: So, you agree that matter, or in other words, any material thing, can inform.

**J**: I agree that any material thing can inform, but only as a result of prior human action. This would imply that there is no information in matter before living beings engage with such matter. Info-autopoiesis is the key to understand this connection.

This brings us to your second question of whether all living beings create information or only some of them.

My opinion is that **all** living beings, from the simplest unicellular organism to the most complex multicellular organism have the capacity for info-autopoiesis, or the process of self-production of information.

**M**: I can completely agree with you that all living beings create information.

**J**: Now, I will answer your third question of whether all information is created by living beings or only a part of it.

As I mentioned before, all living beings have the capacity for info-autopoiesis, or the process of self-production of information, but this is not all the information that exists in the infosphere. As argued before, human beings are capable of designing, building, and putting into operation systems that are fully capable of generating information, e.g., Information and Communication Technologies (ICTs).

Consequently, the conclusion is: "not only living beings create information but there are other systems that create information". These systems are conceived, designed, and built at the behest of living beings, for the benefit of living beings, as part of the process of info-autopoiesis. The process of info-autopoiesis is part and parcel of this creative process of conceiving, designing, and building systems that have the capacity for creating information.

**M**: Thus, you come to the conclusion that not only living beings but many other systems create information. I would like to add that according to the general theory of information, any system contains information and any active system creates information [11].

**J**: The only caveat that I would add to the general theory of information is that only living systems or systems created by living beings engage in a process of info-autopoiesis or of creating information.

Now, I will answer your fourth question of whether people only create information or they also receive information.

People in the process of information creation engage with the design, construction, and functioning of Information and Communication Technologies (ICTs). These ICTs allow messages to be composed by humans or machines, coded, and optimally transmitted as communication signals that are received, denoised, decoded, and interpreted by humans or machines. So, the conclusion is: "living beings do not only create information but also receive information from other systems".

**M**: Thus, you come to the conclusion that living beings do not only create information but also receive information from other systems. If we try to use Bateson's definition, we have to say that people receive differences. We intuitively know and information theory explains what it means to receive information, but what does it mean to receive differences?

**J**: In the illustrative thermostat example above, we have said that the notions of *difference*, *error,* and *information* can be equivalent in some situations. In the information theory of the communication process, a syntactic process, what is transmitted are a series of 0s and 1s. These 0s and 1s are certainly differences and can be characterized as syntactic information. So, in any communication process, such as a telegraph or a digital telephone, syntactic information in the guise of differences allows messages to be composed by humans/machines, coded, and optimally transmitted as communication signals, which are received, denoised, decoded, and interpreted by humans/machines.

On a more fundamental basis, humans have access to their environment primarily by way of their five major senses: sight, hearing, touch, smell, and taste. Concentrating on our sense of hearing, we find that it comes about as a result of air pressure variations (differences) that impinge on our eardrums and that we learn to interpret in an info-autopoietic process. Similar arguments can be made concerning the other senses. However, what is most noteworthy is that if humans are sensorially deprived, it is tantamount to torture. Sometimes, after a short period of time, individuals feel unease and even experience hallucinations [43]. What this might show is that our senses expect differences in order to function effectively. Of course, this is part of the process of interactivity with the environment, and it exists in the normal course of living and becoming who we are.

**M**: I can agree that variations can be treated as differences but it is strange to understand symbols such as zeroes and ones as differences.

**J**: As you know, Claude Shannon created the possibility of digital communication, which involves the use of syntactic information represented by a series of zeros and ones. Syntactic information is nothing more than the coding of language, images, sound, etc., which humans can recognize in terms of 0 and 1 (binary digits) that allow messages to be composed by humans or machines, coded, and optimally transmitted as communication signals that are received, denoised, decoded, and interpreted by humans or machines. So indeed, the symbols 0 and 1 express a difference, so that these digitalization communication processes can take place.

Now, I will answer your fifth question as to whether people create or produce information from nothing or from something else.

The process of information creation or production clearly needs to be specified.

An organism may be considered, for the most part, as a collection of reflex-actions, i.e., involuntary and nearly instantaneous movements in response to a stimulus. An important assumption here is that reflex-actions are phylogenetic behavior. For example, a reflex-action such as blinking in humans ontogenically is made possible by neural pathways called reflex arcs, which can act on an impulse before that impulse reaches the brain, implying a response to a stimulus that phylogenetically relies on anticipatory ontogenic-derived behavior. If such reflex-actions did not exist, the human organism

would not operate as intended. That is, phylogenetic behavior can be considered the first layer of homeorhesis.

Figure 2 shows a feedback simulation of the organism-in-its-environment (O/E) depicting cyclic self-referenced reflex-action operations to keep homeorhetic trajectories. This figure is drawn with similar elements as those of Figure 1, so to make possible an eventual comparison as to its functioning. This depiction seeks to follow that of Maturana and Varela relating to autopoietic unity [21]. Note that the ORGANISM enclosing line has two arrows on opposite sides denoting a counterclockwise sense. This symbolizes the metabolism of the organism. Additionally, there are two arrows going in opposite directions, to the sides of the organism, to denote the structural coupling of the organism with its environment. The difference in the arrows (one is dashed) denotes the asymmetric relationship between the organism and the environment. Some elementary actions of organisms are phylogenetic reflex-actions, which generally have something to do with keeping our internal milieu within homeorhetic bounds. It is common that feedback control mechanisms can be ascribed to these phylogenetic reflex-actions, although each type of reflex-action obeys its own non-mechanistic homeorhetic requirements. What we would like to elucidate is how Bateson information, *a difference which makes a difference*, may be used to explain how these homeorhetic processes can occur, although not surrogated, in feedback simulations and compare its functioning to the previously presented one-parameter semi-closed-loop cybernetic feedback homeostatic mechanism.

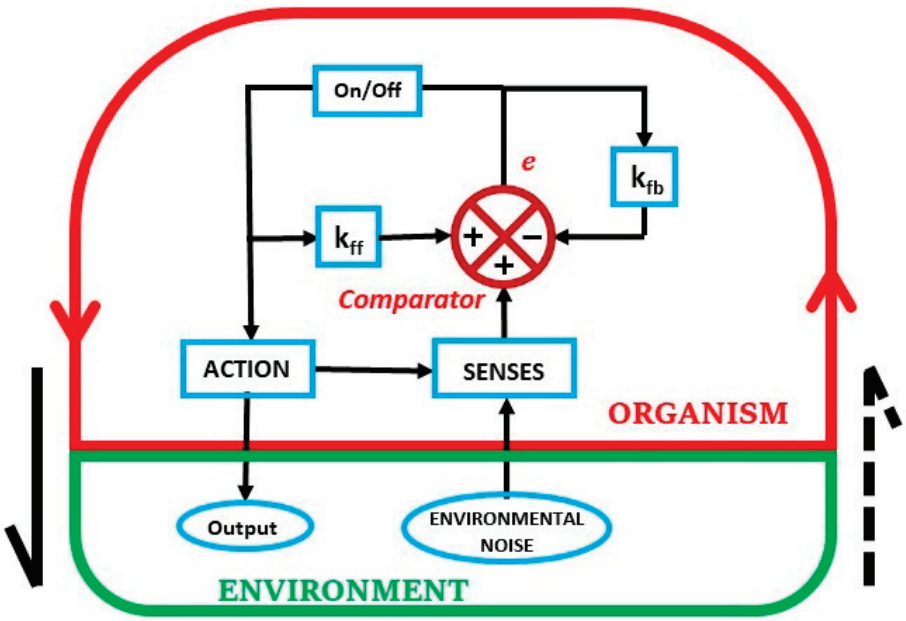



**Figure 2.** A one-parameter feedback loop simulation of homeorhesis.

Referring to Figure 2, consider the beginning of the O/E cyclic interactions as the detection of *environmental noise* (perturbation) by the *senses* of the organism. This is the only window that the O/E has to access the environment. *Environmental noise* is particular to each individual O/E, since each individual O/E has a particular set of senses that are attuned to its phylogenetic and ontogenetic development within a specified environment. The primary motivation of the O/E in sensing the noisy environment that may resemble white noise, particular to the O/E, is to maintain its individuation and homeorhetic trajectories in epigenetic landscapes due to dynamic openness [44,45]. For example, the O/E needs to satisfy its energy needs and is tuned to particular cues in the white noise that leads it to satisfy them. This is true of all our *senses* that permit these cues to synchronize to recognize environmental invariance. In time, the organism is particular only to these cues. This is akin to being able to talk with another person in a noisy room.

The portrayal in Figure 2 defines the fundamental relationship of the O/E as it exists embedded in its environment. There are two essential connections with the environment. The first, shown directly connected to the one-parameter feedback loop, is the single sense element that is the intermediary between the external environment and the internal milieu of the organism. This single sense element represents a microcosm of reality, since a typical human organism is composed of millions of these sense elements that define each particular sense organ in the human body. The second connection is the capacity of the O/E to physically impact the environment, either directly or by other means, including our body, tools, and machines in the case of humans. It is represented by an ACTION that results in an Output to the environment. It must be recognized at the outset that these two essential connections define an asymmetrical relationship between the organism and its environment, i.e., the impact that the organism has on the environment is not a mirror reflection of the impact of the environment on the organism. The intent in what follows is to concentrate on the sensorial side of this dichotomy, as the single sensor element is the only means that an O/E has to ascertain the reality of the external environment to successfully engage it.

Each sense element has a transduction role; it changes the physical (touch, sound, light) or chemical signals (smell, taste) to a corresponding electrical signal or action potential (AP). It is this AP that needs to be interpreted by the human organism, irrespective of origin, either locally or centrally to generate information, *a difference which makes a difference*. In a similar way, as with the thermostat, a *Comparator* is used to show how information is created using this single sensor element to begin the cyclic process of self-referenced information. Info-autopoiesis is the self-generation of information of the organism, in structural coupling, as a corollary of its (M, R)-autopoiesis [46].

This process is akin to the Principle of Undifferentiated Encoding, which is described in the following way:

"The response of a nerve cell does not encode the physical nature of the agents that caused its response. Encoded is only "how much" at this point on my body, but not "what" [47]."

However, the AP that needs to be interpreted, which involves info-autopoiesis or the information that results from the process of info-autopoiesis. As a result, I suggest that the Principle of Undifferentiated Encoding may be alternatively defined in terms of info-autopoiesis. I argue that because of the specificity of the sensors, info-autopoiesis does imply something not just about the "how much" but about the "why", "what", "when", and "where" aspects of cognition. There is greater specificity in its realization. Examining Figure 2 shows that the *Comparator* has a feedback circuit that incorporates a quantity $k_{fb}$ and a feedforward circuit with quantity $k_{ff}$ to modify the error $e$. The feedback signal independently modifies the incoming sensory AP by subtracting a factor $e\,k_{fb}$, while the feedforward signal independently modifies the same sensory AP by adding a factor $e\,k_{ff}$, if and when $e$ is able to overcome the trigger level of the On–Off trigger switch. We note that the feedback circuit represented in Figure 2 is neither a closed-loop cybernetic feedback circuit nor a semi-closed-loop cybernetic feedback circuit. The info-autopoiesis circuit is independent of the resulting actions that stem from its instantiation.

An equation that can be obtained from looking at the *Comparator*, where $e$ is the error and $AP$ is the Action Potential, yields

$$e = AP + ek_{ff} - ek_{fb}$$

leading to,

$$e - ek_{ff} + ek_{fb} = AP$$

which after factoring, we obtain,

$$e\left(1 - k_{ff} + k_{fb}\right) = AP \tag{1}$$

yielding a relationship between input and output given by,

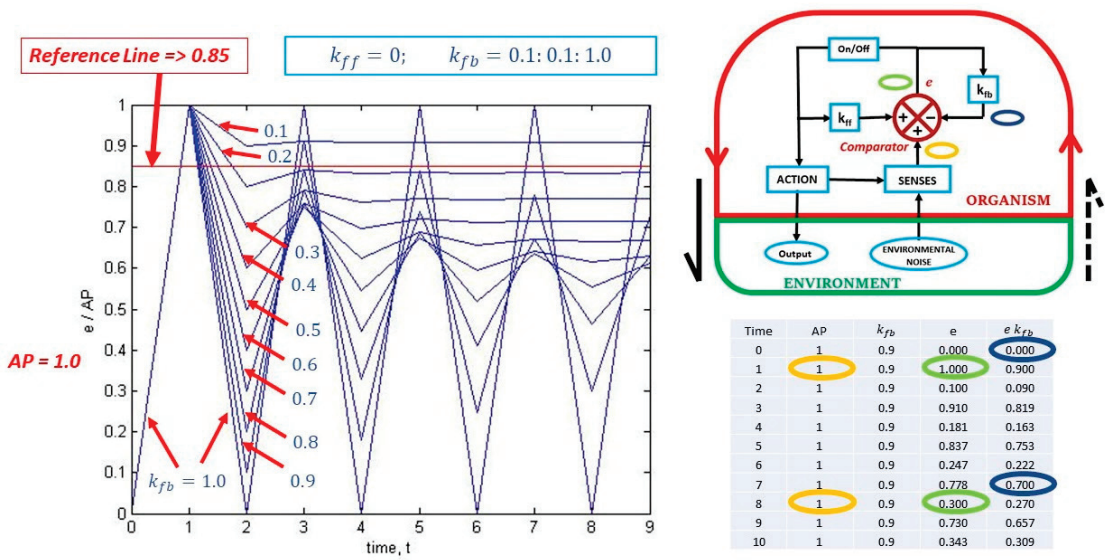
$$\frac{e}{AP} = \frac{1}{\left(1 - k_{ff} + k_{fb}\right)} \tag{2}$$

Each of the quantities $k_{fb}$ and $k_{ff}$ may be regarded as functions of *difference/error/information*, $e$, of time, of historical and other factors particular to the organism/organ under consideration.

The relationship between input and output is capable of many fluctuations, allowing this basic first-order feedback system to be capable of accommodating multifaceted behavior. For example, with this type of approach, reflex-actions and actions requiring a longer fuse may be accommodated. Note also that $k_{ff}$ and $k_{fb}$ do not have to exist simultaneously, or they may be even triggered by differing phenomena. In addition, all settings related to how $k_{fb}$. and $k_{ff}$ come about are internal to the organism, which does not exclude external influences by way of the environment influencing their behavior.

For the moment, consider only that the comparator has a feedback circuit that incorporates a constant $k_{fb}$ to modify the *difference/error/information*, $e$, that is generated as the result of the action of the *Comparator* on the incoming sensory AP and the feedback signal $e\,k_{fb}$, that is, we assume $k_{ff} = 0$. The case for $k_{ff} \neq 0$ is not further considered here, because the intent is to compare the one-parameter feedback loop to that of a mechanism. In general, the effect of the $e\,k_{ff}$ term is to either enhance or conserve $e$, acting similar to a memory function. We further note that in this homeorhetic example that the *error* is not a term that acts as a cybernetic correction factor, since we are not dealing with either a closed-loop cybernetic feedback circuit or with a semi-closed-loop cybernetic feedback circuit. Rather, it reflects a self-referenced comparison of the sensor element. It reflects what the homeorhetic organism identifies as information in the environment.

Figure 3 is a plot of the output over the input, i.e., $e/AP$. In this particular case, a constant value of AP = 1 is used, and $k_{fb}$ varies from 0.1 to 1.0 in decrements of 0.1, resulting in 10 curves generated at 10 time-steps. Note that the color-coding in the two small images on the right side is used to show how the calculation of $e/AP$ is performed. The curves in the graph show the versatility of the reflex-action depending on the value of $k_{fb}$. The curve for a value of $k_{fb} = 1.0$ envelops all the other curves as it oscillates between the values of 0 and 1 over time, implying continuous triggering of the reflex-action. All successive curves show an oscillatory reduction over time. For example, the curve for $k_{fb} = 0.1$ after reaching a peak of 1.0 at time interval 1 has a tendency to be stable around a value of 0.9 after a time interval of 2.




| Time | AP | $k_{fb}$ | e | $e\,k_{fb}$ |
|---|---|---|---|---|
| 0 | 1 | 0.9 | 0.000 | 0.000 |
| 1 | 1 | 0.9 | 1.000 | 0.900 |
| 2 | 1 | 0.9 | 0.100 | 0.090 |
| 3 | 1 | 0.9 | 0.910 | 0.819 |
| 4 | 1 | 0.9 | 0.181 | 0.163 |
| 5 | 1 | 0.9 | 0.837 | 0.753 |
| 6 | 1 | 0.9 | 0.247 | 0.222 |
| 7 | 1 | 0.9 | 0.778 | 0.700 |
| 8 | 1 | 0.9 | 0.300 | 0.270 |
| 9 | 1 | 0.9 | 0.730 | 0.657 |
| 10 | 1 | 0.9 | 0.343 | 0.309 |

**Figure 3.** The effect of decreasing $k_{fb}$ in the generation of richer dynamical behaviors.

Referring again to Figure 2 (inset in Figure 3), note that an On/Off trigger switch is present. This trigger switch will remain On for *difference/error/information*, $e$, values above a certain *reference value*, but it will remain Off below that same *reference value*. Looking at Figure 3, if an arbitrary trigger *reference value* is set to the value $e = 0.85$, the reflex-action will trigger once for all values of $k_{fb}$, but it

will trigger four additional times for the value $k_{fb} = 1.0$, one additional time for a value of $k_{fb} = 0.9$, and it will remain triggered continuously for a value of $k_{fb} = 0.1$. Note that this *reference value*, as well as the value of $k_{fb}$, are both fully defined by the organism in question, depending on many factors, including those mentioned above.

To summarize, the above description gives the context of info-autopoiesis as it relates to phylogenetically derived reflex-actions in the context of the definition of Bateson information: *a difference which makes a difference*. This definition of information implies a quantitative portion (a difference) and a qualitative portion (a difference which makes a difference) [48]. The quantitative portion is the difference/error/information $e = AP - ek_{fb}$ that is evaluated by the *Comparator* that results in the movement of an organism appendage due to the reflex-action; the qualitative portion is the homeorhetic dynamic of the organism. For example, if the feeling of hunger by an individual recurs, the act of eating restores the homeorhetic dynamic as the organism goes about its cognitive business.

In short, here is a four-step conceptualization of how this feedback simulation of the organism consisting of a one-parameter *Sensor* and *Comparator* works:

1. A comparator is set to a reference parameter, which is defined by $ek_{fb}$;
2. The *sensor* (one of the main senses) distinguishes the value of the *detected parameter*;
3. The comparator obtains the difference/error between the detected parameter and the reference parameter;
4. The detected *difference/error* is the *information* needed to either allow or not allow the actuation of a reflex-action.

**M**: You described an interesting example of information production, but it only shows that information is created/produced from something that already exists and functions.

**J**: This example is related to how a living being functions in the realm of info-autopoiesis or the self-production of information. And you are right, this is "an interesting example of information production but it only shows that information is created/produced from something that already exists and functions". In this case, that 'something' that already functions and exists is a living being. Furthermore, that 'something' is the only thing that can create information either directly or indirectly. This is the significance of info-autopoiesis.

Now, I will answer your sixth question of whether only living beings contain/store information or if other natural and/or artificial systems can also contain/store information. Additionally, what does it mean to store difference?

From the above description, not only living beings contain/store information. All types of artifacts, as well as all systems related to Information and Communication Technologies (ICTs), contain/store information. In other words, information is contained/stored in all creations by living beings. Information does not exist in Nature except as a result of the action of all living beings in information creation or info-autopoiesis. If living beings did not exist, there is no information. Or, simplifying, life is information, and information is life. This also implies that there is intrinsic information neither in the genome nor, for that matter, in our environment.

This is in opposition to many information researchers, beginning with Norbert Wiener, who stated that: "Information is information, not matter or energy. No materialism which does not admit this can survive at the present day" [26]. While defining information, in this instance, in terms of itself, Wiener does go on to develop a definition of information [26] that parallels that of Shannon. Further, Wiener's statement implies that information is another fundamental quantity in nature, in addition to matter and energy. However, information is not a fundamental quantity of the universe. Organisms either sense moving matter/energy or shift position/perspective: in a sense creating motion. So, information is a product of the continuous perception of dynamic sensory maps by living beings if you use Bateson's *difference which makes a difference*. Yet Wiener's claim has remained unquestioned by many, including physicists, and his affirmation of the fundamental nature of information is taken as gospel. This poses an impossible quandary for materialism and motivates the persistence of information as an independent

and/or quantitative/objective entity. An interesting question that might be posed to explore these ideas further is: How to explain geological, cosmological, and even biological scientific discoveries as fruits of info-autopoiesis, rather than as the result of taking advantage of intrinsic information in Nature?

Another approach that deals with the existence of information in natural systems, purportedly to deal with questions such as the one posed above, is that proposed by Kun Wu that states:

"Because of the universality of material interactions, and also because there is no beginning of matter and time, no material system in the universe can have remained in its original initial state with no interaction with other material systems. In the term I have coined to describe this state of affairs, any object is a self-evolved "informosome" that has condensed all kinds of information about past, present and future structures and states. Since it is the material nature of an informosome that the properties of any object constitute a unity of direct and indirect existence, this unity also has the triple property of being the source of information, the carrier of information and the information itself at the same time. As a result, we can see the scope of informational existence: Information and matter coexist. Materialists should thus accept that the term information refers to something that can be completely summed up as material but with its own unique mode and status of existence. Due to internal and external objective interactions, objects constantly radiate and reflect quanta to the external world and in this process send information about the mode and state of their own existence depending on the nature of those quanta and their distribution. This shows that the informational ground that produces and reflects the mode and state of physical existence is in the material world itself, in its own physical movement. It is this physical property, showed by it, that makes this world a knowable world." [49].

If I understand Wu correctly, he is postulating the existence of information as co-extensive with the existence of matter and energy, which is a similar perspective to that of Wiener. These are perspectives that ignore the central role of human beings in the process of information. This is clearly in direct opposition to my interpretation of what Bateson proposes.

As to what it means to store difference, this goes back to the equivalence made before among *difference/error/information*. To store information in any media, what you are really storing is differences: writing on a page shows differences; the grooves in a compact disc are also differences that are recorded and then interpreted by a machine, as are all other instances of recorded information in media. This could even be applied to neuronal imprints in our brain that reflect changes in neuronal circuits, apart from the electrical pathways that are enacted due to use and reuse.

**M**: Thus, you want to say that when Wiener and many other researchers assert objective existence of information, it contradicts your interpretation of Bateson's "definition" or in other words, you claim that information does not exist in an objective form.

**J**: Physicists seek to interpret the world in objective terms. What Bateson's "definition" achieves is an interpretation of the world that is subjective and objective at the same time. It is syntactic and semantic at the same time. The question to ask someone that believes in only an objective perspective of the world is: how does a subjective being such as you achieve objectivity? However, the point that is really, at the heart of information is whether or not information predates the existence of living beings: whether it is inherent to nature since the origin of the universe. Bateson's position is that information only exists since the time when life originated in the universe. In addition, a postulate that information is inherent to the universe does not prove its validity. Bateson's argument relies on the existence of the equivalence of matter and/or energy. Living beings are able to detect moving matter and/or energy; therefore, they detect spatial/temporal differences, and this helps them satisfy their physiological and/or social needs. So, it is a question of what you can postulate as primary: matter and/or energy, or, matter and/or energy and information. The primacy of information is a postulate that needs to be proven: not so for matter and/or energy.

**M**: Actually, living beings are also able to detect information in many systems and processes.

**J**: A more accurate statement would be that living beings use their sensory organs to detect the motion of matter/energy that through a process of info-autopoiesis becomes information. Living beings do not have the sensory organs for the direct detection of information.

Now, I will answer your seventh question about the meaning of information transmission, as well as what it means to transmit difference.

My understanding of 'information transmission' is that after the process of info-autopoiesis by living beings, information has the capacity to be transmitted into the surrounding environment. Since the surrounding environment includes other living beings, sharing it with these other living beings would be part of information transmission. Let us narrow the scope to human beings, since this is the sphere that most represents what might be true of other living beings. As generally specified above, we transfer information to our environment in two ways: by in-forming matter by our physical interactions (as illustrated by Bateson above), as well as by the act of communicating with another person. The levels of sophistication achieved by humankind in these two modes of 'information transmission' are impressive when considering the multifarious related technological manifestations. As before, I want to emphasize that *difference/error/information* are equivalent expressions. Therefore, in transmitting information, we are transmitting difference.

**M**: I would like to remind you that before, you stated that information can also be transmitted from inanimate matter to people. Besides, it is unclear why the idea of information as the essence produced only by living beings is implicitly contained in Bateson's conception of information.

**J**: Bateson's conception of information as 'a difference which makes a difference' contains two important aspects that make it amenable to both objective/syntactic and subjective/semantic treatment. The objective/syntactic aspect is given by 'a difference', which for any specific example can be characterized explicitly and quantitatively. The thermostat example clearly shows the objective/syntactic calculation that can be performed to obtain/calculate 'a difference'. At the same time, it cannot be the result of a subjective/semantic judgment of a human being as to whether the setting is the right one, i.e., 'a difference which makes a difference'. For example, it goes without mentioning that the thermostat is a reality just because it is designed and built for a specific purpose by a human being in her efforts to satisfy a perceived need that leads to a more comfortable life with less effort. So, a thermostat, an inanimate object, does transmit information to a living being, but only as a result of the design by that living being. The thermostat is a machine built for a specific purpose by a human being. In that role, it has an information function. Central to any concept of information are human beings, which is also the case for the development of science.

The idea of information as the essence produced only by living beings is implicitly contained in Bateson's conception of information because only living beings can simultaneously generate an objective/syntactic and subjective/semantic judgment of their living. That is the essence of 'a difference which makes a difference'.

**M**: However, we still have the question about the shortcomings of Bateson's "definition" of information from the scientific point of view.

**J**: I don't really understand why there is a question about any shortcomings of Bateson's description of information from the scientific point of view. The various explanations above provide a sufficient scientific basis to Bateson's musings. Bateson's perspective of information has to be examined from an objective/syntactic and also a subjective/semantic perspective. Human beings when engaging in scientific pursuits cannot but start from a subjective perspective, i.e., the subjective drive that feeds their curiosity to postulate personal/subjective hypotheses that are tested, to aspire to achieve results that allow them to be scrutinized by the larger scientific community. If the results pass muster, then they are accepted as scientific truths, which in any case are generally superseded in time by new and more general findings. It has to be recognized that this never-ending process has subjective/objective elements. Indeed, the question that every scientist should be able to answer is: How is it that as a subjective being, you able to achieve the objective perspective that you so revere? As mentioned before, one interesting aspect to Bateson is that the subjective aspect is not far away, in a world where "true" scientists emphasize the objective aspect. This apparent contradiction is certainly something that needs to be taken seriously. How does one of these objective scientists become so, if they start as subjective beings? Does Bateson hold the promise of simultaneous subjectivity and objectivity?

This is why it is important for me to repeat the following quote that encapsulates how we can be in a better position to deal with information if we identify the Fundamental Problem of the Science of Information, which states:

"A key issue in reaching a unified definition of information is the fundamental problem of identifying how a human organism, in a self-referential process, develops from a state in which its knowledge of the human-organism-in-its-environment is almost non-existent to a state in which the human organism not only recognizes the existence of the environment but also sees itself as part of the human-organism-in-its-environment system. This allows a human organism not only to self-referentially engage with the environment and navigate through it, but also to transform it into its own image and likeness. In other words, the Fundamental Problem of the Science of Information concerns the phylogenetic development process, as well as the ontogenetic development process of Homo sapiens from a single cell to our current multicellular selves, all in a changing long-term and short-term environment, respectively [48]."

This process has been at the center of our phylogenetic and ontogenetic development since the beginning of our existence. It is only since the middle of the last century that we can say that we are beginning to ask fundamental questions about our existence.

**M**: Thus, we can see that there are many problems with Bateson's "definition" of information. Besides, there is no analysis of how Bateson's "definition" is related to other definitions of information, for example, to Shannon's entropy. At the same time, there is the general theory of information (GTI) where a comprehensive definition of information is given.

**J**: What has not been said in the above exposition related to Bateson information is that Claude Shannon's perspective of information is subsumed by Bateson. The shortest argument is to recognize that Bateson's difference which makes a difference is both syntactic and semantic in its approach. The syntactic approach by Shannon may be shown to be subsumed by Bateson simply because the binary combination (0, 1) is the difference that is used to develop a mathematical theory of communication. Bateson does not take anything away from Shannon's syntactic approach, but he complements it by adding the semantic portion, which is still under development.

**M**: Yes, your explanation not only makes clearer Bateson's ideas but you elaborate a more precise description of information based on these ideas.

**J**: I expect that this dialogue has provided me a platform from which to expound on the virtues and disadvantages of Bateson's approach to information. Now, I would like to hear something about the general theory of information (GTI).

**M**: The general theory of information (GTI) is an innovative approach, which provides a powerful means for all areas of information studies. It rigorously represents static, dynamic, and functional aspects and features of information. These features are modeled and explored by algebraic, analytical, and topological structures of operators in functional spaces as well as functors in the categorical setting forming information algebras, calculi, and topological spaces. It is possible to get acquainted with the GTI in the book *Theory of Information* [11], but I will briefly explain this theory here.

The first thing that we need to know is that the GTI has three components: the *axiomatic foundations*, the *mathematical core*, and the *functional hull*.

**J**: Please, explain what the axiomatic foundations of the GTI are as well as what the mathematical core and functional hull mean.

**M**: At first, let us look at the axiomatic foundations of the GTI. They consist of the principles, postulates, and axioms of the GTI.

**J**: Interesting, usually postulates and axioms are used in logically formalized mathematical theories. Is the GTI a logically formalized mathematical theory?

**M**: The answer to your question is yes and no, because the GTI has three parts. One is an informal methodological system, another is an operational mathematical theory, and the third is a logically formalized mathematical theory. Thus, the GTI is a methodological system and a logically formalized mathematical theory at the same time.

The first part of the GTI is built on principles, which describe and explain the essence and main regularities of the information terrain. It is possible to find the most detailed exposition of this methodological system in the book [11].

The second part utilizes various mathematical structures such as categories, functors, epistemic spaces, named sets, operators, and transformations. It is possible to get acquainted with the operational mathematical theory of the GTI in the publications [12–14,18,19].

The third part uses postulates and axioms to deduce the properties of information and its functioning. Postulates are formalized representations of principles, while axioms describe mathematical and operational structures used in the general theory of information. This part is only initiated and is still waiting for its full development.

**J**: Do you mean that to understand the GTI, we have to start from the principles?

**M**: You are right. So, let us look at the principles of the GTI. There are two groups of principles. Principles from the first group are called *ontological principles*. They explain the essence of information as a natural and artificial phenomenon as well as depict regularities of information functioning. Here is the list of these principles.

**Ontological Principle O1** (**the *Locality Principle***). It is necessary to separate information in general from information (or a portion of information) for a system *R*.

**Ontological Principle O2 (the *General Transformation Principle*).** In a broad sense, *information* for a system *R* is a capacity to cause or to prevent changes in the system *R*.

**Ontological Principle O3** (**the *Embodiment Principle***). For any portion of information *I*, there is always a *carrier C* of this portion of information for a system *R*.

**Ontological Principle O4** (**the *Representability Principle***). For any portion of information *I*, there is always a *representation C* of this portion of information for a system *R*.

**Ontological Principle O5** (**the *Interaction Principle***). A transaction/transition/transmission of information goes on only in some *interaction* of *C* with *R*.

**Ontological Principle O6** (**the *Actuality Principle***). A system *R* accepts a portion of information *I* only if the transaction/transition/transmission causes corresponding transformations in *R*.

**Ontological Principle O7** (**the *Multiplicity Principle***). One and the same carrier *C* can contain different portions of information for one and the same system *R*.

As you can see these principles can be divided into three groups:

- *Substantial ontological principles* [O1, O2, and its modifications O2g, O2a, O2c] define information.
- *Existential ontological principles* [O3, O4, O7] describe how information exists in the physical world.
- *Dynamical ontological principles* [O5, O6] show how information functions.

**J**: Indeed, these principles provide an organized structural view of how information functions. They allow developing a far-reaching theory of information. However, what about a definition of information?

**M**: Actually, Ontological Principle O2 gives a definition of information in a generalized sense providing possibilities to define a variety of different types and kinds of information. The most important of them are specified in the following modifications of Ontological Principle O2 [11,50]. They based on the concept of an infological system. This concept is used as a variable parameter in the definition of information. To have this ability, an infological system is not formally defined in a general case but is specified for certain types of information. Examples of infological systems are:

- A system of knowledge (thesaurus)
- A system of knowledge, beliefs, ideas, convictions, and principles
- A system of values, estimates and measures
- A system of propositions
- A system of theories

**Ontological Principle O2g (the *Relativized Transformation Principle*).** *Information* for a system *R relative to the infological system* IF(*R*) is a capacity to cause changes in the system IF(*R*) or to prevent such changes.

**Ontological Principle O2a (the *Special Transformation Principle*).** *Information in the strict sense* or *proper information* or, simply, *information* for a system *R*, is a capacity to change structural infological elements from an infological system IF(*R*) of the system *R* or to prevent such changes.

**Ontological Principle O2c (the *Cognitive Transformation Principle*).** *Cognitive information* for a system *R*, is a capacity to cause changes in the cognitive infological system IFC(*R*) of the system *R* or to prevent such changes.

The Ontological Principle O2 and its modifications allow affording a scientific meaning to the before mentioned statement of Bateson. Indeed, according to the Ontological Principle O2, information causes changes. Any change results in some difference between what was before action and after it.

At the same time, taking into account the assertion of Wiener that information is neither energy nor matter, we can consider information as difference. Thus, we come to Bateson's statement "information is difference that makes difference."

However, this is only half of the coin because according to Ontological Principle O2, information also can prevent changes. This results in the absence of difference between what was before and after. Thus, we see even in its more exact and viable interpretation that Bateson's statement does not reflect the concept of information.

It is necessary to remark that this interpretation is exactly what Bateson had in mind when he made his statement about the essence of information. The general theory of information only provides a scientific interpretation of Bateson's idea. In a similar way, physics provided a scientific interpretation of the idea of atoms suggested by ancient Greek philosophers.

**J**: Now I can better see the connection between GTI and the approach of Bateson. However, I would venture to say that Bateson does not exclude differences of differences, which would allow not only a detection of trends, but even an absence of differences. Moreover, it is possible to envision the creation of a mathematical theory with its axioms, theorems, and lemmas based on the Ontological Principles.

**M**: I can agree with you that these principles can be easily converted to postulates and axioms, building a base for a mathematical theory. However, a vital tool of any scientific theory is measurement, which connects theory with reality. Thus, the general theory of information (GTI) needs more principles.

Principles from the second group are called *axiological principles*. They explain how to measure information, which measures of information are necessary, and how to build them.

**Axiological Principle A1 (the *Correspondence Principle*).** A measure of information *I* for a system *R* is some measure of changes caused or prevented by *I* in *R* (for information in the strict sense, in **IF**(*R*)).

This principle shows that what people call Shannon's information is the measure of information elaborated by Claude Shannon.

**Axiological Principle A2 (the *Temporality Principle*).** According to time orientation, there are three types of measures of information: (1) *potential* or *perspective*; (2) *existential* or *synchronic*; and (3) *actual* or *retrospective*.

This principle reflects the importance of time in measuring information.

**Axiological Principle A3 (the *Spatiality Principle*).** According to spatial orientation, there are three types of measures of information: *external*, *intermediate*, and *internal*.

This principle reflects the orientation of changes caused or prevented by information.

**Axiological Principle A4 (the *Determinacy Principle*).** With respect to how the measure is determined and evaluated, there are three constructive types of measures of information: *abstract*, *grounded*, and *experimental*.

This principle reflects the application area of the measure.

**Axiological Principle A5 (the *Totality Principle*).** With respect to information relations, there are three constructive types of measures of information: *absolute*, *fixed relative*, and *variable relative*.

This principle reflects the relativity of measurement.

**Axiological Principle A6** (**the *Scaling Principle***)**.** According to the scale of measurement, there are two groups, each of which contains three types of measures of information: (1) qualitative measures, which are divided into descriptive, operational, and representational measures, and (2) quantitative measures, which are divided into *numerical*, *comparative,* and *splitting* measures.

This principle reflects possible types of measurement scales.

**Axiological Principle A7** (**the *Communication Principle***). The measure of information transmission from a carrier *C* to a system *R* reflects a relation (such as ratio or difference) between measures of information that is accepted by the system *R* in the process of transmission and information that is presented by *C* in the same process.

This principle reflects the acceptability properties of information transmission.

**J**: To make the picture complete, do you need to describe the mathematical core of the GTI?

**M**: The mathematical core of the GTI is the mathematical theory based on the principles of the GTI. Now, there are three approaches to the construction of this theory: algebraic, functional, and categorical approaches. Each of them provides efficient mathematical models. As a result, there are three types of models of information dynamics: information algebras, operator models based on functional analysis, and operator models based on category theory. Functional representations of information dynamics preserve internal structures of information spaces associated with infological systems as their state or phase spaces [11,13,19]. The categorical image of information dynamics displays external structures of information spaces associated with infological systems [12,14]. An algebraic portrayal of information dynamics maintains intermediate structures of information spaces [11]. These models allow researchers to discover intrinsic properties of information.

In addition to its own mathematical information theories, the mathematical core of the GTI includes other information theories, such as Shannon's communication theory [42], quantum information theory [27] or the semantic information theory of Bar-Hillel and Carnap [51], as well as its special sub-theories.

**J**: Now only the third component of the GTI needs to be specified, i.e., the functional hull.

**M**: The functional hull of the GTI consists of methods, measures, and algorithms that allow obtaining properties of information used by people as well as developing more powerful and reliable information processing systems.

**J**: All said, I would like to know whether your theory is only a beautiful abstract edifice far from life of people or if it solves some theoretical and practical problems that other information theories were not able to solve.

**M**: The general theory of information solves several fundamental problems, which are urgent and may be even vital for people because we live in the information age and information has come to the forefront of society and individuals in all countries and all walks of life. Although many information theories have been created, none of them have been able to solve those problems.

Here, we consider only three of those problems.

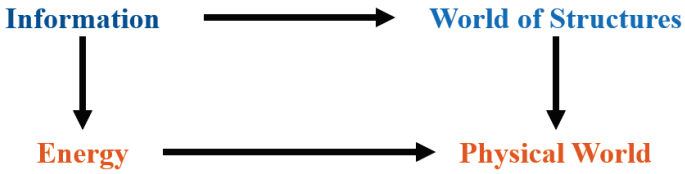
The first Problem is elaboration of the exact and all-inclusive definition of information. Understanding the importance of information, many researchers suggested a variety of information definitions. However, each of them had its shortcomings and boundaries. Moreover, as information was present in a diversity of areas, many researchers started thinking that it would be impossible to elaborate a precise consistent definition of information, which will be good for all these areas. Some researchers even tried to prove this [34,36–38]. However, the general theory of information provides such a definition, which encompasses all areas. This was possible to achieve due to the invention of a new type of definition. Namely, the general theory of information gives a parametric definition of information. Only a parametric definition has been able to unite all existing kinds, types, forms, sorts, and classes of information into one comprehensive and constructive concept.

The second Problem is unification of the theoretical knowledge about information. Many diverse theories of information have been elaborated, giving birth to the problem of creation of a unified theory of information. The general theory of information is a unified theory of information because it

is demonstrated all existing theories of information, treating them as sub-theories of the general theory of information. At the same time, it is necessary to understand that the existence of the general theory of information does not exclude the necessity of special information theories, which can go deeper into specific aspects of information or in particular areas of information habitat.

The third Problem is finding the place of information in the world as a whole. The general theory of information also solves this problem in an innovative way based on thorough observations and detailed reasoning, which are described in other publications on this theory. Many researchers treated information as an enhanced kind of data or as a simplified kind of knowledge. It means that it was assumed that information has the same nature as data and knowledge. The GTI explains that information has an essentially different nature. It is possible to say that information is related to data and knowledge in the same way that energy is related to matter.

Since data and knowledge are types of structures, it is possible to identify the place of information in the world more exactly through displaying it in the following diagram.




As the world of structures is a scientific explication of Plato's World of Ideas [52], we can say that information is energy in the Platonic World of Ideas.

**J**: How did you come to such innovative ideas about the essence of information?

**M**: The work on this theory started at the end of the 20th century when I understood that information is similar to energy while data and knowledge are similar to matter. This idea took me outside the box, in which data, information, and knowledge were treated as essences of the same kind. Having this idea in mind, I started a diligent exploration of the properties of information discovered by other researchers. Thorough analysis of the existing knowledge on information allowed me to formulate and then further develop the ontological and axiological principles of the general theory of information (GTI), building its axiomatic foundations.

**J**: Now, when we discussed our approaches to the definition of information and to the adequate construction of a validated theory of information, let us make some conclusions.

**M**: Indeed, we discussed two different approaches to these problems and can conclude that these approaches do not contradict but rather complement one another. The general theory of information is a comprehensive unified approach to the realm of information, while the difference theory of information suggested by you is a special theory of information. These theories coexist in the world of structures in the same way as the special relativity and general relativity theories coexist in the physical world.

**Author Contributions:** Both authors have equally contributed to the article, its conceptualization, methodology, investigation, writing—original draft preparation, review and editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References


1. Taylor, C.C.W.; Lee, M.-K. The Sophists. The Stanford Encyclopedia of Philosophy. 2016. Available online: https://plato.stanford.edu/archives/win2016/entries/sophists/ (accessed on 17 August 2020).
2. Karamanolis, G. Plutarch. The Stanford Encyclopedia of Philosophy. Available online: https://plato.stanford.edu/archives/sum2020/entries/plutarch/ (accessed on 17 August 2020).
3. Sweeney, E. Literary Forms of Medieval Philosophy. The Stanford Encyclopedia of Philosophy. Available online: https://plato.stanford.edu/archives/sum2019/entries/medieval-literary/ (accessed on 17 August 2020).
4. Knuth, D.E. *Surreal Numbers: How Two Ex-Students Turned on to Pure Mathematics and Found Total Happiness*; Addison-Wesley: Reading, MA, USA, 1974.
5. Roth, W.-M.; McRobbie, C.; Lucas, K.B. Four dialogues and metalogues about the nature of science. *Res. Sci. Educ.* **1998**, *28*, 107–118. [CrossRef]
6. Dodig-Crnkovic, G.; Müller, V. A Dialogue Concerning Two Possible World Systems, Info-Computational vs. Mechanistic. In *Information and Computation*; World Scientific: New York, NY, USA; London, UK; Singapore, 2011.
7. Suzuki, W.A.; Feliú-Mójer, M.I.; Hasson, U.; Yehuda, R.; Zarate, M. Dialogues: The Science and Power of Storytelling. *J. Neurosci.* **2018**, *38*, 9468–9470. [CrossRef]
8. Chilvers, J. Reflexive engagement? Actors, learning, and reflexivity in public dialogue on science and technology. *Sci. Commun.* **2012**, *35*, 283–310. [CrossRef]
9. Phillips, L. *The Promise of Dialogue: The Dialogic Turn in the Production and Communication of Knowledge*; John Benjamins: Amsterdam, The Netherlands; Philadelphia, PA, USA, 2011.
10. Walton, D.N. *Dialog Theory for Critical Argumentation*; John Benjamins Publishing Company: Amsterdam, The Netherlands, 2007.
11. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
12. Burgin, M. Information Operators in Categorical Information Spaces. *Information* **2010**, *1*, 119–152. [CrossRef]
13. Burgin, M. Epistemic Information in Stratified M-Spaces. *Information* **2011**, *2*, 697–726. [CrossRef]
14. Burgin, M. Information Dynamics in a Categorical Setting. In *Information and Computation*; World Scientific: New York, NY, USA; London, UK; Singapore, 2011; pp. 35–78.
15. Burgin, M. Information: Concept Clarification and Theoretical Representation. *TripleC* **2011**, *9*, 347–357. [CrossRef]
16. Burgin, M. Information in the Structure of the World. *Inf. Theor. Appl.* **2011**, *18*, 16–32.
17. Burgin, M. A System Approach to Information Structuring in the Context of Social Interaction, Information: Theories & Applications. *Inf. Theor. Appl.* **2012**, *19*, 3–13.
18. Burgin, M. Evolutionary Information Theory. *Information* **2013**, *4*, 224–268. [CrossRef]
19. Burgin, M. Weighted E-Spaces and Epistemic Information Operators. *Information* **2014**, *5*, 357–388. [CrossRef]
20. Trewavas, A.J.; Baluska, F. The ubiquity of consciousness. *Eur. Mol. Biol. Organ. Rep.* **2011**, *12*, 1221–1225. [CrossRef] [PubMed]
21. Maturana, H.R.; Varela, F.J. Biology of cognition. In *Autopoiesis and Cognition: The Realization of the Living*; Reidel Publishing Company: Dordrecht, The Netherlands, 1980; pp. 5–58.
22. Maturana, H.R.; Varela, F.J. *The Tree of Knowledge—The Biological Roots of Human Understanding*; New Science Library: Boston, MA, USA, 1987.
23. Bateson, G. *Steps to an Ecology of Mind*; Jason Aronson, Inc.: Northvale, NJ, USA, 1972.
24. Bateson, G. *Mind and Nature: A necessary Unity*; E. P. Dutton: New York, NY, USA, 1979.
25. Wheeler, J.A. *Information, Physics, Quantum: The Search for Links, in Complexity, Entropy, and the Physics of Information*; Addison-Wesley: Redwood City, CA, USA, 1990; pp. 3–28.
26. Wiener, N. *Cybernetics: Or Control and Communication in the Animal and the Machine*; John Wiley: New York, NY, USA, 1948.
27. Alber, G.; Beth, T.; Horodecki, M.; Horodecki, P.; Horodecki, R.; Rötteler, M.; Weinfurter, H.; Werner, R.; Zeilinger, A. *Quantum Information: An Introduction to Basic Theoretical Concepts and Experiments*; Springer Verlag: Berlin/Heidelberg, Germany; New York, NY, USA, 2001.
28. Smolin, L. *The Life of the Cosmos*; Oxford University Press: Oxford, UK; New York, NY, USA, 1999.
29. Merriam-Webster Online Collegiate Dictionary. Available online: https://www.merriam-webster.com/dictionary/difference (accessed on 20 February 2020).

30. Cárdenas-García, J.F.; Ireland, T. Bateson Information Revisited: A New Paradigm. *Proceedings* **2020**, *47*, 5. [CrossRef]
31. Cárdenas-García, J.F. The Process of Info-Autopoiesis—The Source of all Information. *Biosemiotics* **2020**, *13*, 199–221. [CrossRef]
32. Varela, F.J. Describing the Logic of the Living: The Adequacy and Limitations of the Idea of Autopoiesis. Available online: https://cepa.info/1908 (accessed on 21 August 2020).
33. MacKay, D.M. *Information, Mechanism and Meaning*; MIT Press: Cambridge, UK, 1969.
34. Capurro, R. *Leben im Informationszeitalter*; Akademie Verlag: Berlin, Germany, 1995.
35. Capurro, R. Past, present, and future of the concept of information. *TripleC* **2009**, *7*, 125–141. [CrossRef]
36. Capurro, R.; Hjorland, B. The Concept of Information. *Ann. Rev. Inf. Sci. Technol.* **2003**, *37*, 343–411. [CrossRef]
37. Capurro, R.; Fleissner, P.; Hofkirchner, W. Is a Unified Theory of Information Feasible. In *The Quest for a Unified Theory of Information, Proceedings of the 2nd International Conference on the Foundations of Information Science, Vienna, Austria, 1996*; Hofkirchner, W., Ed.; Gordon and Breach: Amsterdam, The Netherlands, 1999; pp. 9–30.
38. Fleissner, P.; Hofkirchner, W. In-formatio revisited: Wider den dinglichen Informationsbegriff. *Inform. Forum* **1995**, *3*, 126–131.
39. Peters, J.D. Information: Notes toward a Critical History. *J. Commun. Inq.* **1988**, *12*, 9–23. [CrossRef]
40. Gibbons, A. Strongest Evidence of Early Humans Butchering Animals Discovered in North Africa. Available online: https://www.sciencemag.org/news/2018/11/strongest-evidence-early-humans-butchering-animals-discovered-north-africa (accessed on 30 March 2020).
41. Wong, K. Ancient Cut Marks Reveal Far Earlier Origin of Butchery. Scientific American. 2010. Available online: https://www.scientificamerican.com/article/ancient-cutmarks-reveal-butchery/?print=true (accessed on 17 August 2020).
42. Shannon, C.E. The Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [CrossRef]
43. Bexton, W.H.; Heron, W.; Scott, T.H. Effects of decreased variation in the sensory environment. *Can. J. Psychol. Rev. Can. Psychol.* **1954**, *8*, 70–76. [CrossRef]
44. Von Weizsäcker, C.F. *Die Einheit der Natur*; Deutscher Taschenbuch Verlag: Munich, Germany, 1974.
45. Waddington, C.H. The basic ideas of biology. In *Towards a Theoretical Biology: Prolegomena*; Waddington, C.H., Ed.; Atheneum: New York, NY, USA, 1968; pp. 1–41.
46. Letelier, J.C.; Marín, G.; Mpodozis, J. Autopoietic and (M, R) systems. *J. Theor. Biol.* **2003**, *222*, 261–272. [CrossRef]
47. Von Foerster, H. *Understanding: Essays on Cybernetics and Cognition*; Springer: New York, NY, USA, 2003.
48. Cárdenas-García, J.F.; Ireland, T. The Fundamental Problem of the Science of Information. *Biosemiotics* **2019**, *12*, 213–244. [CrossRef]
49. Wu, K. The Essence, Classification and Quality of the Different Grades of Information. *Information* **2012**, *3*, 403–419. [CrossRef]
50. Burgin, M. The General Theory of Information as a Unifying Factor for Information Studies: The noble eight-fold path. *Proceedings* **2017**, *1*, 164. [CrossRef]
51. Bar-Hillel, Y.; Carnap, R. Semantic Information. *Br. J. Philos. Sci.* **1958**, *4*, 147–157. [CrossRef]
52. Burgin, M. Ideas of Plato in the context of contemporary science and mathematics. *Athens J. Hum. Arts* **2017**, *4*, 161–182. [CrossRef]


© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

*proceedings*

MDPI

*Proceedings*

# Discerning Potential and Impact Information †

**Mark Burgin**

Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

† Conference Theoretical Information Studies, Berkeley, CA, USA, 2–6 June 2019.

Published: 13 May 2020

**Abstract:** This paper contains further analysis of the concept of information aimed at discovering new features of this mysterious but very important phenomenon. We base our analysis on the general theory of information and contemporary theoretical physics. This approach allows for the explication of two basic complementary types of information—potential and impact information. In such a way, we achieve a better understanding of information as a natural and social phenomenon, which serves as a base for developing novel tools for measuring information.

**Keywords:** information; energy; work; force; potential information; impact information; measure

## 1. Introduction

In spite all its importance, information remains a mysterious phenomenon, many properties of which are still waiting to be discovered. Looking at physics, we see that many significant properties of physical systems were discovered by the utilization of physical theories. The same is true for studies of information where there are different theories of information. The most comprehensive is the general theory of information, which encompasses many known information theories, including, among others, Shannon's information theory, algorithmic information theory, quantum information theory and economic information theory [1]. The general theory of information illuminates intrinsic relations between information theory and physics, making possible to use physical theories for exploration of information. Here, we use this theory for the analysis of information dynamics, starting our exposition with the necessary information about the general theory of information.

## 2. Elements of the General Theory of Information

The general theory of information (GTI) is an innovative theoretical system with three components: the axiomatic foundation, mathematical core and functional hull [1].

The *axiomatic foundation* of the general theory of information consists of principles, postulates and axioms as well as of their consequences:

*Principles* describe and explain the essence and main regularities of the information terrain.

*Postulates* are formalized representations of these principles.

*Axioms* describe mathematical structures used in the general theory of information.

The *mathematical core* of the general theory of information consists of mathematical theories based on the axiomatic foundation of the general theory of information.

The *functional hull* of the general theory of information contains informal theories based on the axiomatic foundation of the general theory of information, as well as applications of the general theory of information.

There are two classes of principles, the formalized representation of which constitute postulates of the general theory of information:

- *Ontological principles* clarify the essence of information as a natural and artificial phenomenon.
- *Axiological principles* describe how to evaluate information and what measures of information are necessary.

There are seven main and several additional ontological principles, which are divided into three groups:

- *Substantial principles* [O1, O2 and its modifications O2g, O2a, O2c] define information.
- *Existential principles* [O3, O4, O7] describe how information exists in the physical world.
- *Dynamical principles* [O5, O6] show how information functions.

In what follows, we use ontological principles O2 and O2g.

**Ontological Principle O2 (the *General Transformation Principle*).** In a broad sense, *information* fora system *R* is a capacity to cause or prevent changes in the system *R*.

Thus, we may understand information in a broad sense as a capacity (ability or potency) of things, which can be material, mental or abstract, to change other things or to prevent changes. In such a way, the ontological principle O2 provides a definition of information in a broad sense.

Our experience and scientific studies exhibit that information exists in the form of *portions* or *pieces of information*. For instance, we have such portions or pieces of information as information in a word, information in a letter or information in a book.

It is possible to define a *portion of information* as information contained in one information carrier. Examples of information carriers are physical letters, the memory of a computer, a database, or flashcard.

Our experience and scientific studies exhibit that information exists in the form of *portions* or *pieces of information*. For instance, we have such portions or pieces of information as information in a word, information in a letter or information in a book.

It is possible to define a *portion of information* as information contained in one information carrier. Examples of information carriers are physical letters, the memory of a computer, a database, or flashcard.

In a similar way, it is possible to define a *piece of information* as information contained in one information representation. Examples of information representations are texts, signs, symbols or speeches.

There is also the concept of a *slice of information*, which is a portion of information about some object (domain, system or subject). Examples of information slices are information about the Earth, information about the Moon or information about Sherlock Holmes.

Algorithmic complexity $C(x)$ is a measure of a slice of computational information about object $x$.

The ontological principle O2 unifies dynamic aspects of reality because information in a broad sense is projected onto three primal components of reality:

- physical reality
- Mental reality
- Structural reality

As a result, the ontological principle O2 also amalgamates the following three fundamental conceptions into one comprehensive concept:

- Information
- Physical energy
- Mental energy

Being extremely wide-ranging, this definition supplies meaning to and an explanation of:

- The conjecture of von Weizsäcker that energy might in the end turn out to be information.
- The aphorism of Wheeler *It from Bit.*
- The statement of Smolin that the three-dimensional energetic world is the flow of information.

The ontological principle O2 unifies three fundamental concepts but does not allow for differentiation between them. Thus, we need one more principle for identifying information in the strict sense.

**Ontological Principle O2g** (**the *Relativized Transformation Principle***). *Information* for a system *R relative to the infological system* IF(*R*) is a capacity to cause or prevent changes in the system IF(*R*).

The concept of infological system plays the role of a free parameter in the general theory of information, providing for representation of different kinds and types of information in this theory. This is why the concept of *infological system*, in general, should not be limited by boundaries of exact definitions. A free parameter must really be free. Identifying an infological system IF(*R*) of a system *R*, we can define different kinds and types of information.

To build an efficient model of information, it would be dexterous to employ the concept of an *infological space*, points of which would serve as representations of the states of an infological system.

## 3. Varieties of Information

According to the second ontological principle O2 (the General Transformation Principle), of the general theory of information, information plays the same role in the World of Structures as energy plays in the Physical (material) World [2]. Physicists studied energy much longer than researchers explored information. It makes learning what physicists discovered about energy and applying this knowledge to the study of information possible.

In general, physicists treat energy as work capability or the ability to do work (cf., for example, [3]). In this context, work is viewed as physical activity involving physical effort done to produce some organized changes in physical systems. At the same time, in many situations, it is necessary to do work to prevent changes. For instance, an individual receives information that James Clerk Maxwell was a great physicist who was born in 1831 and passed away in 1879. Some people can remember this information without effort. However, the majority needs to perform definite work, for example, repeating these data, writing them down on paper or in a computer, to preserve this information in the memory. Moreover, when time passes, people are prone to forgetting. As a result, the preservation of information demands performing new work [4].

Another example of work done for preservation is related actually to any building. Natural forces and sometimes people tend to destroy it. Keeping it the same, i.e., eliminating changes, demands definite restoration work. Besides, we know that the preservation of endangered species in nature also demands some work.

Note that preservation is the prevention of changes in an existing system and it is possible to consider the prevention of changes in this system as a change in the conditional configuration space.

However, energy does not directly cause or prevent changes. To do this, energy has to be transformed (converted) to force and then this force changes a physical system or preserves it from changes.

Analyzing the concept of information with the ontological principles of the general theory of information, we find that, according to the second ontological principle O2a (the Special Transformation Principle), information, per se, is the capacity or capability to change (transform) structural systems, such as knowledge systems [5,6]. At the same time, we know that this capacity can be actual, when information induces changes (causes transformation), or potential, when there are no actual changes (transformation). This gives two types of information—potential information and impact information.

*Potential information* $I_P$ reflects only a possibility of changes (transformations), while *impact information*, or *information force*, $I_F$ performs changes (transformation).

In physics, energy is measured using the concept of *work* and physicists elaborated exact mathematical descriptions of work. In kinematics, work is defined by a spatial integral of the force acting on an object and is described by the following mathematical formula:

$$W = \int \overline{F} \cdot \mathrm{d}\overline{x}$$

In this formula, $\overline{F}$ denotes the force acting on the object and $\overline{x}$ denotes the special displacement of the object.

When the force $\overline{F}$ is constant, we have the simpler formula:

$$W = \overline{F} \cdot S$$

where $S$ is the displacement.

In this context, it is possible to define *information work* $W_I$ (which is also called *structural work*) as the size (amount) of structural changes (transformation) times the measure (amount) of impact information.

In physics, forces and infinitesimal displacements have direction and thus are modeled by vectors. In the case of information, if the infological space of action is modeled by a vector space or a topological manifold, impact information and infological displacement also become vectors.

Let us apply this concept to epistemic information, which changes the knowledge of a system, because information theory elaborated definite measures of knowledge and knowledge transformations. We will use here the epistemic measure studied in [5,7].

To measure epistemic displacement, we describe an epistemic space as a network of finite sets (systems) of knowledge units defining the distance between two units equal to 1 as the first approximation to the epistemic metrics. For instance, if knowledge is represented by logical formulas of the proposition calculus, we take elementary propositions as units of knowledge. It is also possible to take all propositions as units of knowledge.

At first, let us figure out how to measure displacements in an epistemic space. The transformation of knowledge systems is performed by two elementary operations: the addition of a knowledge unit and the elimination of a knowledge unit. Each such operation determines a displacement of the length equal to 1. Thus, when a knowledge system $R$ is transformed into a knowledge systems $P$, the displacement is equal to the number of elementary operations performed in this transformation. We call this process $P$ knowledge unit exchange and measure it by the number of performed elementary operations $N(P)$.

Assuming that the transformation (change) is performed under the action of impact information (information force), we have to measure this force. In the case of epistemic information, it is natural to suppose that information comes to the system in the form of data. Then, it is possible to measure the force by the size of input data, e.g., in bits or in bytes.

This gives us the following mathematical formula for information work:

$$W_I = m(I_F) \cdot D$$

where $D$ is the structural displacement.

Examples of other measures of epistemic information are Shannon's entropy [8], Hartley measure [9], Rényi entropy [10], Fisher information [11] and some others. It would be interesting to build measures for information work using these measures.

After defining information force, a crucial question is how to measure it. Let us consider this problem in the context of such a mental infological system as a thesaurus or system of knowledge. In this case, there are several measures to estimate changes of knowledge—cognitive displacement. As we already discussed, it is possible to use Shannon's entropy or knowledge unit exchange for this purpose.

Let us consider two processes—learning some material and decoding a coded text—to measure information force. In these cases, it is possible to suppose that information force $I_F$ is proportional to the mental effort *Eff* in this process, i.e., $m(I_F) = k \cdot m(Eff)$. Thus, to find the measure of the information force, we need to determine the coefficient of proportionality $k$ and to measure the mental effort in these processes.

The mental effort of an individual who learns some portion of knowledge or decodes a coded text can be measured by the time of this learning or decoding. It is also possible to determine mental effort measuring the intensity of the brain activity during learning or in the process of decoding.

In general, and in these processes in particular, it is essential to discern instantaneous information force and average information force, as well as instantaneous mental effort and average mental effort.

## 4. Conclusions

Two basic complementary types of information—potential and impact information—are described and studied. Adequate tools for measuring these two types of information are discussed. The next step in this direction is the elaboration of a mathematical theory of potential and impact information, in which the latter can be modeled by action vectors or operators in an infological space.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
2. Burgin, M. Information in the Structure of the World. *Inf. Theor. Appl.* **2011**, *18*, 16–32.
3. Shope, R.K. Physical and Psychic Energy. *Philos. Sci.* **1971**, *38*, 1–12.
4. Anderson, J.R.; Bower, G.H. *Human Associative Memory*; V.H. Winston and Sons: Washington, DC, USA, 1973.
5. Burgin, M. Weighted E-Spaces and Epistemic Information Operators. *Information* **2014**, *5*, 357–388.
6. Burgin, M. The General Theory of Information as a Unifying Factor for Information Studies: The noble eight-fold path. *Proceedings* **2017**, *1*, 164, doi:10.3390/IS4SI-2017-04044.
7. Mizzaro, S. Towards a theory of epistemic information. In *Information Modelling and Knowledge Bases*, *12*; IOS Press: Amsterdam, The Netherlands, 2001; pp. 1–20.
8. Shannon, C.E. The Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423.
9. Hartley, R.V. Transmission of information. *Bell Syst. Tech. J.* **1928**, *7*, 335–363.
10. Rényi, A. On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*; The Regents of the University of California: Berkeley, CA, USA, 1961; pp. 547–561.
11. Fisher, R.A. *Statistical Methods and Scientific Inference*; Oliver and Boyd: Edinburgh, UK, 1956.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

Proceeding Paper

# World Structuration and Ontological Information †

**Mark Burgin [1,*] and H. Roman Krzanowski [2]**

[1] Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA
[2] Faculty of Philosophy, The Pontifical University of John Paul II in Krakow, Kanonicza Street 9, 31-002 Kraków, Poland; mkrzan@gmail.com
* Correspondence: markburg@cs.ucla.edu
† Presented at the Conference on Theoretical and Foundational Problems in Information Studies, IS4SI Summit 2021, online, 12–19 September 2021.

**Abstract:** Based on the general theory of information (GTI), we study ontological information, which is complementary to the concept of mental information in general and epistemic information in particular.

**Keywords:** information; ontological information; mental information; epistemic information; physical energy; psychic energy; general theory of information

**Citation:** Burgin, M.; Krzanowski, H.R. World Structuration and Ontological Information. *Proceedings* **2022**, *81*, 93. https://doi.org/10.3390/proceedings2022081093

Academic Editor: K. Markov

Published: 30 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Diversity of realms where people encounter information resulted in overlooking many kinds and types of information without a unifying concept and consent about the essence of the information phenomenon. Some researchers have even predicted the impossibility of finding such a unifying concept. However, in spite of all difficulties with understanding information, the general theory of information (GTI) was elaborated, giving the all-encompassing definition of information and comprising the variety of existing information theories [1,2]. This definition predicted not only a variety of specific kinds and types of information that were observed before, but also clustering of these kinds and types into a much bigger classes. Regardless of this prediction and observations of biologists, researchers continued to associate information with knowledge, either directly by ascribing them the same nature or indirectly by implying that information is epistemic (or epistemological), and thus, being a mental phenomenon.

According to the GTI, the second approach gave a correct but still incomplete image of information. The situation changed when Roman Krzanowski discovered the existence of the class of ontological information, which is complementary to the concept of epistemic information [3–5]. Here, based on the GTI, we further develop the concept of ontological information. To understand this discovery in the context of the existing knowledge in the domain of information in general and the GTI in particular, we start with the description of the global structure of the world and then continue with discussing the principles of the GTI, which describe information as an intrinsic phenomenon of the world. In Section 4, we explore ontological information as a natural phenomenon. In Section 5, we consider relations between ontological information and physical energy. In Section 6, we contemplate relations between mental information and mental energy.

## 2. The Existential Triad of the World

With the advance in science, scientists began reducing the whole world to the physical (material) reality and rejecting religious views of other realities as nonscientific fantasies. Only some outstanding thinkers were able to see further than this, going beyond the material domain on the rational basis. The most formidable and, at the same time, mysterious extension of the physical world was made by Plato, who introduced and defended the idea

of the world of Forms or Ideas [6]. However, for a long time, this world was not correctly understood without scientific elucidation.

Another extension of the physical world was described by René Descartes, who suggested the dualistic approach to reality, dividing the human being into the body (as a part of the Material World) and the mind (as a part of the Mental World) [7].

Karl Raimund Popper attempted to provide a unified scientific image of the ideas of Plato and Descartes in a unified triadic structure [8,9] but his representation of mental reality was essentially incomplete and his description of the world of ideas was incorrect, as other philosophers explained.

A consistent comprehensive scientific global structure of the world was described and modeled by the Existential Triad of the World [10] (cf. Figure 1).

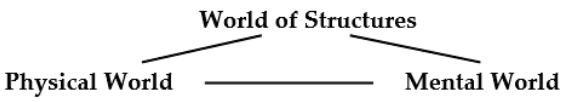



**Figure 1.** The Existential Triad of the World.

In the Existential Triad of the World [10], the Physical (Material) World represents the physical reality studied by natural and technological sciences, the Mental World encompasses different forms and levels of mentality, the lower levels of which are studied by psychologists and sociologists, and the World of Structures consists of a diversity of ideal structures, which is the scientific personification of the world of Plato Ideas or Forms [11]. While the Physical and Mental Worlds are accessible by human senses, the World of Structures can be achieved only by the intellect as Plato predicted.

It is possible to learn more about the Existential Triad of the World in [10,11].

## 3. Principles of Information Ontology

In the general theory of information, the definition of information in the broad sense is given in the second ontological principle, which has several forms [1,2].

**Ontological Principle O2 (the *General Transformation Principle*).** In abroad sense, *information* for a system *R* is the potentiality/cause of formations and transformations (changes) in the system *R*.

Thus, we may understand information in a broad sense as a capacity (ability or potency) of things—material, as well as mental and abstract—to change other things. Information exists in the form of *portions*, *pieces*, or *instances of information*.

However, the common usage of the word *information* does not imply such wide generalizations as the Ontological Principle O2 implies. To define information per se, the GTI uses the concept of an *info logical system* IF(*R*) of the system *R* for the information definition. Elements from IF(*R*) are called *info logical elements*.

**Ontological Principle O2a** (**the *Special Transformation Principle***). *Information in the strict sense; proper information;* or, simply, *information* for a system *R*, is the potentiality/cause of formations and transformations (changes) of the structural info logical elements from an info logical system IF(*R*) of the system *R*.

Information in the strict sense is stratified according to the global structure of the world represented by the *Existential Triad* of the world, which is composed of the top-level components of the world as a unified whole reflecting the unity of the world. This triadic structure is rooted in the long-standing tradition coming from Plato and Aristotle and consists of three components: the Physical (Material) World, the Mental World, and the World of Structures [10,11].The Physical (Material) World represents the physical reality studied by natural and technological sciences, the Mental World encompasses different forms and levels of mentality, and the World of Structures consists of various kinds and types of ideal structures.

## 4. Ontological versus Mental Information

All living organisms are autopoietic and cognitive. Autopoiesis refers to a system where The Existential Triad entails the differentiation of information into two comprehensive classes: ontological information and mental information.

*Ontological information* is the potentiality/cause of formations and transformations of structures in the physical world, i.e., of physical systems.

As ontological information functions in the physical world, it is natural to treat it as a natural phenomenon [3].

*Mental information*, e.g., epistemic information, is the potentiality/cause of formations and transformations of structures in the mental world, i.e., of mental systems.

Ontological information is orthogonal and complementary to mental information. Epistemic information, which has been studied by different researchers, is a type of mental information and, thus, it is orthogonal to ontological information.

It is possible to ask the question of how information belonging to the World of Structures can act on physical systems. To solve this puzzle, we introduce two levels of ontological information: $\text{information}_{\text{IO}}$, which belongs to the World of Structures, and ontological $\text{information}_{\text{O}}$, which belongs to the Physical World and is studied in [3–5].

Connections between these two levels of ontological information are explained utilizing two more ontological principles of the GTI—the Embodiment Principle O3 and the Representability Principle O4. They postulate existence of representations and carriers of information, which in particular, can belong to the Physical World. In this framework, we see that $\text{information}_{\text{O}}$ is a physical (material) representation of $\text{information}_{\text{IO}}$. $\text{Information}_{\text{IO}}$ is embedded in physical objects becoming $\text{information}_{\text{O}}$ and acquiring theability to act on physical systems. In the same way, the mind embedded in the human body is able to operate in the physical world.

## 5. Ontological Information versus Physical Energy

According to the general theory of information, (physical) energy is a kind of generalized information, which is situated in the physical world, being a potentiality/cause for changing or preserving physical systems. Ontological information is the genuine information but it is also a potentiality/cause for changing or preserving physical systems because it acts in such a way on physical systems. In spite of this similarity, there is an essential difference between energy as generalized information and ontological information—namely, energy directly acts on physical systems while ontological information acts only on physical systems having a physical representation and being embedded in a physical carrier. In particular, ontological information can have physical energy as its representation.

## 6. Mental/Epistemic Information versus Mental/Psychic Energy

A similar relation exists between mental information and mental/psychic energy. According to the general theory of information, mental/psychic energy as generalized information in the mental world is a potentiality/cause for changing or preserving mental systems. Mental information, e.g., epistemic information, is the genuine information but is also potentiality/cause for changing or preserving mental systems because it acts in such a way on mental systems. In spite of this similarity, there is an essential difference between mental/psychic energy as generalized information and mental information—namely, mental/psychic energy directly acts on mental systems while mental information acts only on systems with a mental representation and embedded in a mental carrier. An example of such mental systems is knowledge situated in the mentality (mind) of people. In particular, mental information can have mental energy as its representation.

## 7. Conclusions

The concept of ontological information in the context of the general theory of information clarifies and resolves several problems that have plagued many previous studies of information, including the question of the relations between physical carrier and epistemic

information, the nature of the relations between energy and information, the nature of causality of information in mental and physical worlds, and the fusion of information with meaning in physical systems [1,2].

Moreover, the proposed conceptual framework provides a reliable basis for further studies of information ontology, information causality, and information processes as computation in natural systems.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. *Theory of Information*; World Scientific Publishing: New York, NY, USA, 2010.
2. Burgin, M. The General Theory of Information as a Unifying Factor for Information Studies: The noble eight-fold path. *Proceedings* **2017**, *1*, 164. [CrossRef]
3. Krzanowski, R. Ontological Information. Investigation into the Properties of Ontological Information. Ph.D. Thesis, UPJP2, Kraków, Poland, 2020. Available online: http://bc.upjp2.edu.pl/dlibra/docmetadata?id=5024&from=&dirids=1&ver_id=&lp=2&QI= (accessed on 15 March 2022).
4. Krzanowski, R. What Is Physical Information? *Philosophies* **2020**, *5*, 10. [CrossRef]
5. Krzanowski, R. Why can information not be defined as being purely epistemic? *Philos. Probl. Sci.* **2020**, *68*, 37–62.
6. Plato. *The Collected Dialogues of Plato*; Princeton University Press: Princeton, NJ, USA, 1961.
7. Descartes, R. *The Philosophical Writings of Descartes*; Cambridge University Press: Cambridge, UK, 1984.
8. Popper, K.R. Replies to my critics. In *The Philosophy of Karl Popper*; Open Court: La Salle, IL, USA, 1974; pp. 949–1180.
9. Popper, K.R. *Objective Knowledge: An Evolutionary Approach*; Oxford University Press: New York, NY, USA, 1979.
10. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
11. Burgin, M. Ideas of Plato in the context of contemporary science and mathematics. *Athens J. Humanit. Arts* **2017**, *4*, 161–182. [CrossRef]

proceedings

Proceeding Paper

# Physical Facets of Information, Information Materialization or the Grand Illusion †

**Mark Burgin**

Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA; markburg@cs.ucla.edu

† Presented at the Conference on Theoretical and Foundational Problems in Information Studies, IS4SI Summit 2021, online, 12–19 September 2021.

**Abstract:** Many researchers believe that information is physical. The goal of this paper is to show thatthis is only a grand illusion while in reality information in the strict sense belongs to the World of Structures, which is the scientific interpretation of the World of Plato Ideas or Forms. In addition, it is studied how the ideal information acquires a physical representation and is embedded in a physical carrier in the process of materialization. The process of idealization, which is inverse to materialization and in which information is extracted from its physical carrier as an ideal essence, is also discussed. People do not often make a distinction between abstract and ideal objects. That is why, herein, it is explicitly defined what an abstract object is and how an ideal object comes into being elucidating the difference between these entities.

**Keywords:** information; world of structures; physical world; mental world; system; carrier; representation; materialization; idealization

**Citation:** Burgin, M. Physical Facets of Information, Information Materialization or the Grand Illusion. *Proceedings* **2022**, *81*, 76. https://doi.org/10.3390/proceedings2022081076

Academic Editor: Rao Mikkilineni

Published: 24 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Information representsan important phenomenon in nature, society, and technology. At the same time, the huge variety of information manifestations beara common feature—when people encounter information, it is associated with some physical entity. As Bawden and Robinson write, "whenever we find information, we find it inscribed or encoded somehow in a physical medium of whatever kind" [1]. This situation brought many researchers to the conclusion that information is physical (cf., for example, [2–7]). Moreover, in the same venue, Deutsch and Marletto write, "Despite being physical, information also has a counterfactual character . . . " [8].

However, as it is demonstrated here, this is only a grand illusion. Information in the strict sense belongs to the World of Structures, which is the scientific manifestation of the World of Plato Ideas or Forms [9]. In addition, it is studied how the ideal information acquires a physical representation and is embedded in a physical carrier in the process of materialization. The process of idealization, which is inverse to materialization and in which information is extracted from its physical carrier as an ideal essence, is also discussed. Researchers often do not make a clear distinction between abstract and ideal systems. That is why here it is described explicitly what an abstract object is and how an ideal object comes into being, making possible the transparent understanding of the difference between abstract and ideal entities.

## 2. The Global Structure of the World

To understand the essence of information and its place in reality, we need to have a clear vision of the global structure of the world. This structure is described and modeled by the Existential Triad of the World [9] (cf. Figure 1).

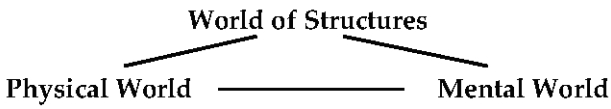



**Figure 1.** The Existential Triad of the World.

The Existential Triad of the World consists of three components: the Physical (Material) World, the Mental World, and the World of Structures [9]. The Physical (Material) World represents the physical reality studied by natural and technological sciences, the Mental World encompasses different forms and levels of mentality, and the World of Structures consists of various kinds and types of ideal structures.

There is scientific evidence of the distinction between the Physical (Material) World and the Mental World. The Mental World, as a rule, contains an image of the physical world. However, it can contain much more. For instance, according to the psychological object-relation theory of Melanie Klein, from the earliest moments of life, children construct imaginary "phantasies" creating the world of the "unreal real" [10,11]. Adults, especially creative individuals, also create mental images that do not exist in reality.

The most mysterious is the World of Structures. It is demonstrated that this world is the scientific incarnation of the world of Plato Ideas or Forms [12]. While the Physical and Mental Worlds are accessible by human senses, the World of Structures can be achieved only by the intellect as Plato predicted. To better understand the World of Structures, it is helpful to envision its necessity for the completion and elucidation of the interplay between two sensible worlds.

Indeed, it is known that there are different physical systems that bearthe same structure. In this situation, we can speak about physical structures that are instantiations or reflections of some ideal structure and in this sense, the physical systems have the same structure. At the same time, one physical system can have several structures. In this case, it is useful to assume existence of the unified ideal structure of the considered system while the physical structures are reflections of this unified structure. In such a way, ideal structures complete the configuration of the world as a whole.

This conclusion is also supported by the **straight line argument**, which is described below.

*One of the basic concepts of the Euclidean geometry is a straight line. It is infinite in both directions and continuous in the classical sense. Assuming these properties, we ask whether it exists in the Physical World or in the Mental World.*

*Analyzing the first possibility, we observe that in the physical reality, there are no infinite objects and all lines cannot be continuous in the classical sense because they are formed from molecules and atoms.*

*Excluding the first possibility, it is possible to suggest that a straight line from the Euclidean geometry belongs to the Mental World, for example, in the individual mentality of mathematicians. However, if we analyze abstract objects in people's mentality, we see that there is no even a single infinite and continuous straight line there. In mentality, we have only descriptions and images of such a line because all images in the mentality are finite.*

*Thus, we come to the dilemma either to assume that such object as a straight line from the Euclidean geometry does not exist or to postulate such a world where it exists. This is exactly what Plato did for all general concepts and the World of Structures is the scientific instantiation of the world of Plato Ideas or Forms as it is demonstrated in* [12]. *Acknowledging the existence of this world, we can deem that a straight line from the Euclidean geometry exists in this world.*

*This shows that to have such an object as a straight line from the Euclidean geometry, we need the World of Structures.*

Here is another observation in support of the necessity of World of Structures. Let us take an actual infinite set, even something assimple as the set $\boldsymbol{N}$ of all natural numbers. As we know, in physical reality, there are no infinite objects. In individual mentality, there are only descriptions and/or images of $\boldsymbol{N}$. At the same time, to have a meaningful set of mathematics, we need such an object as $\boldsymbol{N}$. Thus, we come to the conclusion that this infinite set exists only in the World of Structures and to have this object, we need this

world as a component of the world as a whole.The World of Structures makes the whole world complete.

This situation is analogous to some procedures in mathematics. For instance, the field of rational numbers is rendered complete by introducing irrational numbers, understanding the existence of which demands an essential mental effort. One more example is given by introduction of the point at infinity in projective geometry with the goal of completing geometrical spaces. In a similar way, it is possible to conceive the World of Structures as being situated at infinity from the physical world.

## 3. Materialization and Idealization of Information

In contrast to the opinion regardingthe physical nature of information, the general theory of information (GTI) places information per se in the ideal World of Structures, which is the scientific incarnation of the World of Plato Ideas or Forms [12]. According to the Ontological Principle O2 of the GTI and its additional forms [9] information plays the same role in the World of Structures as energy plays in the Physical (Material) World.

Note that although some researchers did not agree that information is physical, they did not identifyan appropriate place for information in our world [1,13].

We see that positioning of information in the World of Structures [14] looks contradictory to the assumption that information is physical and to the fact of the incessant presence of information in nature, society, and technology. The goal of this work is to solve this paradox explaining the connections between the ideal and material and further developing the approach to materialization introduced in [15].

According to the Ontological Principle O2 of the GTI and its additional forms [9] information plays the same role in the World of Structures as energy plays in the Physical (Material) World.

However, according to the Ontological Representability Principle (Ontological Principle O3) of the GTI, for any portion of information $I$,there is always arepresentation $Q$ of this portion of information for a system $R$. Often this representation is material, and as a result, being materially represented, information becomes physical. Consequently, a physical representation of information can be treated as the materialization of this information.

Moreover, according to the Ontological Embodiment Principle (Ontological Principle O4) of the GTI, for any portion of information $I$,there is always acarrier $C$ of this portion of information for a system $R$. This carrier is, as a rule, material, and this makes information even more physical. A physical carrier of information can alsobe treated as the materialization of this information, or more precisely, the materialization of the second level.

Now we can see that the paradox of the existing impact of such an ideal essence as information in the physical reality is caused by the very popular confusion of information per se, its representations, and carriers.

To better explain this, clarifying the meaning of materialization and why being ideal, information only seems physical to people, it would be useful to employthe Flying Metaphor.

## 4. The Flying Metaphor of Materialization

*If we ask the question whether people can fly, the answer will be yes and no. Yes, an individual can take a plane or a helicopter and fly to another city, another country or even another continent. The answer is also no, because without technical means people cannot fly. So, to fly, an individual has to embed herself or himself into a technical device designed for flying*.

In a similar way, to come to the physical world of people, information, which belongs to the ideal world of structures, must be embodied into a physical carrier. The process of this embodiment is called materialization.

Some researchers argue that information is physical because it acts on physical things. This is similar to the argument that birds can fly and people are birds because people can fly.

The difference between a portion of information, its representation, and its carrier is demonstrated by the following example. Let us consider a letter/text written/printed on a piece of paper. Then the text is a representation of information in this text while the piece

of paper is a carrier of this information. Note that the text is not information since the same information can be represented by another text.

This difference is the distinction between the essence of a phenomenon and its appearance where the representation of information is its appearance of the first order while the carrier of information is its appearance of the second order. There are also carriers that are appearances of higher orders. For instance, a text on a piece of paper is a representation of information in this text, the piece of paper is the carrier (appearance) of the second order. When this piece of paper is placedin an envelope, the envelope becomes the carrier (appearance) of the third order.When this envelope is carried by a mailman, truck or plane, themailman, truck or plane become carriers (appearances) of the fourth order.

Additional evidence for the ideal nature of information gives the situation when different physical things carry (provide) the same information. Thus, physical things are not portions of information but contain a portion of information. For instance, different mails (letters) with the same text definitely contain the same information. The same phrase written on the paper and displayed on the screen of the computer in both cases contains the same information. Even different sentences can contain the same information. All aforementionedmails, letters, printed or written phrases and sentences are physical embodiments of information. Physical embodiment of information is similar to clothes worn by people. A person wearing different outfits nevertheless remains the same person.

It is interesting that while stating information is physical, Landauer at the same time writes about the physical representation of information in the following words: "Information is not an abstract entity but exists only through a physical representation . . . " [2].

Further evidence for the ideal nature of information are suggested by Lombardi, Holik and Vanni, who providepersuasive arguments that there is only one kind of information, physically neutral, which can be encoded by means of classical or of quantum states [16].

Relations between portions of ideal information in the World of Structures induce relations between material representations of these portions of information. However, there is no direct correspondence between these relations. For instance, if a portion $I$ of ideal information is a part of a portion $J$ of ideal information, then this does not necessarily entail that a material representation $MI$ of $I$ is part of a material representation $MJ$ of $J$.

Relations between material representations can also induce relations between the corresponding portions of ideal information. In this context, materialization of information has two meanings. First, materialization of information is the process of representing this information by a material object/system. Second, it is a material/physical representation of this information, that is, a result of the materialization process.

In the quantum world, materialization of information is performed by encoding it into quantum states [17].

There is also the process of information idealization, which goes in the opposite direction and is reciprocal but not always inverse to materialization of information. Both these processes are formally represented as named sets.

It is necessary to stress the difference between abstraction and idealization, between abstract and ideal objects.

*Abstraction* is built and lives (exists) in mentality, abolishing more and more properties of physical objects or mental objects on the lower level of abstraction.

*Idealization* reflects physical and mental objects in the realm of ideal structures.

Here are some examples:

An abstract straight line from the Euclidean geometry is only **imagined** and **described** by the Mind as perfectly straight and infinite in both directions.

An ideal straight line, which corresponds to the ideal image of the Euclidean geometry, **is** perfectly straight and infinite in both directions.

It is possible to build different models of the abstract straight line from the Euclidean geometry. To each of these models, an ideal straight line corresponds. Moreover, there is an ideal straight line, which unifies all these abstract straight lines.

## 5. Conclusions

Here, only the first steps in the study of information materialization and idealization are made. This process demands further theoretical study combined with experimental observations and analysis.

To conclude, we drawattention to the process of information mentalization. It represents a mirror image of information materialization. While information materialization is the physical embodiment of information, information mentalization is the mental personification of information.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Bawden, D.; Robinson, L. "Deep down things": In what ways is information physical, and why does it matter for information science? *Inf. Res.* **2013**, *18*, C03. Available online: http://InformationR.net/ir/18-3/colis/paperC03.html (accessed on 11 November 2021).
2. Landauer, R. Information is physical. *Phys. Today* **1991**, *44*, 23–29. [CrossRef]
3. Landauer, R. Information is a physical entity. *Phys. A* **1999**, *263*, 63–67. [CrossRef]
4. Landauer, R. Information is Inevitably Physical. In *Feynman and Computation: Exploring the Limits of Computers*; Westview Press: Oxford, UK, 2002; pp. 76–92.
5. Deutsch, D.; Marletto, C. Constructor theory of information. *Proc. R. Soc. A* **2015**, *471*, 20140540. [CrossRef] [PubMed]
6. Karnani, M.; Pääkkönen, K.; Annila, A. The physical character of information. *Proc. R. Soc. A* **2009**, *465*, 2155–2175. [CrossRef]
7. Davies, P.; Gregersen, N.H. *Information and the Nature of Reality: From Physics to Metaphysics*; Cambridge University Press: Cambridge, UK, 2010.
8. Vedral, V. *Decoding Reality: The Universe as Quantum Information*; Oxford University Press: Oxford, UK, 2010.
9. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
10. Klein, M. *The Psycho-Analysis of Children*; Hogarth: London, UK, 1932.
11. Solod, R.N.; Monte, C.F.; Wilson, J.P. *Beneath the Mask*; Wiley: Hoboken, NJ, USA, 2009.
12. Burgin, M. Ideas of Plato in the context of contemporary science and mathematics. *Athens J. Humanit. Arts* **2017**, *4*, 161–182.
13. Rosinger, E.E. Is Indeed Information Physical? HAL Preprint 2012, hal-00674036. Available online: https://hal.archives-ouvertes.fr/hal-00674036/file/IS_INFORMATION_PHYSICAL.pdf (accessed on 11 November 2021).
14. Burgin, M. Information in the Structure of the World. *Inf. Theor. Appl.* **2011**, *18*, 16–32.
15. Burgin, M.; Markov, K. A formal definition of materialization. In *Mathematics and Education in Mathematics*; Institute of Mathematics: Sofia, Bulgaria, 1991; pp. 175–179.
16. Lombardi, O.; Holik, F.; Vanni, L. What is quantum Information? *Stud. Hist. Philos. Mod. Phys.* **2016**, *56*, 17–26. [CrossRef]
17. Schumacher, B. Quantum coding. *Phys. Rev. A* **1995**, *51*, 2738–2747. [CrossRef] [PubMed]

information

MDPI

Article

# Is Information Physical and Does It Have Mass?

**Mark Burgin [1] and Rao Mikkilineni [2,*]**

[1] Department of Mathematics, University of California, 405 Hilgard Ave. Los Angeles, Los Angeles, CA 90095, USA
[2] Ageno School of Business, Golden Gate University, San Francisco, CA 94105, USA
* Correspondence: rmikkilineni@ggu.edu

**Abstract:** Some researchers suggest that information is a form of matter, calling it the fifth state of matter or the fifth element. Recent results from the general theory of information (GTI) contradict this. This paper aims to explain and prove that the claims of adherents of the physical nature of information are inaccurate due to the confusion between the definitions of information, the matter that represents information, and the matter that is a carrier of information. Our explanations and proofs are based on the GTI because it gives the most comprehensive definition of information, encompassing and clarifying many of the writings in the literature about information. GTI relates information, knowledge, matter, and energy, and unifies the theories of material and mental worlds using the world of structures. According to GTI, information is not physical by itself, although it can have physical and/or mental representations. Consequently, a bit of information does not have mass, but the physical structure that represents the bit indeed has mass. Moreover, the same bit can have multiple representations in the form of a physical substance (e.g., a symbol on a paper or a state of a flip-flop circuit, or an electrical voltage or current pulse.) Naturally, these different physical representations can have different masses, although the information is the same. Thus, our arguments are not against Landauer's principle or the empirical results of Vopson and other adherents of the physical nature of the information. These arguments are aimed at the clarification of the theoretical and empirical interpretations of these results. As the references in this paper show, recently many publications in which it is claimed that information is a physical essence appeared. That is why it is so important to elucidate the true nature of information and its relation to the physical world eliminating the existing misconceptions in information studies.

**Keywords:** information; physics; general theory of information; material carrier; material representation; knowledge; mass–energy–information–knowledge correspondence

**Citation:** Burgin, M.; Mikkilineni, R. Is Information Physical and Does It Have Mass? *Information* **2022**, *13*, 540. https://doi.org/10.3390/info13110540

Academic Editor: Lorenzo Mucchi

Received: 16 September 2022
Accepted: 7 November 2022
Published: 15 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Physical science is a branch of natural science that studies non-living systems, in contrast to life science, which studies living things. On the other hand, information science, according to the Merriam-Webster dictionary, is primarily concerned with the analysis, collection, classification, manipulation, storage, retrieval, movement, dissemination, and protection of information. However, while mathematicians, philosophers, biologists, physicists, and information scientists, to mention but a few, have all postulated various definitions of information since the notion of information emerged in human society, it is not an exaggeration to say that there is no consensus on what information really is.

Does information exist independently of our own existence? Does information processing require only living organisms, or also other material structures in the physical world to process information? Unlike humans, do the technical information-processing structures know that they are processing information? How is knowledge related to the information? While these are profound questions, the purpose of this paper is not to answer them. For answers, we refer the reader to the general theory of information (GTI) in [1–9] and in other related publications where these questions are studied and the answers are obtained. We use this theory in this paper because it is demonstrated that GTI gives the

most comprehensive definition of information, encompassing and clarifying what other researchers wrote about information.

In this paper, we investigate the mass-energy–information equivalence principle suggested in [10–14] and the related claims that information is physical, has mass, and is the fifth state of matter. "*For over 60 years, we have been trying unsuccessfully to detect, isolate or understand the mysterious dark matter," said Vopson. "If information indeed has mass*," he continued, "*a digital informational universe would contain a lot of it, and perhaps this missing dark matter could be information*" [10]. This statement is based on the mass-energy–information equivalence principle, which was suggested by Vopson and claims that information is transformed into mass or energy depending on its physical state. In addition, the existence of the intrinsic information underpinning the fundamental characteristics of elementary particles in the universe implies that stable, non-zero rest mass elementary particles store fixed and quantifiable information about themselves [10–22]. These so-called information conjectures also seem to imply that the information is a form of matter, which is called the fifth state of matter or the fifth element by Vopson [11–14].

It is necessary to remark that while the suggestion that information has mass is not accepted by many researchers, many of them, and the majority of lay people, think that information is physical by its nature. Thus, the main goal of our paper is to explain that this is not true and elucidate the true nature of the information.

To validate the assertions about the physical nature of information together with similar claims, we analyze the assumptions behind the formulated mass–energy–information equivalence principle using the GTI, and demonstrate that information is not physical by itself but has a physical representation. Naturally, this physical representation has mass and complies with physical laws.

In contrast to this, Landauer wrote:

> *"Information is inevitably inscribed in a physical medium. It is not an abstract entity. It can be denoted by a hole in a punched card, by the orientation of a nuclear spin, or by the pulses transmitted by a neuron. The quaint notion that information has an existence independent of its physical manifestation is still seriously advocated [23]. This concept, very likely, has its roots in the fact that we were aware of mental information long before we realized that it, too, utilized real physical degrees of freedom."* [17] p. 64

We argue that the physical properties that Landauer [15–18], Vopson [11–14], and other researchers ascribe to information [19–22] are actually the properties of the physical representations of information.

Note that while other researchers also repudiated the physical nature of information, nobody described the correct place of information in the world (cf., for example, [23–25]), while the general theory of information explains where information, in the strict sense, exists. It is important to understand the difference between information and its physical carrier because different physical carriers can contain the same information. Various observations support this statement.

Information per se belongs to the world of structures and does not have mass, but its representation (carrier) in the form of a physical structure possesses mass. In the physical world, genes, and neurons, for example, process information to convert it into knowledge. They communicate information, which is represented as biological and neurological structures, using chemical or electrical signals. In the digital world, a 'bit' of information does not exhibit mass, but a physical material that represents the bit indeed has mass. The same bit can have multiple representations in the form of physical material (e.g., a symbol on a paper or a state of a flip-flop circuit, or an electrical voltage or current pulse). Information is carried by the physical structures in the same way thermometers "carry" temperature.

Thus, the physical properties that Landauer and other researchers deduced, ascribing them to information [10–22], are actually the properties of the physical representation of information. This is in good agreement with what Landauer actually wrote, stating that "*information is inevitably tied to a physical representation*," and not with his more far-reaching claims such as "*information is a physical entity*" [17] p. 64.

It is necessary to remark that our arguments are not against Landauer's principle or the empirical results of Vopson. They are useful scientific results (cf., for example [26–28]). Our goal is the clarification of the theoretical and empirical interpretations of these results, explaining that they are about carriers of information but not about information itself. It is important to understand that the properties of information representations and information carriers are very important because people do not interact directly with information but work, for example, computing only with information representations and information carriers.

At the same time, it is important to know how to derive properties of information from properties of information representations and information carriers, and the general theory of information (GTI) provides efficient means for performing this. For instance, the whole area of cryptography studies how to find good information representations for secure transmission and preservation of information (cf., for example, [29,30]. Coding is a transformation of information representations and sometimes of information physical carriers. Programming is also a transformation of information representations. In this respect, our results complement the results of Landauer, Vopson, and other researchers who study the properties of physical information representations and information carriers.

The paper has the following structure. In Section 2, we present the ideas and conceptions from the GTI about information, its representation, and the relationship between information and knowledge. In Section 3, we discuss the mass-energy–information equivalence principle in light of the GTI. In Section 4, we put forward general observations from this study and conclusions.

## 2. General Theory of Information

The general theory of information (GTI) [1,3] states that "knowledge is related to information as the matter is related to energy". At the same time, the material structures in the physical world carry the information that represents the state and the dynamics of the structure under consideration. In the physical world, material structures are governed by the transformation laws of matter and energy. Energy has the potential to create or change material structures. All physical and chemical structures, which are created or changed by the transformation of matter and energy, obey the laws governing their transformations. All physical structures contain potential information that characterizes their structure, the functions of their constituent parts interacting with each other and with their surroundings, and their behaviors when internal and external factors cause fluctuations in their interactions. In fact, it means that there is a definite relationship between the characteristics of physical objects allowing the possibility of the conversion of mass into the energy of physical objects described by these characteristics. The famous formula $E = mc^2$ connects the energy and mass of physical objects. However, in contrast to what many people think, this formula does not mean that substance (matter) is equal to energy, but it shows the maximal amount of energy in a physical object with a given mass.

The states of physical structures and the regularities of their evolution are described by the laws of physics, which are mental structures created by humans (mainly by physicists and mathematicians). Living organisms have developed physical structures that exploit matter and energy transformations to acquire a unique identity and the ability to sense and process information that is carried by material structures and convert it into knowledge in the form of mental structures. While all living organisms have varying degrees of the ability to perceive, process, and convert information into knowledge, humans have developed the highest level of representing and managing mental structures using ideal structures in the form of named sets or fundamental triads [1]. The fundamental triad provides the schema and operations to create knowledge in the form of entities, their relationships, and their evolution consisting of event-driven behaviors [7–9]. Events are caused by fluctuations in the interactions among the components of the structures and their interaction with their environment. Thus, functions, structure, and fluctuations play important roles in the system's microscopic and macroscopic behaviors [31].

It is important to note that the mental models created by processing information are observer-dependent, as they depend on the previous knowledge of the observer in addition to many other idiosyncratic factors.

According to [1,3], the GTI places information per se in the ideal world of structures, which is the scientific manifestation of the world of Plato's Ideas or Forms [4]. Namely, the concept *structure* provides the scientific representation of Platonic Ideas, while the existence of the world of structures, which can be naturally equated to the world of Plato's Ideas, is proved by scientific means.

According to the ontological principle, O2 and its additional forms in the GTI ([1] (p. 99), [3]), information plays the same role in the world of structures as energy plays in the physical (material) world. While being associated with material structures in the physical world, the information does not belong to this world and can only be materialized in a physical form as asserted in the GTI [2]. Relations between information and structures were also considered by Stonier, who claimed that information has the power to exhibit itself as a structure when added to matter [32,33].

According to the ontological representability principle (ontological principle O4) of the GTI ([1] (p. 123), [3]), for any portion of the information I, there is always a representation Q of this portion of information for a system R. Often this representation is material, and as a result, since information is materially represented, many people comprehend information as physical. Consequently, a physical representation of information can be treated as the materialization of this information [2]. Thus, information not being physical by itself has a physical representation, and naturally, this physical representation complies with physical laws.

Moreover, according to the ontological embodiment principle (ontological principle O3) of the GTI ([1] (p. 120), [3]), for any portion I of information, there is always a carrier C of this portion of information for a system R. This carrier is, as a rule, material, and this makes information even more present in the physical world. The physical carrier of information can also be treated as the materialization of this information, or more precisely, the materialization on the second level. Materialization of information can require an agent or an observer to perform the process of materialization. An example is representing information in the form of symbols on the carrier, which is a piece of paper using a pen as a tool for materialization.

To show the difference between carriers and representations of information, we explain that any physical representation of information is also a physical carrier of the same information. A physical carrier of a portion I of information is any physical thing that contains this portion of information. At the same time, a physical representation of a portion I of information is such a physical carrier that allows direct extraction of this information. Thus, any physical representation is a physical carrier, but not any physical carrier is a physical representation. For instance, an envelope is the physical carrier of information contained in the letter this envelope encloses, the piece of paper on which the text of the letter is printed or written is also a physical carrier of the same information, and, finally, the text of the letter is also a physical carrier of the same information. However, direct extraction of information is possible only from the text. We cannot extract this information from the envelope or the piece of paper without the text. Consequently, the envelope that contains this letter or the paper on which this text is printed or written, as well as this piece of paper, are only carriers but not representations of the information in this letter.

The carrier of the information I that is not a representation of this information is called the enveloping carrier of I.

In the mental world created by living organisms, information received from the environment using the five senses enables mental representation and is converted into mental structures formed of fundamental triads [1]. There are two forms of mental structures —those that are derived from external observations and those that are created by the human mind representing the ideal structures. Mathematics is used to represent the ideal structures

and operations with them, as well as to model the systems from the material world, their states, and evolution.

Similarly, the mental reality (mental world) consists of various mental structures, which participate in the transformational processes involving information and knowledge. These transformational processes are defined by the physical information-processing structures, which consist of genes and neurons. The formula that is similar to Einstein's mass–energy equivalence also exists in the information realm of mentality.

To elaborate on this formula, it is necessary to explain that knowledge in the strict sense belongs to the world of structures, because knowledge consists of knowledge structures. At the same time, similar to information, knowledge has representations and carriers in the material (physical) and mental worlds. Various books and journals are physical carriers of knowledge, containing different knowledge representations. For instance, a formula, such as $E = Mc^2$, in the textbook in physics is a mathematical representation of knowledge about physical reality. Mental representations of knowledge exist in the mentality of the people and the mentality of groups of people, such as the community of physicists or mathematicians. However, many people call by the name knowledge what is really the mental representation of knowledge.

With this in mind, we introduce a new characteristic of mental knowledge named mental knowledge mass. Namely, the mental mass MK of a mental knowledge unit K is the measure of the knowledge object inertia concerning the structural movement in the mental world. Each mental knowledge mass reflects properties of the structural components of mental knowledge, their relationships, and behaviors. One mental knowledge structure interacts with other mental knowledge structures by sharing information using various means of communication facilitated by the information-processing physical systems such as genes and neurons, which use chemical and neuronal signals for communication.

Based on the concept of mental knowledge mass, we obtain the equivalence formula, which has the form $I = MK*p$, where $p > 0$ is the constant that connects the information I and mental knowledge K of mental systems just as energy and matter are connected in the physical world. This is a theoretical conjecture, which needs experimental validation. Finding the numerical value of the constant p could allow the estimation and measurement of information contained in mental knowledge systems.

With respect to mental mass, it is important to understand that mental knowledge has mental mass but not knowledge and information, which belong to the world of structures. Besides, energy, which is the physical counterpart of information, also does not have mass but only its measure is proportional to the mass of physical objects.

As the result, we arrive at the equivalence between the theory of physical structures and the theory of mental structures. Each such structure with a certain mass interacts with other structures based on various relationships defined by interaction potentials. In such a way, each structure provides guidelines for functional behavior and a network of structures provides guidelines for collective behavior based on interactions between structures. Wired together structural nodes of the network also fire together, shaping the collective behavior of the system. This allows us to represent the mental structures using the same mathematical representations of physical structures in the form of state vectors and their evolution.

In this context, a knowledge network is an assembly of components with specific functions, which interact as ideal structures and produce a stable behavior (equilibrium) when conditions are right. However, fluctuations change the interactions and cause non-equilibrium conditions. This leads to emergent behaviors leading to chaos. However, biological systems have developed an overlay of information-processing structures, which support and manage the system stability, safety, sustenance, etc., while monitoring the impact of environmental fluctuations.

## 3. GTI and the Mass–Energy–Information Equivalence Principle

Armed with this knowledge about information, we can now respond to the questions: is information physical and does it have mass? Answering the first question, we explain that information is associated with physical and mental structures, as its representations and carriers are embedded in other physical and mental structures that act as carriers of information. Answering the second question, we conjecture that the knowledge in mentality has a mental mass just as the matter has physical mass, while the information carriers (both physical and mental) have physical or mental mass but not the information itself.

These conclusions put us at odds with those researchers who claim information has mass [10–19]. For instance, Landauer claims that information is physical. However, at the beginning of his paper [17] p. 64, he writes

> *"Information is inevitably tied to a physical representation."*

It means that, according to Landauer, information is only tied to its physical representation but this tells nothing about the essence of information per se.

Another statement from his work is:

> *"Information is not a disembodied abstract entity; it is always tied to a physical representation" asserts what information is not telling anything of what information per se is"*. [16] p. 188

Similarly, Melvin Vopson claims

> *"A computational process creates digital information via some sort of physical process, which obeys physical laws, including thermodynamics."* [11]

As we explained before, this statement is misleading. The correct statement should be:

> *"A computational process creates digital information via some sort of physical process, which works with physical representations of digital information and obeys physical laws, including thermodynamics."*

Namely, only by changing physical representations, the physical process changes information [2]. In particular, erasing information changes the physical objects that were carriers of this information, while writing information transforms some physical objects into the carriers of the written information.

For instance, the Landauer principle states that logically irreversible computation can be only implemented by thermodynamically irreversible processes. In this setting, logical or abstract computation is performed with linguistic (symbolic) representations of information, while physical computation operates with physical representations and carriers of information [26].

Accordingly, the Formula (6) from [11] can be interpreted not as the mass of a bit of information, but as the mass of the physical representation of a bit of information.

Besides, there is a problem with the interpretation of Shannon's measure of information (information entropy) H. It measures information not directly but utilizes information's physical representations—signals or texts. When this measure is applied to the states of physical systems, it means that the state of a physical system is a representation of information while the corresponding system is the carrier of this information.

As the result, the mass-energy–information equivalence conjectured by Vopson in [11] is not valid because the same portion of information can have different physical representations. In other words, the mass and energy of the different representations of the same information can vary.

This situation is clearly explained by the general theory of information (GTI) mentioned above. Indeed, according to the ontological principle O4, for any portion of information I, there is always a representation Q of this portion of information for a system R [1,3]. Often this representation is material, and as a result, being materially represented, information becomes, in some sense, physical. In this context, a physical representation of information becomes the materialization of this information allowing people and other

systems to obtain this information [2]. For instance, the process of DNA replication shows that not only living beings but also unanimated systems such as molecules can transform and transmit information from one physical representation to another one.

Thus, information is not physical by itself but has a physical representation and, naturally, this physical representation complies with physical laws. This is in good agreement with what Landauer actually wrote in some of his works and not with his and his adherents' more far-reaching claims.

Similarly, some people can say that thoughts or feelings are physical because they are in the brain, which is physical. However, according to contemporary psychology, the brain is only the carrier of thoughts and feelings, the nature of which is essentially not physical. In particular, thinking is defined as "a mental process that involves the manipulation of information" [34].

One more argument that demonstrates that information is not physical is presented in [23]. Based on the conjecture of the physical nature of information, Kosso asserts that "information is transferred between states through interaction" because physical influences can be transferred only through interactions [35]. However, this assertion, for example, contradicts the so-called 'negative experiments' [36] where an object or event can be observed by noticing the absence of another object or event. It means that the observer obtains information without interaction with the object or event that contains this information [37].

Thus, the physical properties that Landauer, Vopson, and other researchers ascribe to information [10–19] are actually the properties of the physical representations of information.

## 4. Where Information Belongs

There are also other researchers who explain that information is not physical. For instance, Timpson justifies that the claim "information is physical" is essentially wrong because the term information "doesn't serve to refer to a material thing or substance" [38–40]. To make his approach complete, Timpson suggests that information exists in the form of "pieces of information, quantum or classical," while these pieces of information "are abstract types" and "they are not physical" [38]. This understanding is also supported in [23].

To understand the pitfall of this approach, we need to know what an abstract object or abstract type is. Philosophers elaborated a theory of abstract objects and abstract type (cf., for example, [41–43]). The main underpinning of this theory is the distinction between abstract and concrete, which did not play a noteworthy role in philosophy before the 20th century. However, in the 20th century, abstraction came to the forefront of mathematics and science. As a result, several philosophers tried to elaborate a clear and exact form of the notion of *abstract objects*, but mostly concluded that ordinary objects, such as trees and tables, are possibly concrete, while abstract objects, such as number 1 or straight lines, are not concrete [43].

Although the modern distinction between abstract and concrete objects bears some resemblance to Plato's differentiation of Ideas and Sensibles, this only conflates the concepts of ideal and abstract without any well-grounded reason. Being unable to find an explanation of the ideal reality of Plato, some philosophers decided to change the term *ideal* to the term *abstract* as contemporary science and mathematics went to higher and higher levels of abstraction, making the latter term more comprehensible.

In this context, the most reasonable approach to abstract objects is assuming that an abstract object consists of a name and a set of properties [44]. Based on this assumption, Edward Zalta built a formal axiomatic theory of abstract objects [43]. However, this theory does not answer the question about the place of abstract objects in the world.

Let us try to answer this important question. It is natural to suppose that as their name suggests, abstract objects are formed in the process of abstraction. This is an elaborate mental process that goes through various stages and achieves different levels of abstraction [45]. This situation implies that abstract objects as results of abstraction dwell in mentality. Some

of them belong only to individual mentality, while others also come to the group and social mentalities.

This correlates with the opinions of philosophers. For instance, Falguera, Martínez-Vidal, and Rosen write:

> *"The modern distinction* [between abstract and concrete, M.B. & R.M] *bears some resemblance to Plato's distinction between Forms and Sensibles. But Plato's Forms were supposed to be causes par excellence, whereas abstract objects are generally supposed to be causally inert."* [41]

Now we can analyze and answer the question of whether the information is an abstract object. We know definitely that there is information in mentality but the GTI also tells us that there is ontological information, which exists in nature being wrath encapsulated in physical systems independently of any mentality [46–48]. This persuasively shows that information is not an abstract object but, as the GTI demonstrates, it belongs to the world of ideal structures and comes to the physical and mental worlds through materialization and mentalization [2]. In particular, abstract objects are mental representations of information from the world of ideal structures.

Discussing abstract objects, it is important to understand that being names, with properties, which can be described by axioms in the formalized setting, abstract objects are special kinds of structures, and namely, they are external structures in the sense of the general theory of structures [5].

## 5. Conclusions

As it is possible to see from the discussion above, information is not physical by itself but has a physical representation and, naturally, this physical representation complies with physical laws. This is in good agreement with what Landauer actually wrote and not with his more far-reaching claims. Thus, the physical properties that Landauer and other researchers conjectured, ascribing them to information [10–19], are actually the properties of the physical representation of information.

The argument of Vopson that *"Archibald Wheeler . . . postulated that the universe emanates from the information inherent within it and he coined the phrase "It from bit"* [12] does not prove the physical nature of information because, according to the GTI, coming from the world of structures, information has a strong impact on the physical world [48].

Recently there were many publications in which it is claimed that information is a physical essence [10–19,49–53]. That is why it is so important to elucidate the true nature of information and its relation to the physical world eliminating the existing misconceptions in information studies.

In addition to this paper, the true nature of information and its relation to physical reality is also explained in [1–5] and related publications. It is possible to explain this only based on the GTI because there is no other theory of information in which it is proven that information, in the strict sense, belongs to the world of ideal structures.

It is important to emphasize the conclusions drawn from GTI. Information plays an important role in describing the material structures in the physical world, as well as the mental structures in the mental world created by biological systems through evolution and natural selection. Both material structures and mental structures are involved in receiving information, processing information, and communicating information. Information, in essence, describes the state of a structure and its evolution when events change it. The state of a material structure and its evolution are governed by the transformation laws of energy and matter. The information about a material structure can also be materialized and communicated using information carriers, which are also material structures. Communication of information using material structures, therefore, also obeys the transformation laws of matter and energy. While information per se has no mass, the materialized information (e.g., a symbol on a paper or a state of a flip-flop circuit, or an electrical voltage or current pulse) has mass. On the other hand, information received by the biological systems is processed and converted into knowledge in the form of mental structures. These mental

structures are materialized in the form of multi-layered networks of genes and neurons. Genes use sequences of symbols (DNA) and neurons use sub-symbolic computing to process information and use the knowledge to execute "life processes." These mental processes are distinguished by the self-organizing properties of biological systems in contrast to the dynamics of material structures subject to the laws of physics or behave the same way as complex adaptive systems that can exhibit emergence under fluctuations. GTI provides the tools to model both the material and mental structures and describe the conversion processes transforming information and knowledge.

To conclude our discussion, we remind the reader that mathematicians were able to understand the difference between numbers and their representations by numerals a long time ago. Hopefully, information scientists and other researchers will also be able to understand the difference between information and its physical representations. More importantly, they will be able to use the GTI to improve how we use information and knowledge, as well as to enhance our understanding of how nature operates and additionally design the digital world, which would imitate living organisms with such behaviors as autopoiesis and cognitive reasoning [7–9].

**Author Contributions:** Conceptualization, M.B. and R.M.; Formal analysis, M.B.; Investigation, R.M.; Writing—original draft, R.M.; Writing—review & editing, M.B. and R.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not Applicable.

**Informed Consent Statement:** Not Applicable.

**Data Availability Statement:** Not Applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. *Theory of Information: Fundamentality, Diversity, and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
2. Burgin, M. Physical Facets of Information, Information Materialization or the Grand Illusion. *Proceedings* **2022**, *81*, 76. [CrossRef]
3. Burgin, M. The General Theory of Information as a Unifying Factor for Information Studies: The Noble Eight-Fold Path. *Proceedings* **2017**, *1*, 164. [CrossRef]
4. Burgin, M. Ideas of Plato in the context of contemporary science and mathematics. *Athens J. Humanit. Arts* **2017**, *4*, 161–182. [CrossRef]
5. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
6. Burgin, M. Triadic Automata and Machines as Information Transformers. *Information* **2020**, *11*, 102. [CrossRef]
7. Burgin, M.; Mikkilineni, R.; Phalke, V. Autopoietic Computing Systems and Triadic Automata: The Theory and Practice. *Adv. Comput. Commun.* **2020**, *1*, 16–35. [CrossRef]
8. Mikkilineni, R. A New Class of Autopoietic and Cognitive Machines. *Information* **2022**, *13*, 24. [CrossRef]
9. Mikkilineni, R. Infusing Autopoietic and Cognitive Behaviors into Digital Automata to Improve Their Sentience, Resilience, and Intelligence. *Big Data Cogn. Comput.* **2022**, *6*, 7. [CrossRef]
10. Gasparini, A. Testable Theory Suggests Information Has Mass and could Account for Universe's Dark Matter. Scilight. 2019. Available online: https://aip.scitation.org/doi/10.1063/1.5126530 (accessed on 11 November 2022).
11. Vopson, M.M. The mass-energy-information equivalence principle. *AIP Adv.* **2019**, *9*, 095206. [CrossRef]
12. Vopson, M.M. The information catastrophe. *AIP Adv.* **2020**, *10*, 085014. [CrossRef]
13. Vopson, M.M. The information content of the universe and the implications for the missing dark matter. **2019**. [CrossRef]
14. Vopson, M.M. Estimation of the information contained in the visible matter of the universe. *AIP Adv.* **2021**, *11*, 105317. [CrossRef]
15. Landauer, R. Information is physical. *Phys. Today* **1991**, *44*, 23–29. [CrossRef]
16. Landauer, R. The physical nature of the information. *Phys. Lett. A* **1996**, *217*, 188–193. [CrossRef]
17. Landauer, R. Information is a physical entity. *Phys. AStat. Mech. Its Appl.* **1999**, *263*, 63–67. [CrossRef]
18. Landauer, R. Information is Inevitably Physical. In *Feynman and Computation: Exploring the Limits of Computers*; Westview Press: Oxford, UK, 2002; pp. 76–92.
19. Hong, J.; Lambson, B.; Dhuey, S.; Bokor, J. Experimental test of Landauer's principle in single-bit operations on nanomagnetic memory bits. *Sci. Adv.* **2016**, *2*, e1501492. [CrossRef] [PubMed]
20. Gaudenzi, R.; Burzurí, E.; Maegawa, S.; Van Der Zant, H.S.J.; Luis, F. Quantum Landauer erasure with a molecular nanomagnet. *Nat. Phys.* **2018**, *14*, 565–568. [CrossRef]

21. Bérut, A.; Arakelyan, A.; Petrosyan, A.; Ciliberto, S.; Dillenschneider, R.; Lutz, E. Experimental verification of Landauer's principle linking information and thermodynamics. *Nature* **2012**, *483*, 187–189. [CrossRef]
22. Jun, Y.; Gavrilov, M.; Bechhoefer, J. High-precision test of Landauer's principle in a feedback trap. *Phys. Rev. Lett.* **2014**, *113*, 190601. [CrossRef]
23. Penrose, R. *The Emperor's New Mind*; Oxford University Press: Oxford, UK, 1989.
24. Lombardi, O.; Holik, F.; Vanni, L. What is Shannon information? *Synthese* **2016**, *193*, 1983–2012. [CrossRef]
25. Rovelli, C. Relational quantum mechanics. *Int. J. Theor. Phys.* **1996**, *35*, 1637–1678.
26. Earman, J.; Norton, J.D. Exorcist XIV: The wrath of Maxwell's demon. Part II: From Szilard to Landauer and beyond. *Stud. Hist. Philos. Mod. Phys.* **1999**, *30*, 1–40. [CrossRef]
27. Ladyman, J. Physics and Computation: The Status of Landauer's Principle. In *Computation and Logic in the Real World. CiE 2007. Lecture Notes in Computer Science, vol 4497*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 446–454.
28. Ladyman, J.; Robertson, K. Landauer Defended: Reply to Norton. *Stud. Hist. Philos. Mod. Phys.* **2013**, *44*, 263–271. [CrossRef]
29. Mollin, R.A. *Codes: The Guide to Secrecy from Ancient to Modern Times*; Discrete Mathematics and Its Applications; Chapman & Hall/CRC: Boca Raton, FL, USA, 2005.
30. Katz, J.; Lindell, Y. *Introduction to Modern Cryptography*; CRC Press: Boca Raton, FL, USA, 2007.
31. Prigogine, I. Time, structure, and fluctuations. *Les Prix Nobel* **1979**, *132*, 263–285. [CrossRef] [PubMed]
32. Stonier, T. *Information and the Internal Structure of the Universe: An Exploration into Information Physics*; Springer: New York, NY, USA, 1990.
33. Stonier, T. Information as a basic property of the universe. *Biosystems* **1996**, *38*, 135–140. [CrossRef]
34. Davis, S.F.; Palladino, J.J. *Psychology*; Pearson: Harlow, UK, 2012.
35. Kosso, P. *Observability and Observation in Physical Science*; Kluwer: Dordrecht, The Netherlands, 1989.
36. Jammer, M. *The Philosophy of Quantum Mechanics*; Wiley: New York, NY, USA, 1974.
37. Lombardi, O. What is information? *Found. Sci.* **2004**, *9*, 105–134. [CrossRef]
38. Timpson, C. Quantum Information Theory and the Foundations of Quantum Mechanics. Ph.D. Thesis, University of Oxford, Oxford, UK, 2004.
39. Timpson, C. Philosophical aspects of quantum information theory. In *The Ashgate Companion to the New Philosophy of Physics*; Rickles, D., Ed.; Ashgate: Aldershot, UK, 2008; pp. 197–261.
40. Timpson, C. *Quantum Information Theory and the Foundations of Quantum Mechanics*; Oxford University Press: Oxford, UK, 2013.
41. Falguera, J.L.; Martínez-Vidal, C.; Rosen, G. Abstract Objects, The Stanford Encyclopedia of Philosophy. 2022. Available online: https://plato.stanford.edu/archives/sum2022/entries/abstract-objects/ (accessed on 30 January 2022).
42. Falguera, J.L.; Martínez-Vidal, C. (Eds.) *Abstract Objects: For and Against, Synthese Library, v. 422*; Springer: Cham, Switzerland, 2020.
43. Zalta, E.N. *Abstract Objects: An Introduction to Axiomatic Metaphysics*; D. Reidel: Dordrecht, The Netherlands, 1983.
44. Hale, B. *Abstract Objects*; Oxford University Press: Oxford, UK, 1987.
45. Burgin, M.; Diaz Nafria, J.M. Relational Structures of Conceptual Spaces. *Proceedings* **2022**, *81*, 39. [CrossRef]
46. Krzanowski, R. Ontological Information. Investigation into the Properties of Ontological Information. Ph.D. Thesis, UPJP2, Kraków, Poland, 2020. Available online: http://bc.upjp2.edu.pl/dlibra/docmetadata?id=5024&from=&dirids=1&ver_id=&lp=2&QI= (accessed on 15 March 2022).
47. Burgin, M.; Krzanowski, R. World Structuration and Ontological Information. *Proceedings* **2022**, *81*, 93. [CrossRef]
48. Krzanowski, R. *Ontological Information*; World Scientific: New York, NY, USA; London, UK; Singapore, 2020.
49. Bormashenko, E. The Landauer Principle: Re-Formulation of the Second Thermodynamics Law or a Step to Great Unification? *Entropy* **2019**, *21*, 918. [CrossRef]
50. Bormashenko, E. Informational Reinterpretation of the Mechanics Notions and Laws. *Entropy* **2020**, *22*, 631. [CrossRef]
51. Džaferović-Mašić, E. Missing information in the Universe as a dark matter candidate based on the mass-energy-information equivalence principle. *J. Phys. Conf. Ser.* **2021**, *1814*, 012006. [CrossRef]
52. Herrera, L. Landauer Principle and General Relativity. *Entropy* **2020**, *22*, 340. [CrossRef] [PubMed]
53. Vopson, M.M. Experimental protocol for testing the mass–energy–information equivalence principle. *AIP Adv.* **2022**, *12*, 035311. [CrossRef]

*Entropy* **2012**, *14*, 2066-2080; doi:10.3390/e14112066

OPEN ACCESS

*entropy*

**ISSN 1099-4300**

www.mdpi.com/journal/entropy

*Article*

# Unconventional Algorithms: Complementarity of Axiomatics and Construction

**Gordana Dodig Crnkovic [1,*] and Mark Burgin [2]**

1 Department of Computer Science and Networks, School of Innovation, Design and Engineering Mälardalen University, Västerås, 72123, Sweden

2 Department of Mathematics, UCLA, Los Angeles, CA 90095, USA; E-Mail: mburgin@math.ucla.edu (M.B.)

* Author to whom correspondence should be addressed; E-Mail: gordana.dodig-crnkovic@mdh.se; Tel.: +46 21 15 17 25.

*Received: 20 August 2012; in revised form: 23 September 2012; / Accepted: 19 October 2012 / Published: 25 October 2012*

---

**Abstract:** In this paper, we analyze axiomatic and constructive issues of unconventional computations from a methodological and philosophical point of view. We explain how the new models of algorithms and unconventional computations change the algorithmic universe, making it open and allowing increased flexibility and expressive power that augment creativity. At the same time, the greater power of new types of algorithms also results in the greater complexity of the algorithmic universe, transforming it into the algorithmic multiverse and demanding new tools for its study. That is why we analyze new powerful tools brought forth by local mathematics, local logics, logical varieties and the axiomatic theory of algorithms, automata and computation. We demonstrate how these new tools allow efficient navigation in the algorithmic multiverse. Further work includes study of natural computation by unconventional algorithms and constructive approaches.

**Keywords:** unconventional computing; computation beyond the Turing limit; axiomatic *vs.* constructive models; unconventional models of computation

---

## 1. Introduction

The development of computer science and information technology brought forth a diversity of novel algorithms and algorithmic schemas, unconventional computations and nature-inspired

processes, advanced functionality and conceptualizations. The field of information processing and information sciences encountered the stage of transition from classical subrecursive and recursive algorithms, like finite automata, recursive functions or Turing machines, to new super-recursive algorithms, such as inductive Turing machines or limiting recursive functions.

Super-recursive algorithms controlling and directing unconventional computations exceed the boundary set by Turing machines and other recursive algorithms, resulting in an open algorithmic universe and revealing new levels of creativity. As the growth of possibilities involves a much higher complexity of the new, open world of super-recursive algorithms, unconventional computations, innovative hardware and advanced organization, we discuss means of navigation in this open algorithmic world.

The paper is organized as follows: in Section 2, we compare characteristics of local and global mathematics, explaining why local mathematics allows better modelling of reality. Section 3 addresses relationships between local mathematics, local logics and logical varieties, while Section 4 offers the discussion of projective mathematics versus reverse mathematics versus classical mathematics. Section 5 answers the question of how to navigate in the algorithmic multiverse. Finally, Section 6 presents our conclusions and provides directions for future work.

## 2. Local Mathematics *vs.* Global Mathematics

As an advanced knowledge system, mathematics exists as an aggregate of various mathematical fields. If at the beginning there were only two fields—arithmetics and geometry, now there are hundreds of mathematical fields and subfields. However, mathematicians always believed in mathematics as a unified system striving to build common and in some senses absolute foundations for all mathematical fields and subfields. At the end of the 19th century, mathematicians came very close to achieving this goal as the emerging set theory allowed the construction of all mathematical structures using only sets and operations with sets. However, in the 20th century, it was discovered that there are different set theories. This brought some confusion and attempts to find the "true" set theory.

To overcome this confusion, Bell [1] introduced in 1986 the concept of local mathematics. The fundamental idea was to abandon the unique, absolute universe of sets central to the orthodox set-theoretic account of the foundations of mathematics, replacing it by a plurality of local mathematical frameworks. Bell suggested taking elementary topoi as such frameworks, which would serve as local replacements for the classical universe of sets. Having sufficient means for developing logic and mathematics, elementary topoi possess a sufficiently rich internal structure to enable a variety of mathematical concepts and assertions to be interpreted and manipulated. Mathematics interpreted in any such framework is called *local mathematics* and admissible transformation between frameworks amounts to a (definable) *change of local mathematics.* With the abandonment of the absolute universe of sets, mathematical concepts in general lose absolute meaning, while mathematical assertions liberate themselves from absolute truth values. Instead they possess such meanings or truth values only *locally*, *i.e.*, *relative* to local frameworks. This means that the *reference* of any mathematical concept is accordingly not fixed, but *changes* with the choice of local mathematics.

It is possible to extend the approach of Bell in three directions. First, we can use an arbitrary category as a framework for developing mathematics. When an internal structure of such a framework

is meager, the corresponding mathematics will also be indigent. Second, it is possible to take a theory of some structures instead of the classical universe of sets and develop mathematics within that framework without reference to the universal framework. Third, as we know, there are different axiomatizations of set theory. Developed axiomatics are often incompatible, e.g., axiomatics in which the Continuum Hypothesis is true and axiomatics where it is false. Thus, developing mathematics based on one such axiomatics also results in a local mathematics.

A similar situation emerged in computer science, where mathematics plays a pivotal role. Usually, to study properties of computers, computer networks and computational processes and elaborate more efficient applications, mathematicians and computer scientists use mathematical models. There is a variety of such models: Turing machines of different kinds (with one tape and one head, with several tapes, with several heads, with $n$-dimensional tapes, non-deterministic, probabilistic, and alternating Turing machines, Turing machines that take advice and Turing machines with oracle, *etc.* [2]), Post productions, partial recursive functions, neural networks, finite automata of different kinds (automata without memory, autonomous automata, accepting automata, probabilistic automata, *etc.*), Minsky machines [3], normal Markov algorithms [4], Kolmogorov algorithms [5], formal grammars of different kinds (regular, context free, context sensitive, phrase-structure, *etc.*), Storage Modification Machines or simply Shönhage machines [6], Random Access Machines (RAM) [7], Petri nets [8], which like Turing machines have several forms (ordinary, regular, free, colored, self-modifying, *etc.*), and so on. All these models are constructive, *i.e.*, they have tractable explicit descriptions and simple rules for operation. Thus, the constructive approach dominates in computer science.

This diversity of models is natural and useful because each type is suited to a particular type of problem. In other words, the diversity of problems that are solved by computers gives rise to a corresponding diversity of models. For example, general problems of computability involve such models as Turing machines and partial recursive functions. Finite automata are used for text search, lexical analysis, and construction of semantics for programming languages. In addition, different computing devices demand corresponding mathematical models. For example, universal Turing machines and inductive Turing machines allow one to investigate characteristics of conventional computers [2]. Petri nets are useful for modeling and analysis of computer networks, distributed computation, and communication processes [9]. Finite automata model computer arithmetic. Neural networks reflect properties of the brain. Abstract vector and array machines model vector and array computers [2].

To utilize some models that are related to a specific type of problem, we need to know their properties. In many cases, different classes of models have the same or similar properties. As a rule, such properties are proved for each class separately. Thus, alike proofs are repeated many times in similar situations involving various models and classes of algorithms.

In contrast to this, the *projective* (also called *multiglobal*) *axiomatic theory* of algorithms, automata and computation suggests a different approach [10]. Assuming some simple basic conditions (in the form of postulates, axioms and conditions), many profound and far-reaching properties of algorithms are derived in this theory. This allows one, when dealing with a specific model, not to prove this property, but only to check the conditions from the assumption, which is much easier than to prove the property under consideration. In such a way, we can derive various characteristics of types of computers and software systems from the initial postulates, axioms and conditions.

The projective approach in computer science has its counterpart in mathematics, where systems of unifying properties have been used for building new encompassing structures, proving indispensable properties in these new structures and projecting these properties on the encompassed domains. Such projectivity has been explicitly utilized in category theory, which was developed and utilized with the goal of unification [11].

Breaking the barrier of the Church-Turing Thesis drastically increased the variety of algorithmic model classes and changed the algorithmic universe of recursive algorithms to the multiverse of super-recursive algorithms [2], which consists of a plurality of local algorithmic universes. Each class of algorithmic model forms a local algorithmic universe, providing means for the development of local computer science in general and a local theory of algorithms in particular.

Local mathematics brings forth local logics because each local mathematical framework has its own logic and it is possible that different frameworks have different local logics.

## 3. Logical Varieties as a Unification of Local Logics

Barwise and Seligman [12] developed a theory of information flow reflecting the dynamics of information processing systems. In this theory, the concept of a *local logic* plays a fundamental role in the modeling of commonsense reasoning, which is an important kind of information processing.

The basic concept of the information flow theory is a *classification*. A natural interpretation of a classification, which is a typical named set [13], is a representation of some domain in the physical or abstract world by a system of symbols, which denote types of objects from the represented domain. Each local logic corresponds to a definite classification describing properties of the domain and the classification in a logical language and allowing one to deduce previously unknown properties. This implies a natural condition that each domain has its own local logic and different domains may have different local logics.

In a similar way, each class of algorithms from the algorithmic multiverse, as well as a constellation of such classes, forms a local algorithmic universe, which has a corresponding local logic. These logics may be essentially different. For instance, taking two local algorithmic universes formed by such classes as the class ***T*** of all Turing machines and the class ***TT*** of all total, *i.e.*, everywhere defined, Turing machines, we find that the first class satisfies the axiom of universality [10], which affirms existence of a universal algorithm, *i.e.*, a universal Turing machine in this class. However, the class ***TT*** does not satisfy this axiom [10].

Barwise and Seligman [12] assumed that the totality of local logics forms a set. However, analyzing the system of local logics, it is possible to see that there are different relations between them and it would be useful to combine these logics in a common structure. As is explained in [13], local logics form a deductive logical variety or a deductive logical prevariety, which were introduced and studied in [14] as a tool to work with inconsistent systems of knowledge. Logical varieties and prevarieties provide a unified system of logical structures, in which local logics are naturally integrated.

Minsky [15] was one of the first AI researchers who brought attention to the problem of inconsistent knowledge. He wrote that consistency is a delicate concept that assumes the absence of contradictions in systems of axioms. Minsky also suggested that in artificial intelligence (AI) systems this assumption was superfluous because there were no completely consistent AI systems. In his

opinion, it is important to understand how people solve paradoxes, find a way out of a critical situation, and learn from their own or others' mistakes or how they recognize and exclude different inconsistencies. In addition, Minsky [16] suggested that consistency and effectiveness may well be incompatible. He further writes [17]: "An entire generation of logical philosophers has thus wrongly tried to force their theories of mind to fit the rigid frames of formal logic. In doing that, they cut themselves off from the powerful new discoveries of computer science. Yes, it is true that we can describe the operation of a computer's hardware in terms of simple logical expressions. But no, we cannot use the same expressions to describe the meanings of that computer's output -- because that would require us to formalize those descriptions inside the same logical system. And this, I claim, is something we cannot do without violating that assumption of consistency." Minsky [17] continues, "In summary, there is no basis for assuming that humans are consistent - nor is there any basic obstacle to making machines use inconsistent forms of reasoning". Moreover, it has been discovered that not only human knowledge but also representations/models of human knowledge (e.g., large knowledge bases) are inherently inconsistent [18]. Logical varieties or prevarieties provide powerful tools for working with inconsistent knowledge.

There are different types and kinds of logical varieties and prevarieties: *deductive* or *syntactic varieties* and *prevarieties*, *functional* or *semantic varieties* and *prevarieties* and *model* or *pragmatic varieties* and *prevarieties*. Syntactic varieties, prevarieties, and quasi-varieties (which were introduced in [19]) are built from logical calculi as building blocks. Semantic varieties and prevarieties (which were introduced and studied in [20]) are built from logics, while model varieties and prevarieties (also introduced and studied in [20]) are built from separate logical models.

Let us consider a logical language $L$, an inference language $R$, a class **K** of syntactic logical calculi, a set $Q$ of inference rules ($Q \subseteq R$), and a class **F** of partial mappings from $L$ to $L$.

A triad $\mathbf{M} = (A, H, M)$, where $A$ and $M$ are sets of expressions that belong to $L$ ($A$ consists of axioms and $M$ consists of theorems) and $H$ is a set of inference rules, which belong to the set $R$, is called:

(1) a *projective syntactic* (**K,F**)-*prevariety* if there exists a set of logical calculi $\boldsymbol{C}_i = (A_i , H_i , T_i )$ from **K** and a system of mappings $f_i$: $A_i \to L$ and $g_i$: $M_i \to L$ ($i \in I$) from **F** in which $A_i$ consists of all axioms and $M_i$ consists of all theorems of the logical calculus $\boldsymbol{C}_i$, and for which the equalities $A = \cup_{i\in I} f_i(A_i)$, $H = \bigcup_{i\in I} H_i$ and $M = \bigcup_{i\in I} g_i(M_i)$ are valid (it is possible that $\boldsymbol{C}_i = \boldsymbol{C}_j$ for some $i \neq j$).

(2) a *projective syntactic* (**K,F**)-*variety* with the depth $k$ if it is a projective syntactic (**K,F**)-quasi-prevariety and for any $i_1 , i_2 , i_3 , \dots , i_k \in I$ either the intersections $\bigcap_{j=1}^{k} f_{ij}(A_{ij})$ and $\bigcap_{j=1}^{k} g_{ij}(T_{ij})$ are empty or there exists a calculus $\boldsymbol{C} = (A, H, T)$ from **K** and projections $f$: $A \to \bigcap_{j=1}^{k} f_{ij}(A_{ij})$ and $g$: $N \to \bigcap_{j=1}^{k} g_{ij}(M_{ij})$ from **F** where $N \subseteq T$;

(3) a *syntactic* **K**-prevariety if it is a projective syntactic (**K,F**)-prevariety in which $M_i = T_i$ for all $i \in I$ and all mappings $f_i$ and $g_i$ that define **M** are bijections on the sets $A_i$ and $M_i$, correspondingly;

(4) a *syntactic* **K**-*variety* if it is a projective syntactic (**K,F**)- variety in which $M_i = T_i$ for all $i \in I$ and all mappings $f_i$ and $g_i$ that define **M** are bijections on the sets $A_i$ and $M_i$, correspondingly.

The calculi $\boldsymbol{C}_i$ used in the formation of the prevariety (variety) **M** are called *components* of **M**.

Note that different components of deductive logical varieties and prevarieties can not only contain distinct axioms and theorems but also employ distinctive deduction rules. For instance, one component can use classical deduction, while another component of the same variety can be based on a relevant logic.

We see that the collection of mappings $f_i$ and $g_i$ makes a unified system called a prevariety or quasi-prevariety out of separate logical calculi $\boldsymbol{C}_i$, while the collection of the intersections $\bigcap_{j=1}^{k} f_{ij}(A_{ij})$ and $\bigcap_{j=1}^{k} g_{ij}(T_{ij})$ makes a unified system called a variety out of separate logical calculi $\boldsymbol{C}_i$. For instance, mappings $f_i$ and $g_i$ allow one to establish a correspondence between norms/laws that were used in one country during different periods of time or between norms/laws used in different countries. In a similar way, relations between components of logical varieties and prevarieties allow one to establish a correspondence between properties of different models of computation and algorithmic classes.

The main goal of syntactic logical varieties is in presenting sets of formulas as a structured logical system using logical calculi, which have means for inference and other logical operations. Semantically, it allows one to describe a domain of interest, e.g., a database, knowledge of an individual or the text of a novel, by a syntactic logical variety dividing the domain in parts that allow representation by calculi.

In comparison with varieties and prevarieties, logical quasi-varieties and quasi-prevarieties studied in [20] are not necessarily closed under logical inference. This trait allows better flexibility in knowledge representation.

While syntactic logical varieties and prevarietis synthesize local logics in a unified system, semantic logical varieties and prevarieties studied in [20] unify local mathematics forming a holistic realm of mathematical knowledge. Local meaning of mathematical concepts is defined by model logical varieties and prevarieties and is relative with respect to each component of the corresponding variety or prevariety. In the context of local mathematics, mathematical assertions liberate themselves from absolute truth values acquiring relative truth values specifically defined by each component of the corresponding semantic logical variety or prevariety.

In addition, syntactic logical varieties and prevarieties found diverse applications to databases and network technology providing tools for working with inconsistency, imprecision, vagueness, non-monotonic inference, knowledge base unification and database integration (cf., for example, [21]).

## 4. Projective Mathematics *vs.* Reverse Mathematics *vs.* Classical Mathematics

According to Suppe [22, p. 9], "axiomatization is a formal method for specifying the content of a theory wherein a set of axioms is given from which the remaining content of the theory can be derived deductively as theorems. The theory is identified with the set of axioms and its deductive consequences, which is known as the closure of the axiom set. The logic used to deduce theorems may be informal, as in the typical axiomatic presentation of Euclidean geometry; semiformal, as in reference to set theory or specified branches of mathematics; or formal, as when the axiomatization consists in augmenting the logical axioms for first-order predicate calculus by the proper axioms of the theory."

Mathematics suggests an approach for knowledge unification, namely, it is necessary to find axioms that characterize all theories in a specific area and to develop the theory in an axiomatic context. This approach has worked extremely well in a variety of mathematical fields, providing rigorous tools for mathematical exploration.

Axiomatization has often been used in physics (Hilbert's sixth problem refers to axiomatization of branches of physics in which mathematics is prevalent and researchers found that, for example, finding

the proper axioms for quantum field theory is still an open and difficult problem in mathematics), biology (according to Britannica, [23] the most enthusiastic proponent of this approach, the British biologist and logician Joseph Woodger, attempted to formalize the principles of biology—to derive them by deduction from a limited number of basic axioms and primitive terms—using the logical apparatus of the *Principia Mathematica* by Whitehead and Bertrand Russell), and some other areas, such as philosophy or technology. It is interesting that the axiomatic approach was also used in areas that are very far from mathematics. For instance, Spinoza used this approach in philosophy, developing his ethical theories and writing his book *Ethics* in the axiomatic form. More recently, Kunii [24] developed an axiomatic system for cyberworlds.

Since the advent of computers, deductive reasoning and axiomatic exposition have been delegated to computers, which performed theorem-proving, while the axiomatic approach has come to software technology and computer science. Logical tools and axiomatic description have been used in computer science for different purposes. For instance, Manna [25] built an axiomatic theory of programs, while Milner [26] developed an axiomatic theory of communicating processes. An axiomatic description of programming languages was constructed by Meyer and Halpern [27]. Many researchers have developed different kinds of axiomatic recursion theories (cf., for example [28–33]).

However, in classical mathematics, axiomatization has a global character. Mathematicians tried to build a unique axiomatics for the foundations of mathematics. Logicians working in the theory of algorithms tried to find axioms comprising all models of algorithms.

This is the classical approach – axiomatizing the studied domain and then deducing theorems from axioms. All classical mathematics is based on deduction as a method of logical reasoning and inference. *Deduction* is a type of reasoning process that constructs and/or evaluates *deductive arguments*, where the conclusion follows from the premises with logical necessity. In logic, an argument is called deductive when the truth of the conclusion is purported to follow necessarily or be a logical consequence of the assumptions. Deductive arguments are said to be valid or invalid, but never true or false. A deductive argument is valid if and only if the truth of the conclusion actually does follow necessarily from the assumptions. A valid deductive argument with true assumptions is called sound. A deductive argument which is invalid or has one or more false assumptions or both is called unsound. Thus, we may call classical mathematics by the name *deductive mathematics*.

The goal of deductive mathematics is to deduce theorems from axioms. Deduction of a theorem is also called proving the theorem. When mathematicians cannot prove some interesting and/or important conjecture, researchers with a conventional thinking try to prove that the problem is unsolvable in the existing framework. Creative explorers instead invent new structures and methods, construct new framework, introducing new axioms to solve the problem.

Some consider deductive mathematics as a part of axiomatic mathematics, assuming that deduction (in a strict sense) is possible only in an axiomatic system. Others treat axiomatic mathematics as a part of deductive mathematics, assuming that there are other inference rules besides deduction.

While deductive mathematics is present in and actually dominates all fields of contemporary mathematics, reverse mathematics is the branch of mathematical logic that seeks to determine what are the minimal axioms (formalized conditions) needed to prove a particular theorem [34,35]. This direction in mathematical logic was founded by [28,36]. The method can briefly be described as going backwards from theorems to the axioms necessary to prove these theorems in some logical system [37]. It

turns out that over a weak base theory, many mathematical statements are equivalent to the particular new postulate needed to prove them. This methodology contrasts with the ordinary mathematical practice where theorems are deduced from *a priori* assumed axioms.

Reverse mathematics was prefigured by some results in set theory, such as the classical theorem that states that the axiom of choice, well-ordering principle of Zermelo, maximal chain priciple of Hausdorff, Zorn's lemma [38], and statements of the vector basis theorem [39] and Tychonov product theorem [40] are equivalent over ZF set theory (Howard and Rubin, 1998) [41]. The goal of reverse mathematics, however, is to study ordinary theorems of mathematics rather than possible axioms for set theory. A sufficiently weak base theory is adopted (usually, it is a subsystem of second-order arithmetic) and the search is for minimal additional axioms for deducing some interesting/important mathematical statements. It has been found that in many cases these minimal additional axioms are equivalent to the particular statements they are used to prove.

Projective mathematics is a branch of mathematics similar to reverse mathematics, which aims to determine what simple conditions are needed to prove the particular theorem or to develop a particular theory. However, there are essential differences between these two directions: reverse mathematics is aimed at a logical analysis of mathematical statements, while projective mathematics is directed at making the scope of theoretical statements in general and mathematical statements in particular much larger whilst extending their applications. As a result, instead of proving similar results in various situations, it becomes possible to prove a corresponding general result in the axiomatic setting and to ascertain validity of this result for a particular case by demonstrating that all axioms (conditions) used in the proof are true for this case. In this way the general result is projected on different situations. This direction in mathematics was founded by Burgin [10]. This approach contrasts with conventional (deductive) mathematics where axioms describe some area or type of mathematical structures, while theorems are deduced from a priori assumed axioms.

Projective mathematics has its precursor in such results as the extension of many theorems initially proved for numerical functions to functions in metric spaces [42], or generalizations of properties of number systems to properties of groups, rings and other algebraic structures [39].

Here we describe how projective mathematics is used for exploration of computations controlled by algorithms and realized by automata. In this application of projective mathematics, the goal is to find some simple properties of computations, algorithms and automata in general, to present these properties in the form of axioms, and to deduce from these axioms theorems that describe much more profound and sophisticated properties of computations, algorithms and automata. This allows one, taking some class **A** of algorithms, not to prove these theorems but only to check if the initial axioms are valid in **A**. If this is the case, then it becomes possible to conclude that all corresponding theorems are true for the class **A**. As we know, computer scientists and mathematicians study and utilize a huge variety of different classes and types of algorithms, automata, and abstract machines. Consequently, such an axiomatic approach allows them to obtain many properties of studied algorithms and automata in a simple and easy way.

It is possible to explain goals of classical (deductive) mathematics, reverse mathematics and projective mathematics by means of relations between axioms and theorems.

A set $A$ of axioms can be:

(1) *Consistent* with some result (theorem) $T$, *i.e.*, when the theorem $T$ is added as a new axiom, the new system remains consistent, allowing one to, in some cases, deduce (prove) this theorem.

(2) *Sufficient* for some result (theorem) $T$, *i.e.*, it is possible to deduce (prove) the theorem $T$ using axioms from $A$.

(3) *Irreducible* with respect to some result (theorem) $T$, *i.e.*, the system $A$ is a minimal set of the axiom that allows one to deduce (prove) the theorem $T$.

After the discovery of non-Euclidean geometries, the creation of modern algebra and the construction of set theory, classical mathematics' main interest has been to find whether a statement $T$ has been consistent with a given axiomatic system $A$ (the logical goal) and then to prove this statement in the context of $A$. Thus, classical mathematics is concerned with the first relation. Reverse mathematics, as we can see, deals with the third relation.

In contrast to this, projective mathematics is oriented at the second relation. The goal is to find some simple properties of algorithms or automata in general, to present these properties in the form of a system $U$ of axioms, and from these axioms, to deduce theorems that describe much more profound properties of algorithms and automata. This allows one, taking some class **A** of algorithms or automata, not to prove these theorems but only to check if all axioms from the system $U$ are valid in **A**. If this is the case, then it is possible to conclude that all corresponding theorems are true for the class **A**. As we know, computer scientists and mathematicians study and utilize a huge variety of different classes and types of algorithms, automata, and abstract machines. In such a way, the axiom system $U$ provides a definite perspective on different classes and types of algorithms, automata, and abstract machines.

It is interesting that Bernays had a similar intuition with respect to axioms in mathematics, regarding them not as a system of statements about a subject matter but as a system of conditions for what might be called a relational structure. He wrote in [43]: "A main feature of Hilbert's axiomatization of geometry is that the axiomatic method is presented and practiced in the spirit of the abstract conception of mathematics that arose at the end of the nineteenth century and which has generally been adopted in modern mathematics. It consists in abstracting from the intuitive meaning of the terms... and in understanding the assertions (theorems) of the axiomatized theory in a hypothetical sense, that is, as holding true for any interpretation... for which the axioms are satisfied. Thus, an axiom system is regarded not as a system of statements about a subject matter but as a system of conditions for what might be called a relational structure... [On] this conception of axiomatics, ... logical reasoning on the basis of the axioms is used not merely as a means of assisting intuition in the study of spatial figures; rather, logical dependencies are considered for their own sake, and it is insisted that in reasoning we should rely only on those properties of a figure that either are explicitly assumed or follow logically from the assumptions and axioms."

It is possible to formalize the approach of projective mathematics using logical varieties. Indeed, let us take a collection $C$ of postulates, axioms and conditions, which are formalized in a logical language as axioms. This allows us to assume that we have a logical variety $M$ that represents a given domain $D$ in a formal mathematical setting and contains the set $C$. For instance, the domain $D$ consists of a system of algorithmic models so that the logic of each model $D_i$ is a component $M_i$ of $M$. Then we

deduce a theorem $T$ from the statements from $C$. Then instead of proving the theorem $T$ for each domain $D_i$, we check whether $C \subseteq M_i$. When this is true, we conclude that the theorem $T$ belongs to the component $M_i$ because $M_i$ is a calculus and thus, the theorem $T$ is valid for the model $D_i$. Because $C$ usually consists of relatively simple statements, to check the inclusion $C \subseteq M_i$ is simpler than to prove $T$ in $M_i$. In addition, this approach provides unification for the whole theory of algorithms, automata and computation as it explicates similarities and common traits in different algorithmic models and abstract automata.

## 5. How To Navigate in the Algorithmic Multiverse

It is possible to see that for a conformist, it is much easier to live in the closed algorithmic universe because all possible and impossible actions, as well as all solvable and insolvable problems can be measured against one of the most powerful and universal classes of algorithms in the algorithmic universe. This has usually been done utilizing Turing machines.

The open world provides many more opportunities for actions and problem solving, but at the same time it demands more work, more effort and even more imagination for solving problems which are insolvable in the closed algorithmic universe. Even the closed algorithmic universe contains many classes and types of algorithms, which have been studied with reference to a universal class of recursive algorithms. In some cases, partial recursive functions have been used. In other cases, unrestricted grammars have been employed. The most popular have been the utilization of Turing machines. A big diversity of new and old classes of algorithms exists that demands specific tools for exploration.

Mathematics has invented such tools and one of the most efficient for dealing with diversity is the axiomatic method. This method was also applied to the theory of algorithms, automata and computation when the axiomatic theory of algorithms, automata and computation was created [10]. In it, many profound properties of algorithms are derived based on some simple, basic conditions (in the form of postulates, axioms and conditions). Namely, instead of proving similar results in various situations, it becomes possible to prove a necessary general result in the axiomatic setting and then to ascertain the validity of this result for a particular case by demonstrating that all axioms (conditions) used in the proof are true for this case. Note that in contrast to 20th century mathematics, where projectivity was based on unifying constructions in a form of new mathematical structures [11], such as categories or heterogeneous algebras, projectivity developed in [10] in the context of computer science extracts only unifying properties without building new structures. In such a way, the general result is projected on different situations. For instance, let us consider some basic algorithmic problems inherent in computer and network functioning. One of these problems is the Fixed Output Problem. In this problem, it is necessary to find an algorithm/automaton $H$ that for an arbitrary algorithm/automaton $A$ from a given class **K** and arbitrary data elements $b$ and $x$ informs whether application of $A$ to $x$ gives $b$ as the result, *i.e.*, whether $A(x) = b$.

In [4], the theorem on undecidability of the Fixed Output Problem is proved based on the projective approach. As a result, this theorem has more than 30 corollaries for various classes of algorithms (computational models), including the famous theorem about the undecidability of the halting problem for Turing machines. Another theorem on the recognizability of the Fixed Output Problem proved in [10]

has more than 20 corollaries for various classes of algorithms (computational models), such as Turing machines, random access machines, Kolmogorov algorithms, Minsky machines, partial recursive functions, inductive Turing machines of the first order, periodic evolutionary Turing machines and limiting partial recursive functions. Note that such algorithmic problems were previously studied separately for each computational model.

The axiomatic context allows a researcher to explore not only individual algorithms and separate classes of algorithms, computational models and automata but also classes of classes of algorithms, automata, and computational models and processes. As a result, the axiomatic approach goes higher in the hierarchy of computer and network models, thus reducing the complexity of their study. The suggested axiomatic methodology is applied to the evaluation of possibilities of computers, their software and their networks, with the main emphasis on such properties as computability, decidability, and acceptability. In such a way, it becomes possible to derive various characteristics of types of computers and software systems from the initial postulates, axioms and conditions.

It is also worth mentioning that the axiomatic approach allowed researchers to prove the Church-Turing Thesis for an algorithmic class that satisfies very simple initial axioms [44,45]. These axioms form a system *C* considered in the previous section and this system provides a definite perspective on different classes of algorithms, ensuring that in these classes the Church-Turing Thesis is true, *i.e.*, it is a theorem.

Moreover, the axiomatic approach is efficient in exploring features of innovative hardware and unconventional organization.

It is interesting to remark that algorithms are used in mathematics and beyond as constructive tools of cognition. Algorithms are often opposed to non-constructive, e.g., descriptive, methods used in mathematics. The axiomatic approach is essentially descriptive because axioms describe properties of the studied objects in a formalized way.

Constructive mathematics is distinguished from its traditional counterpart, axiomatic classical mathematics, by the strict interpretation of the expression "there exists" (called in logic the *existential quantifier*) as "we can construct" and show how to do this. Assertions of existence should be backed up by constructions, and the properties of mathematical objects should be decidable in a finite number of steps.

However, in some situations, descriptive methods can be more efficient and powerful than constructive tools. Language allows one to describe many more objects than it is possible to build by available tools and materials. For instance, sufficiently rich logical languages, according to the first Gödel undecidability theorem, can represent statements that are true but are not provable. That is why descriptive methods in the form of the axiomatic approach came back to the theory of algorithms, automata and computation, becoming efficient tools in computer science.

## 6. Conclusions and Future Work

This paper demonstrates the role of the axiomatic methods for the following paradigms of mathematics and computer science:

-*Classical mathematics,* with global axiomatization and classical logic.

-*Local mathematics,* with local axiomatization, diverse logics and logical varieties.

-*Reverse mathematics,* with axiomatic properties decomposition and backward inference.

-*Projective mathematics,* with view axiomatization, logical varieties and properties proliferation.

Here we have considered only some of the consequences of new trends in the axiomatic approach to mathematical cognition. It would be interesting to study consequences of this approach in other fields such as epistemology and computability theory. Furthermore, inasmuch as computer science is based on mathematics, the new paradigms of mathematics presented in this work form corresponding directions in computer science giving an advantage to unconventional computations and nature-inspired architectures of information processing systems. One of the novel approaches is applying the axiomatic methods of the mathematical theory on information technology [46,47].

Another important direction for future work is the study of physical systems as information processing architectures. Computations beyond the Turing model exist not only in the universe of unconventional algorithms but even in the physical universe. The idea of Pancomputationalism (Naturalist computationalism) [48,51] suggests that all of the physical universe can be modelled on different levels of organization as a network of computational processes on informational structures [48], with information defined in the sense of Informational Structural Realism, see [49].

As a consequence, unconventional computing as it appears in natural systems is developing as an important new area of constructive research. It is presented by Stepney [49] and her article in this Special Issue, Cooper [50], authors in [49–52], as well as in the work of Rozenberg and MacLennan, see [52]. Ziegler's suggestion of axiomatizing physical computational universes [53] correlates with both the natural computationalism and the approach of projective mathematics. It remains for future work to establish the connection between unconventional algorithms and natural computing.

## Acknowledgments

The authors would like to thank Andree Ehresmann, Hector Zenil and Marcin J. Schroeder for useful and constructive comments on the previous version of this work and three anonymous reviewers for numerous insightful reflections and helpful advices which considerably improved the article.

## References and Notes

1. Bell, J.L. From absolute to local mathematics. *Synthese* **1986**, *69*, 409–426.
2. Burgin, M. *Super-Recursive Algorithms*; Science+Business Media Inc.: New York, NY, USA, 2005.
3. Minsky, M. *Computation: Finite and Infinite Machines*; Prentice-Hall: New York/London/Toronto, USA/UK/Canada, 1967.
4. Markov, A.A. Theory of Algorithms. In *Transactions of the Mathematical Institute of the Academy of Sciences of the USSR*; Academy of Sciences of the USSR: Moscow, USSR, 1954.
5. Kolmogorov, A.N. On the Concept of Algorithm, *Russ. Math. Surv.* **1953**, *8*, 175–176.
6. Shönhage, A. Storage Modification Machines. *SIAM. J. Comput.* **1980**, *9*, 490–508.
7. Shepherdson, J.C.; Sturgis, H.E. Computability of Recursive Functions. *J. Assoc. Comput. Mach.* **1963**, *10*, 217–255.
8. Petri, C. *Kommunikation mit Automaten*. Ph.D. Thesis, University of Bonn, Bonn, Germany, 1962.
9. Peterson, J.L. *Petri Net Theory and the Modeling of Systems*; Prentice Hall: Englewood Cliffs, New Jersey, 1981.

10. Burgin, M. *Measuring Power of Algorithms, Computer Programs, and Information Automata*; Nova Science Publishers: New York, NY, USA, 2010.
11. Ehresmann, C. Trends toward unity in mathematics. Available online: http://archive.numdam.org/ARCHIVE/CTGDC/CTGDC_1966__8_/CTGDC_1966__8__A1_0/CTGDC_1966__8__A1_0.pdf, accessed on 18 September 2012.
12. Barwise, J.; Seligman, J. Information Flow: The Logic of Distributed Systems. In *Cambridge Tracts in Theoretical Computer Science 44*; Cambridge University Press: Cambridge, UK, 1997.
13. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: Singapore, 2010.
14. Burgin, M. Knowledge in Intelligent Systems. In Proceedings of the Conference on Intelligent Management Systems, Varna, Bulgaria, 1989; pp. 281–286.
15. Minsky, M. *A Framework for Representing Knowledge*; MIT: Cambridge, MA, USA, 1974.
16. Minsky, M. Society of Mind: A Response to Four Reviews. *Artif. Intell.* **1991**, *48*, 371–396.
17. Minsky, M. Conscious Machines. In *Machinery of Consciousness*, Proceedings of the National Research Council of Canada, 75th Anniversary Symposium on Science in Society, Ottawa, Canada, 1991.
18. Delgrande, J.P.; Mylopoulos J. Knowledge Representation: Features of Knowledge. In *Fundamentals of Artificial Intelligence*; Springer: New York/Berlin/Heidelberg, USA/Germany, 1986; pp. 3–38.
19. Burgin, M.; de Vey Mestdagh, C.N.J. The Representation of Inconsistent Knowledge. In *Advanced Knowledge Based Systems, Lecture Notes in Computer Science*, *Knowlege-Based and Intelligent Information and Engineering Systems*; Springer: Berlin/Heidelberg, Germany, 2011; Volume 6882, pp. 524–537.
20. Burgin, M. Logical varieties and covarieties. In *Theoretical Problems of Mathematics and Information Sciences*; Ukrainian Academy of Information Sciences: Kiev, Ukraine, 1997; pp. 18–34. (In Russian).
21. Burgin, M. Logical Tools for Program Integration and Interoperability. In Proceedings of the IASTED International Conference on Software Engineering and Applications; MIT: Cambridge, MA, USA, 2004; pp.743–748.
22. Suppe, F. Axiomatization. In *Companion to the Philosophy of Science*, Newton-Smith, W.H., Ed.; Blackwell: Oxford, UK, 2001.
23. Biology, Philosophy of. *Encyclopædia Britannica. Encyclopædia Britannica Online Academic Edition*. Available online: http://www.britannica.com/EBchecked/topic/681551/philosophy-of-biology, accessed on 18 September 2012.
24. Kunii, T.L. The Potentials of Cyberworlds—An Axiomatic Approach. In 3rd International Conference on Cyberworlds (CW'04), Tokyo, Japan, 2004; pp. 2–7.
25. Manna, Z. *Mathematical Theory of Computation*; McGraw Hill: New York, NY, USA, 1974.
26. Milner, M. *Communication and Concurrency*; Prentice Hall: Englewood Cliffs, NJ, USA, 1989.
27. Meyer, A.R.; Halpern, J.Y. Axiomatic definitions of programming languages: A theoretical assessment (preliminary report). In Proceedings of the 7th ACM SIGPLAN-SIGACT Symposium on Principles of programming languages, Las Vegas, NV, USA, 1980; pp. 203–212.

28. Friedman, H. Axiomatic recursive function theory. In *Logic Colloquium '69*, *Studies in Logic*; North-Holland: Amsterdam, The Netherlands, 1971; pp. 113–137.
29. Gritliot, T.J. Dissecting abstract recursion. In *Generalized Recursion Theory*; North-Holland: Amsterdam, The Netherlands, 1974; pp. 405–420.
30. Fenstad, J.E. Computation theories: An axiomatic approach to recursion on general structures. In *Lecture Notes in Mathematics*; Springer: Berlin/Heidelberg, Germany, 1975; pp. 143–168.
31. Ershov, A.P. Abstract computability on algebraic structures. In *Algorithms in Modern Mathematics and Computer Science*; Springer: Berlin, Germany, 1981; pp. 397–420.
32. Thompson, S. Axiomatic Recursion Theory and the Continuous Functionals, *J. Symbolic Logic* **1985**, *50*, 442–450.
33. Skordev, D. *Computability in Combinatory Spaces: An Algebraic Generalization of Abstract First Order Computability*; Kluwer Academic Publishers: Dordrecht-Boston-London, Netherlands/USA/UK, 1992.
34. Friedman, H.; Hirst, J. Weak comparability of well orderings and reverse mathematics. *Ann. Pure Appl. Logic* **1990**, *47*, 11–29.
35. Giusto, M. Simpson S.G. Located Sets and Reverse Mathematics. *J. Symbolic Logic* **2000**, *65*, 1451–1480.
36. Friedman, H. Systems of second order arithmetic with restricted induction, I, II (Abstracts). *J. Symbolic Logic* **1976**, *41*, 557–559.
37. Simpson, S.G. Subsystems of Second Order Arithmetic. In *Perspectives in Mathematical Logic*; Springer-Verlag: Berlin, Germany, 1999.
38. Fraenkel, A.A.; Bar-Hillel, Y. *Foundations of Set Theory*; North-Holland: Amsterdam, The Netherlands, 1958.
39. Kurosh, A.G. *Lectures on General Algebra*; Chelsea Publishing Company: New York, NY, USA, 1963.
40. Kelley, J.L. The Tychonoff product theorem implies the axiom of choice. *Fund. Math.* **1950**, *37*, 75–76.
41. Howard, P.; Rubin, J. *Consequences of the Axiom of Choice*. In *Mathematical Surveys and Monographs*; American Mathematical Society: Boston, MA, USA, 1998.
42. Kolmogorov, A.N.; Fomin, S.V. *Elements of the Theory of Functions and Functional Analysis*; Dover Publications: New York, NY, USA, 1999.
43. Bernays, P.; Hilbert, D. In *The Encyclopedia of Philosophy,* Version 3; Macmillan and The Free Press: New York, NY, USA, 1967; pp. 496–504.
44. Boker, U.; Dershowitz, N.A. Formalization of the Church-Turing Thesis for State-Transition Models, 2004. Available online: http://www.cs.tau.ac.il/, accessed on 18 September 2012.
45. Dershowitz, N.; Gurevich, Y.A. Natural Axiomatization of Computability and Proof of Church's Thesis. *Bull. Symbolic Logic* **2008**, *14*, 299–350.
46. Burgin, M. Mathematical Theory of Information Technology. In Proceedings of the 8th WSEAS International Conference on Data Networks, Communications, Computers (DNCOCO'09); Baltimore, MD, USA, 2009; pp. 42–47.

47. Burgin, M. Levels of System Functioning Description: From Algorithm to Program to Technology. In Proceedings of the Business and Industry Simulation Symposium, FL, USA, 2003; pp. 3–7.
48. Dodig-Crnkovic, G.; Müller, V. A Dialogue Concerning Two World Systems: InfoComputational *vs.* Mechanistic. In *Information and Computation*. Dodig-Crnkovic, G., Burgin, M., Eds.; World Scientific: Singapore, 2011.
49. Stepney, S.; Braunstein, S.L.; Clark, J.A.; Tyrrell, A.M.; Adamatzky, A.; Smith, R.E.; Addis, T.; Johnson, C.G.; Timmis, J.; Welch, P.H.; *et al*. Journeys in non-classical computation I: A grand challenge for computing research. *Int. J. Parallel Emergent Distrib. Syst.* **2005**, *20*, 5–19.
50. Cooper, S.B. The Mathematician's Bias- and the Return to Embodied Computation. In *A Computable Universe.* In *Understanding Computation & Exploring Nature As Computation*, Zenil, H., Ed.; World Scientific: Singapore, 2012.
51. Dodig-Crnkovic, G. Significance of Models of Computation from Turing Model to Natural Computation. *Mind. Mach.* **2011**, *21*, 301–322.
52. Burgin, M.; Dodig-Crnkovic, G. From the Closed Classical Algorithmic Universe to an Open World of Algorithmic Constellations. In *Computing Nature,* Dodig-Crnkovic, G., Giovagnoli, R., Eds.; Springer: Heidelberg, Germany, SAPERE series, 2012.
53. Ziegler, M. Physically-relativized Church-Turing Hypotheses. *Appl. Math. Comput*. **2009**, *215*, 1431–1447.

© 2012 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).

information

MDPI

Article

# Prolegomena to an Operator Theory of Computation

**Mark Burgin [1,*] and Gordana Dodig-Crnkovic [2]**

1 Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA
2 Department of Computer Science and Engineering, Chalmers University of Technology, 41296 Gothenburg, Sweden; gordana.dodig-crnkovic@chalmers.se
* Correspondence: mburgin@math.ucla.edu

Received: 20 May 2020; Accepted: 29 June 2020; Published: 4 July 2020

**Abstract:** Defining computation as information processing (information dynamics) with information as a relational property of data structures (the difference in one system that makes a difference in another system) makes it very suitable to use operator formulation, with similarities to category theory. The concept of the operator is exceedingly important in many knowledge areas as a tool of theoretical studies and practical applications. Here we introduce the operator theory of computing, opening new opportunities for the exploration of computing devices, processes, and their networks.

**Keywords:** computation; information; info-computation; computing device; operator; network; computer architecture; composition

## 1. Introduction

The operator approach in computer science is, as we explain below, in essence similar to the category methodology in mathematics, providing innovative tools for the exploration of computing systems and computation.

The concept of a category was introduced by Eilenberg and McLane [1]. It was stimulated by the discovery and study of relations between abstract algebra and topology. The main idea of the category methodology is that the notion of structure-preserving mapping, such as a homomorphism in algebra or a continuous mapping in topology, is in many situations more important than the inner construction of mathematical objects when they are built from separate elements.

Traditionally, systems, in general, and mathematical systems, in particular, have been described by their inner structures, i.e., elements and relations between them [2,3]. For instance, a group is defined as a set $G$ of elements with a binary operation over its elements, which satisfies axioms of closure, associativity, and has an identity element and inverse element.

In contrast to group theory or topology, category theory suggested describing mathematical systems by their *structure-preserving mappings* (see Goldblatt [4]), i.e., by their *outer structures* in the sense of the general theory of structures developed by Burgin [5].

In a similar way, computing systems have traditionally been described by their inner structures in the form of elements, components, and basic operations. For instance, a Turing machine contains three basic components: the control device, read/write head, and memory (tape). The control device governs the functioning of the head while the head performs transformations in the memory of the machine.

In contrast to this, operator theory suggested describing computing systems by their functioning, i.e., by their *pure external structures* in the sense of the general theory of structures [5]. Namely, a computing system generates a process of transformation of the input objects into the output objects. This can be described within the framework of info-computation [6], which models information as a structure and computation as information transformation, i.e., the dynamics of information.

Each such transformation can be a process that consists of many steps of elementary transformations of information structures. The data in this context are atoms of information, or the most elementary units of information, which are building increasingly complex structures through the transformations, that is, the process of computation.

Category theory of mathematical systems changes their descriptions by inner structures to their descriptions by outer structures. In a similar way, operator theory of computing systems changes their descriptions by inner structures to their descriptions by external structures.

In terms of operator theory, the system of all processes generated by a computing system $Q$ is described as an operator $A_Q$, while each such process (transformation) is an application of operator $A_Q$ to the input objects. For instance, a Turing machine operator describes transformation of the input words into the output results by a Turing machine.

Operators can be used to represent important *ontological features* of reality reflecting theirdynamical nature through a process when some objects act on other objects. Our universe exists in interactions of its components and elements, while every interaction can be decomposed in separate actions, each involving an operator.

At the same time, *epistemic* operators express the basic mechanisms of cognition, that is, the process of knowing about the world and its ontology. Cognitive processes, such as search, selection or recognition and learning, are performed by *physical* processes which can be represented by abstract operators decomposable into simpler operators. In the context of epistemic processes, any cognizing (knowledge-generating) physical system, such as a scientist, a measuring device, or a computer, is a *physical operator*.

Given their broad applicability, operators are one of the most important tools in sciences. In theoretical physics, both classical physics and quantum physics have operator formulations (cf., for example [7,8]). In mathematics, there is operator theory, which studies operators in Hilbert or Banach spaces (cf., for example [9–11]). Operators are used in chemistry as well as in computer science. In addition, operators are also becoming an important tool in information theory (cf., for example [5,12–14]).

Computing devices are information processors, transformers, transmitters, and generators of information. That is why in this work, we develop operator models of computing devices and study their properties based on the ontological operator theory originated by Burgin and Brenner [15].

The operator approach has many advantages. For instance, why do physicists call functions from one vector space to another by the new name *operator*? The reason is that in physics operators act mostly in infinite dimensional vector spaces, e.g., Hilbert spaces. Such spaces have many advanced properties. These properties are reflected in properties of operators, which become much more complex and sophisticated than conventional numerical functions. Even more, new properties which numerical functions do not have, such as the spectrum, were discovered in operators and proved very useful.

The same situation has become apparent in computer science. At first, operations of abstract and physical automata were represented by functions, such as the transition function of an automaton. However, processed data started becoming more and more complex and sophisticated bringing the necessity of introducing operators to describe functioning of abstract and physical automata.

In quantum computation, transition to operators happened automatically because quantum processes are traditionally described in physics by operators. For instance, a quantum circuit is a system of quantum gates and each quantum gate is a unitary operator, which acts on states of quantum systems.

However, even before emergence of quantum computation, operations in many programming languages are called operators. For instance, in computer programs, one of the most familiar sets of operators, the Boolean operators, is used to work with true/false values. Boolean operators include AND, OR, NOT (or AND NOT), and NEAR. These (and variations, such as XOR) are used in logic gates. Another class used in computer programming is formed by arithmetical operators, which contain +,

−, × and ÷. Other types of operators used in computer programming include *assignment operators*, which assign a specified value to another value and *relational operators*, which compare two values.

The reason for treating AND, OR, ADD, and other similar programming objects as operators and not only as operations is that an operator is what performs data transformations and an operation is the transformation itself. This distinction is explicitly exposed in the general model of an operator introduced in the ontological operator theory [15].

Another example of operator utilization comes from parallel programming presented by Pingali et al. [16]. A further practical area that brings forth utilization of operators for modeling, exploration, and application is in biological and chemical computers and other natural computers addressed by Rozenberg et al. [17] and Adamatzky [18]

One more theoretical area that brings forth utilization of operators for modeling, exploration and application is the theory of structural machines of Burgin and Adamatzky [19,20].

Morphological computing is a model of physical computation that especially involves transformations of a physical system, typically on a hierarchy of levels of organization, as argued by Dodig-Crnkovic [6]. It involves transformations of information structures (including elementary information structures—data structures), which are efficiently modeled by operators representing information dynamics.

## 2. The Concept of Operator

In physics, an operator is a function over a space of initial physical states to the space of final states. In classical mechanics, the movement of a particle (or a system of particles) is completely determined by the Lagrangian or, equivalently, the Hamiltonian operator of a system.

Operators in classical mechanics are related to symmetries which reflect invariance of motion with respect to a coordinate (Noether's theorem). Thus, translational, rotational, Galilean transformation, parity and T-symmetry, each is connected with a specific classical mechanic operator. Operators in quantum mechanics are integral part of the formulation of QM. Thus, position, momentum, kinetic energy, angular momentum, spin, and Hamiltonian are expressed as operators in QM.

An example from quantum physics is S-matrix (scattering matrix), which denotes an operator that describes the process of transfer of a quantum-mechanical system from the initial state to the final one as a result of a scattering. Taking the set of quantum numbers describing the initial and final states, the scattering amplitudes form a table, which is called the scattering matrix S.

In quantum chemistry according to Levine [21], an operator is defined as "a rule that transforms a given function into another function". The differentiation operator $d/dx$ is an example of operator that transforms a differentiable function $f(x)$ into another function $f'(x)$. Other examples include integration, the square root, and so forth. Numbers can also be considered as operators (they multiply a function). McQuarrie [22] gives an even more general definition for an operator: "An operator is a symbol that tells you to do something with whatever follows the symbol".

Operators are widely used in computer programming as well. For example, the Boolean operators, AND, OR, NOT (or AND NOT), and NEAR, with variations such as XOR, are used in logic gates. Furthermore, assignment operators, which assign a specified value to another value and relational operators, which compare two values are widely used in computer programming.

According to the Techopedia [23], an operator in computer programming is a symbol that usually represents an action or a process. An operator is used for manipulating a certain value or operator. For example, in "1 + 2", the "1" and "2" are the operands and the plus symbol is the operator. Common operators in programming languages are =, ==, +, ++, −, /, *, >, <, etc.

The unified operator theory of Burgin and Brenner [15] provides the most encompassing definitions of operators and related concepts. Here we use definitions from this theory.

**Definition 1.** *An operator is an object (system) that operates, i.e., performs operations on, some objects, systems or processes, which are called operands of this operator.*

This brings us to the following definition:

**Definition 2.** *An operand is an object, system or process operated by an operator.*

These definitions show that being an operator or an operand is a role and a characteristic of a system/object. One and the same system/object can be an operator in some situations and an operand in other situations. In a similar way, a system/object can be an operator with respect to some systems and not an operator with respect to other systems.

Definitions 1 and 2 form the foundation of the unified operator theory [1], which can be specified for a diversity of specialized operator theories, such as operator theories in physics and chemistry, or the theory of programming operators.

Definitions 1 and 2 also express the fundamental dyadic relation between operators and their operands, which is actualized in the form of the operator triad:

$$\text{Operator} \xrightarrow{\text{Operation/function}} \text{Operand}$$

This diagram presents an operation as a component of an operator triad.

The operator triad is a special case of the basic fundamental triad [5,24]. In the symbolic representation, it has the form:

$$(\text{Op, on, Od})$$

where Op is an operator, on is an operation, and Od is an operand.

To construct a general mathematical operator theory in some domain, for example, in the realm of computations, it is necessary to organize the multiplicity of relevant operands in the form of an operating space, i.e., the space that is transformed by an operator.

In this context, the key formal model of an operator Op also has the form of the basic fundamental triad:

$$\text{Op} = (\text{D, on, C})$$

where D = D(Op) is the domain of the operator Op, i.e., a space that contains all objects that are operands of this operator on is the operation that the operator Op performs.

C = D(Op) is the codomain of the operator Op, i.e., a space that contains all objects that are results of this operator.

Together the domain D and codomain C form the operating space of the operator Op.

An arbitrary operator A is not necessarily defined for all elements from its domain D(A). The subspace (subset) of D(A) where A is defined is called the definability domain and denoted by DD(A). For instance, taking a Turing machine $T_\emptyset$ that works with words in the alphabet {0, 1} but never halts independently of its input, we see the domain $D(T_\emptyset)$ is the set of all words in the alphabet {0, 1} while the definability domain $DD(T_\emptyset)$ is the empty set $\emptyset$.

In a similar way, the range R(A) of an operator A, i.e., the set of all elements that are values of A, can be only a part of its codomain C(A). For instance, the codomain $C(T_\emptyset)$ is the set of all words in the alphabet {0, 1} while the range $R(T_\emptyset)$ is the empty set $\emptyset$.

## 3. The Concept of the Information Operator

Different types of operators function in distinct operating spaces. For instance, operators of quantum mechanics operate on Hilbert spaces.

Information operators work in information spaces. As there are diverse types of information, operator representation demands different types of information spaces. According to the multiscale taxonomy of information, Burgin and Dodig-Crnkovic [25] differentiate among syntactic, semantic, and pragmatic information; algorithmic and descriptive information; and cognitive, effective, and emotional information. In this context, each type of information has the corresponding type of

information spaces. Here we should add that, according to recent results from cognitive science and neuroscience, cognitive and emotional phenomena cannot be treated separately [26]. With modern understanding of embodied, embedded, and enacted (EEE) cognition, emotions are an integral part of a real-life process of cognition [26], which is based on information processing.

There are three basic types of information operators:

- *Substantial information operators* transform physical objects into structural objects, i.e., their domain consists of physical objects while structural objects shape their range.
- *Co-substantial information operators* transform structural objects into physical objects, i.e., their domain consists of structural objects while physical objects compose their range.
- *Pure information operators* transform structural objects into structural objects, i.e., their domain and range consist of structural objects.

In all cases, structural objects are interrelated forming information spaces with various operations acting on these objects. In what follows we mostly concentrate on pure information operators, the domain and range of which are some information spaces.

While modeling computing devices by information operators, we treat operators as tools for transformation and generation of distinct kinds of information, and call them *computational information operators*. In the case of computing over information spaces, the most important classes of involved information are syntactic, semantic, and pragmatic information. This brings us to syntactic, semantic, and pragmatic information spaces, which are the most used operating spaces for computational information operators.

### *3.1. Syntactic Information Spaces*

Digital computing devices process information in symbolic form, transforming words of some language. Consequently, it is usual to represent both physical and abstract, *syntactic information spaces* as systems of formal, artificial or natural languages. This implies that treating these devices as operators, we encounter operands of five types: separate symbols, words, texts, languages, and families of languages.

In the case of natural computing that refers to physical computations (morphological, chemical, cognitive, quantum, and other kinds of physical computation) we have *physical objects* that perform computations. Molecules, for example, on which our brains perform computations, are not only symbols as words of an electro-chemical language, but they are concrete physical objects as argued by Alcami and El Hady [27] and Silver [28]. That is why emotions are part of cognitive process, since cognition is not only electro-chemical *symbol* manipulation, but also embodied electro-chemical *material object* manipulation [26]. There is a difference between the symbol/word and the object which it represents. In our case, it is the difference between the classical Turing model of logical computing machine type of computation and natural computation in the real physical world, elaborated by Rozenberg et al. in [17].

Taking an abstract computing device such as a Turing machine, we come to the *syntactic operating space*, which consists of all formal *languages* with the alphabet of a particular Turing machine. At the same time, it is possible to take the space of all *strings* in some alphabet as the syntactic operating space of a Turing machine. The property of the Turing machine is that *it identifies symbolic representation of a machine with a machine itself* as an object in the symbolic world or symbolosphere [29]. That is the basis of programmable computing.

Note that while the Turing machine is not a concrete/physical object, but a symbolic model of computation, in our approach, the grounding of the concept of computation is achieved by allowing the most fundamental model of computation being the physical process itself. It is similar to Rodney Brooks' idea of AI without representation [30]. Philosophically, this points back to the symbol grounding problem, which is resolved in AI through grounding symbols in embodied cognition of agents, based on signals obtained directly from the physical world through sensors/senses.

In the case of finite automata, it is possible to utilize syntactic information spaces of four types. A syntactic information space of the first type consists of all *symbols* from the alphabet of the finite automaton. A syntactic information space of the second type consists of all symbols denoting *states* of the finite automaton. A syntactic information space of the third type consists of all *words* from the formal languages with the alphabet of the finite automaton. A syntactic information space of the fourth type consists of all *formal languages* with the alphabet of the finite automaton.

The majority of abstract automata (computing devices) work with linear (i.e., one-dimensional) languages. However, there are also abstract automata (computing devices) that work with more complex structures. For instance, Kolmogorov algorithms work with arbitrary graphs [31], Turing machines with two-dimensional tapes and two-dimensional cellular automata work with two-dimensional structures while structural machines work with arbitrary structures, according to Burgin and Adamatzky [19,20].

As the result, operators representing different abstract automata (computing devices) have different syntactic operating spaces.

Operators that represent Kolmogorov algorithms are Kolmogorov computation operators [31], the syntactic operating space of which is the collection of formal graph languages, i.e., languages the words of which are graphs, while the definability domain consists of all enumerable (recursively computable) graph languages.

Operators that represent two-dimensional cellular automata, as shown by Codd [32] are two-dimensional cellular computation operators, the syntactic operating space of which is the collection of two-dimensional array languages, i.e., languages the words of which are two-dimensional arrays, while the definability domain consists of all enumerable (recursively computable) two-dimensional array languages.

Operators that represent structural machines are structural computation operators, the syntactic operating space of which is the collection of structural languages, i.e., languages the words of which are structures, while the definability domain consists of all enumerable (recursively computable) structural languages.

Operators that represent Turing machines with one-dimensional tapes are one-dimensional Turing computation operators, the syntactic operating space of which is the collection of formal languages while the definability domain consists of all recursively enumerable (recursively computable) languages.

Operators that represent Turing machines with two-dimensional tapes are two-dimensional Turing computation operators, the syntactic operating space of which is the collection of two-dimensional formal languages while the definability domain consists of all recursively enumerable (recursively computable) two-dimensional languages.

### *3.2. Semantic Information Spaces*

Semantic information spaces can also be distinguished based on different kinds of the semantic theory of information. For instance, in the semantic information theory of Bar-Hillel and Carnap, a semantic information space consists of possible worlds according to Bar-Hillel and Carnap [33]. Often a semantic information space is a conceptual space, as studied by Gärdenfors [34–36]. In the theory of epistemic information, a semantic information space is a conceptual space [5,13]. In the semantic information theory of Shreider [37], a semantic information space is a thesaurus as a system of texts and semantic relations between these texts. Conceptual spaces studied by Burgin and Díaz-Nafríagive one more example of semantic information spaces [38]. Utilization of semantic information spaces in modeling of computing devices by information operators allows studying semantic aspects of computation, computing systems and networks.

Natural computation, as it is presented by Dodig-Crnkovic in [39,40], involves different information spaces. Indeed, if we talk of physical computation, we must start with ontology-epistemology relationship with focus on the material properties of objects, instead of their logical properties. We do not logically derive physics, chemistry and biology as computational phenomena, we observe what there is in nature—that is, not a closed logical system, as Burgin and Dodig-Crnkovic [41] argued

in the article describing the shift from the closed classical algorithmic universe to the open world of algorithmic constellations, where algorithms are physical/chemical/biological mechanisms as well. Especially taking into account the whole loop from ontology to epistemology and back as implemented in robotics, we notice that through the phenomenon of morphological computing, computational control of the central controlling mechanism in the robot can be replaced by natural physical behavior of a material of a robots body. For instance, passive dynamic walker robot walks down the slope without any other control, but the physical properties of its body, anticipated through the knowledge of its direct physical interactions with the environment as presented by Pfeifer and Bongard [42].

In case of natural computing/physical computing such as found in living organisms and robots, information semantics defines the relationship between the physical world and its symbolic representation or a behavior of a computational system. When talking about programming languages semantics (and Turing machines is a programming language equivalent), semantics is evaluation of the meaning of syntactically valid strings of symbols defined by a programming language, to the description of the computation involved, that is computational behavior—so it is language mapping.

In the similar way as syntactic and semantic information spaces, it is also possible to introduce pragmatic information spaces and pragmatic information operators. For instance, it is possible to consider the space of goals with corresponding operations as a pragmatic information space. The space of intentions is another kind of pragmatic information spaces.

## 4. The Concept of Computational Information Operator

Computation can be understood as information processing. Nevertheless, information transmission or communication is typically not seen as computation, although it may be taken as an element or part of computation. Computation is typically considered as having some input and producing some output. However, Alcami and El Hady [27] describe axonal information processing (where transmission of information proceedsfrom the cell body to the nerve terminal through an axon) as computation. Shannon defined his concept of information based on a technical model of human communication. Both computation and communication imply the transformation of information (where transformation can be identity). Bohan Broderick [43] compares notions of communication and computation and concludes that computation and communication are not conceptually distinguishable. They may be distinguished with respect to a given system, so that computation is limited to a process *within* a system (such as in the Turing Machine) and communication is an interaction *between* systems or *between* a system and its environment (such as in interactive computing and natural computing).

Dynamics of information is defined as a general form of computation. If the physical universe is an information structure, natural computation is a process governing the change/dynamics of information. Information and computation are two mutually defining concepts as argued by Dodig-Crnkovic [44], which are conceptually combined based on their complementarity, in the concept of "*info-computation*" [25].

Thus, to specify computational information operator within classical model of computation, it is necessary to delineate computation accordingly.

In the same way as there are varieties of concepts and frameworks for information, there are many approaches to descriptions and definitions of computation, demonstrating that defining computation is still an unsolved problem, as argued by Burgin and Dodig-Crnkovic [45].

There are three levels of generality in understanding the phenomenon of computation:

1. On the top (most general) level, computation is perceived as any transformation of information and/or information representation.
2. On the middle level, computation is distinguished as a *discrete* process of transformation of information and/or information representation.
3. On the bottom level, computation is defined as a *discrete* process of *symbolic transformation* of information and/or symbolic information representation in case of classical computation models. Alternatively, in case of natural and unconventional computing, physical/chemical/biological/cognitive processes

that are interpreted as computation or the basis for computational behaviors of physical systems under consideration are at the bottom level.

It is necessary to remark that if we do not go beyond the bottom level and if we insist on discreteness, we would lose continuous time computation realized by general dynamical systems of Bournez [46], hybrid systems of Gupta et al [47], and special computing devices, such as the differential analyzer of Shannon [48]; Moore [49]).

Computation is traditionally defined as transformation of information representation, see, e.g., Kelemen [50], where transformations can be discrete, continuous, or a mixture.

This definition of computation results in separation of substantial types of computation, as explained by Burgin and Dodig-Crnkovic in [45]:

1. *Symbolic computation* when information is represented by physically- or mentally-given symbols.
2. *Material computation* when information is represented by material objects, such as atoms, or molecules of a biological cell, and can be continuous as there are continuous phenomena in many branches of physics.

Typical artificial devices, such as conventional computers and calculators, perform material computations, which represent symbolic computation that is in focus. The same process of symbolic computation can be realized by different material computations, e.g., on different computers. That is the case of universal or general-purpose computers, which are substrate-independent.

There is another type of computer, the analog computer, which is a model for a certain problem that can then be used to solve analog problem by means of simulating it. In the analog computer there is no stored program controlling its operation. Instead, it is programmed by changing the interconnections between computing elements. This type of computation/information processing is similar to the information processing in the human brain.

Quantum computation is either a kind of symbolic computation embodied in material computation where symbols are represented by quantum states or an analog computation called quantum annealing, which is an optimization of the cost or energy functions of complex systems utilizing quantum fluctuations. This approach is used by D-wave computers, who recently claimed attaining quantum supremacy with a 53-qubit superconducting processor [51].

It is sometimes considered an open question whether symbolic computation is possible without material computation, even though the majority of researchers believe that there is no information without physical representation and there is no computation without information as argued by Szilard, Landauer, Swenson, and Lloyd as quoted by Karnani, Pääkkönen, and Annila [52]. Material computation is possible not only in computers as technological artifacts but in a computing nature [38] as a whole. An example of material computation is all physical computation that goes on in all kinds of physical objects, including living cells and living organisms. Neurons organized in neural networks are the only living cells capable of symbolic computation. Ehresmann [53] presents an info-computational model for (neuro-)cognitive systems capable of creativity built on several levels of organization/abstraction.

At the middle level of abstraction, computation is a discrete process of transformation of information and/or information representation reflected by results in three operational types of computation as presented by Burgin [54] and in the taxonomy of computation and information architecture by Burgin and Dodig-Crnkovic [55]:

1. *Discrete computation* with digital operations performed in elementary separate steps.
2. *Continuous computation* when operation goes without breaks in time.
3. *Piecewise continuous* computation, combining discrete and continuous computation.

In addition, we have three temporal types of computation [54,55]:

1. *Sequential computation*, which is performed in linear time.

2. *Parallel or branching* computation, in which separate steps are synchronized in time.
3. *Concurrent computation*, which does not have synchronization in time.

While parallel computation is completely synchronized, branching computation is not completely synchronized because separate branches acquire their own time and become synchronized only in interactions.

Existence of various types and kinds of computation, as well as a variety of approaches to the concept of computation, shows complexity of understanding of computation in a holistic picture.

In what follows, we consider computational information operators that represent algorithms or computing automata. Analog computing, which is not covered here, is presented in the section on natural computing.

The concept of computation stratifies the system of operands for computational information operators. Namely, if A is a computational information operator, its system of operands consists of three components:

1. Input, or initial, operands
2. Processed operands
3. Output, or resulting, operands

In turn, output (resulting) operands are divided into intermediate, final and analytic outputs of the operator in the form of an algorithm or computing automaton. Final outputs are the results of the computation produced by the algorithm or computing automaton. Analytic outputs are results of the computation, which are not produced by the algorithm or computing automaton but are determined theoretically. As an example of this situation, we consider limit Turing machines, which were defined by Burgin [54,56]. All other outputs are called intermediate.

With regard to outputs, computational information operators have three types:

1. *Explicit computational information operators* represent algorithms or computing automata, which function so that the last output is final and/or it is identified by the algorithm or computing automaton.
2. *Implicit computational information operators* represent algorithms or computing automata, which function so that the final output is not always identified by the algorithm or computing automaton.
3. *Analytic computational information operators* produce analytic outputs.

For instance, Turing machines are represented by explicit computational information operators, inductive Turing machines are represented by implicit computational information operators, and limit Turing machines are represented by analytic computational information operators.

There is a variety of techniques for composition of algorithms and computing devices [55]. These compositions induce corresponding compositions of computational information operators. The most popular of them is sequential composition, definition of which is given in Section 7.

**Proposition 1.** *The sequential composition of explicit computational information operators is an explicit computational information operator.*

Indeed, taking two explicit computational information operators, we see that if the first operator has a final output identified by the algorithm or computing automaton because it is an explicit computational information operator then this output goes to the second operator as its input. As the second operator is also explicit, in the case of producing the final output, this output is identified by the algorithm or computing automaton. It means that the sequential composition of these operators is an explicit computational information operator.

Proposition 1 means that that the class of explicit computational information operators is closed with respect of sequential composition.

At the same time, examples show that for explicit and analytic computational information operators, this statement is not valid in a general case.

According to the results of their application, there are three categories of computational information operators, which are defined by their results on valid inputs, that is, inputs such that application of the operator gives results:

1. A single-valued computational information operator produces at most one result for any valid input.
2. A finite-valued computational information operator can produce a finite number of results for any valid input.
3. An infinite-valued computational information operator can produce an infinite number of results for some valid input.

Deterministic computing automata and algorithms are represented by single-valued computational information operators, while nondeterministic computing automata and algorithms are usually represented by finite-valued or infinite-valued computational information operators. However, nondeterministic accepting automata and algorithms are also represented by single-valued computational information operators.

Finite-valued computational information operators are also represented by multiple computations studied by Burgin in [57].

**Proposition 2.** *The sequential composition of single-valued computational information operators is a single-valued computational information operator.*

Indeed, if the first operator is single valued, it produces, at most, one result, which serves as the input to the second operator in the composition, which also produces, at most, one result because it is also single-valued.

Proposition 2 means that that the class of single-valued computational information operators is closed with respect to sequential composition.

**Proposition 3.** *The sequential composition of finite-valued computational information operators is a finite-valued computational information operator.*

Proof is similar to the proof of Proposition 2.

Proposition 3 means that that the class of finite-valued computational information operators is closed with respect of sequential composition.

At the same time, examples show that, for infinite-valued computational information operators, this statement is not valid in a general case.

Note that a valid input to two operators can be invalid for their composition.

## 5. Computational Information Operators and Natural Computation

An operator formulation of Info-computational framework and its application on natural computation:

$$\text{(Information State Final)} = \text{Operator} \times \text{(Information State Initial)}$$

In this case, the general structure (Operator, Operation, Operand) takes the form of the triad:

$$\text{(Information StateInitial, Computation, Information StateFinal)}$$

Burgin and Dodig-Crnkovic [25] argue that information in the world appears on a multiple scales or levels of organization or levels of abstraction as well as in multiple dimensions.

In natural computing or computing nature [39], the whole of the nature is seen as a network of networks of computational processes on different levels of organization. Similarly, Zenil [58] presents the idea of a computable universe with both ambition to understand computation and exploring nature as computation.

The idea of natural info-computation [59] as a dynamics of information systems successively connects and relates their information structures (states) and makes it possible for a cognizing agent to get an idea about the information content of the world that gets revealed in the agent after sequential processes of information transformation from the source to the receiver. In [26], Dodig-Crnkovic, presents a framework of computing nature [39], where nature is regarded as a network of networks of morphological info-computational processes unfolding through information processing for cognitive agents. "Morphological" stands for a computing model based on information about the form or structure of material system performing computation [6].

Natural computation is a process studied within a field of natural computing or computing nature [39]. As elaborated in the *Handbook of Natural Computing* [17], it consists of three classes of methods:

(1) inspired by nature for the development of novel problem-solving techniques (e.g., cellular automata, neural computation, evolutionary computation, swarm intelligence, artificial immune systems, membrane computing, amorphous computing, cellular computing, molecular computing)
(2) based on the use of computers to synthesize/simulate natural phenomena (e.g., artificial life, artificial chemistry); and
(3) using natural materials (e.g., molecules) to compute (e.g., molecular computing or quantum computing).

Nature is increasingly modeled as computational i.e., information processing system in many research fields, such as systems biology, synthetic biology, cellular computing, cognitive computing, social computing, and morphological computing.

In two research fields of morphological computing—within robotics and a more general one that Turing started with his morphogenesis paper, the relational character of information structures and their dynamics makes it suitable for the application of operator formalism. Fields of application include neuroscience, neurobiology, information processes in neurons and neural systems, bioinformatics, computational biology, learning, memory, neuron, synapse, and biological information systems.

As mentioned, quantum computing is one of the important fields of natural computing. Digital quantum computing uses quantum logic gates to perform computation. It has the advantages of universality, scalability, and quantum error correction, but physical resource requirements to implement error-corrected quantum algorithms are huge. Analog quantum computing (quantum simulation, quantum annealing, and adiabatic quantum computation) is used to avoid the complexity of classical simulations of many-body quantum systems which grows exponentially with the dimension of the system. Feynman suggested the simulation of these problems by another fully-controllable quantum system with a similar encoded dynamics. Utilization of different models of quantum computing involving dissimilar quantum theories makes it an important problem to develop a unified operator theory of quantum computing.

## 6. Computational Information Operators as an Efficient Tool in Computer Science

Let us consider advantages and possibilities opened by operator representation of *computing devices*.

**First possibility**: Operator representation of computing devices allows formulating and solving many problems about these of computing devices in a more general context of operating spaces of operators.

An example of such a problem is the *Definability Problem*, which is called the Halting Problem for Turing machines because definability for a Turing machine is equivalent to halting. In particular,

according to the results of Church and Turing, λ-definable functions are functions that are "effectively" (by mechanical methods) computable. Turing showed that the class of all Turing machines is equivalent to the class of all λ-definable functions. This was an important step in recognition of Turing machine as the supreme model of algorithm.

Let us consider the Definability Problem for operators.

**Definability Problem**. Given a class **K** of operators, is there an operator *B* in **K** such that for any element *x* from the domain of operators from **K** and any operator *A* from **K,** B determines whether *x* belongs to the definability domain of *A* or does not belong.

**Second possibility**: Operator representation of computing devices allows constructing a variety of operator compositions (operations) and developing new schemas of computation as well as new network and computer architectures using operations with (a composition of) operators.

Let us look at two examples of such compositions.

**Example 1.** *Sequential composition*

*Given two operators A and B, their sequential composition is the operator C such that C(x) is equal to B(A(x)) when:*

*(1) A(x) is defined and belongs to the domain of B;*
*(2) B(A(x)) is defined.*

*Otherwise, C gives no result being applied to x.*

**Example 2.** *Disjunctive parallel composition*

*Given two operators A and B, their disjunctive parallel composition is the operator H such that the result of application of D to any operand u is performed so that A and B are applied to u at the same time and D(u) = A(u) if A gets its result at the same time or earlier than B; otherwise, D(u) = B(u).*

**Third possibility**: Operator representation of computing devices allows efficient application of the axiomatic technique for investigation of computing devices, algorithms and computations.

Let us consider a class **K** of algorithms (computing devices) and the corresponding class $\mathbf{O_K}$of operators, which model algorithms (computing devices) from **K**.Here are some examples of axioms, which characterize the class $\mathbf{O_K}$.

***Totality axiom***: For any operator *A* from $\mathbf{O_K}$, $D(A) = DD(A)$.
***Domain stability axiom***: For any operators *A* and *B* from $\mathbf{O_K}$, $D(A) = D(B)$.
***Domain loop axiom***: For any operator *A* from $\mathbf{O_K}$, $D(A) = C(A)$.

This allows obtaining similar axioms for the class **K**.

***Totality axiom***: For any computing device (algorithm) *R* from **K**, $D(A) = DD(A)$.
***Domain stability axiom***: For any computing devices (algorithms) *R* and *Q* from **K**, $D(A) = D(B)$.
***Domain loop axiom***: For any computing device (algorithm) *R* from **K**, $D(A) = C(A)$.

The axiomatic theory of algorithms and computations has been created and developed in the context of functions [60–62]. The transition from functions to operators allows essentially expand the axiomatic theory of algorithms including quantum and natural computations as well as algorithmic functioning of structural machines [63].

Note that it is possible to apply these axioms to algebras of operators such as von Neumann algebras [64] or Kleene algebras [65].

This was sufficient for computations with simple structures, e.g., symbols or words, with which finite automata and Turing machines work. When computational structures become more sophisticated, as in the case of quantum computers, morphological computations and structural machines, it is necessary to utilize operators to represent computational media, computing devices, and computations.

## 7. Conclusions

*Operator representation of computing processes*—both abstract and physical—allows the application of mathematical operator theory to automata, algorithms, and computations. Even though all computation is always performed on some physical substrate, including Turing computation that presupposes a human computer with a pencil and rubber and a piece of paper, classical theory of computing does not investigate its physical substrate and it is developed exactly to *abstract from the detail of physical implementation*. That is how we can run programs on a variety of computational devices, from smallest sensors equipped with some control to supercomputers and various networks.

Classical computing machinery that we have today is energy consuming, non-resilient, and it is not very well suited for simulations of complex quantum systems or representation of the information processing in the brain or in large and interconnected economic or social systems. Thus, we are interested in not only general-purpose, substrate-independent abstract types of computation, but also such computational devices that are dedicated to specific computational problems, such as quantum computers or cognitive computational devices.

An interesting problem is to build operator spaces and to study operators in these different spaces, which represent different classes of systems with different computational characteristics.

The present study demonstrates how to build a variety of novel operations with operators. At the same time, operator algebras studied in mathematics employ only classical (standard) operations, such as sequential composition.

This brings us to one more interesting problem of construction and explorations of operator algebras with nonstandard operations, which, at the same time, may be a contribution to the research in mathematics and in the theory of computing.

**Author Contributions:** Both authors have equally contributed to the article, its conceptualization, methodology, investigation, writing—original draft preparation, review, and editing. Both authors have read and agreed to the published version of the manuscript. All authors have read and agreed to the published version of the manuscript.

**Funding:** The research of the one co-author, GD-C, was funded by Swedish Research Council grant MORCOM@COGS.

**Acknowledgments:** We would like to acknowledge the contribution from the colleagues at IS4SI 2019 conference Theoretical Information Studies in Berkeley where the first version of this work was presented for interesting discussions. Remarks of the reviewers were also useful for the improvement of the paper.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## References

1. Eilenberg, S.; MacLane, S. Relations between homology and homotopy groups of spaces. *Ann. Math.* **1945**, *46*, 480–509. [CrossRef]
2. Bourbaki, N. *Elements de Mathematique. Theorie des Ensembles*; Hermann: Paris, France, 1960.
3. Robinson, A. *Introduction to Model Theory and Metamathematics of Algebra*; North-Holland Publishing: Amsterdam, The Netherlands, 1963.
4. Goldblatt, R. *Topoi: The Categorical Analysis of Logic*; North Holland Publishing: Amsterdam, The Netherlands, 1979.
5. Burgin, M. Epistemic Information in Stratified M-Spaces. *Information* **2011**, *2*, 697–726. [CrossRef]
6. Dodig-Crnkovic, G. Nature as a Network of Morphological Infocomputational Processes for Cognitive Agents. *Eur. Phys. J. Spec. Top.* **2017**, *226*, 181–195. [CrossRef]
7. Von Neumann, J. *Mathematical Foundations of Quantum Mechanics*; Princeton University Press: Princeton, NJ, USA, 1955.
8. Exner, P.; Havlíček, M. *Hilbert Space Operators in Quantum Physics*; Springer: New York, NY, USA, 2008.
9. Brown, A.; Pearcy, C. *Introduction to Operator Theory I: Elements of Functional Analysis*; Springer-Verlag: New York, NY, USA, 1977.

10. Ball, J.A.; Bolotnikov, V.; Helton, J.W.; Rodman, L. (Eds.) *Topics in Operator Theory (Operator Theory: Advances and Applications)*; BirkhäuserVerlag: Basel, Switzerland, 2010.
11. Burgin, M. *Semitopological Vector Spaces: Hypernorms, Hyperseminorms and Operators*; Apple Academic Press: Toronto, ON, Canada, 2017.
12. Burgin, M. Weighted E-Spaces and Epistemic Information Operators. *Information* **2014**, *5*, 357–388. [CrossRef]
13. Harris, Z. *A Theory of Language and Information: A Mathematical Approach*; Oxford University Press: Oxford, UK, 1991.
14. Brenner, J.; Burgin, M. Information as a Natural and Social Operator. *Inform. Theor. Appl.* **2011**, *18*, 33–49.
15. Burgin, M.; Brenner, J. Operators in Nature, Science, Technology, and Society: Mathematical, Logical, and Philosophical Issues. *Philosophies* **2017**, *2*, 21. [CrossRef]
16. Pingali, K.; Nguyen, D.; Kulkarni, M.; Burtscher, M.; Hassaan, M.A.; Kaleem, R.; Lee, T.H.; Lenharth, A.; Manevich, R.; Mendez-Lojo, M.; et al. The Tao of Parallelism in Algorithms. In *PLDI '11: Proceedings of the 32nd ACM SIGPLAN Conference on Programming Language Design and Implementation*; Association for Computing Machinery: New York, NY, USA, 2011; pp. 12–25. [CrossRef]
17. Rozenberg, G.; Back, T.; Kok, J. (Eds.) *Handbook of Natural Computing*; Springer Verlag: Berlin, Germany, 2012.
18. Adamatzky, A. (Ed.) *Unconventional Computing. A Volume in Encyclopedia of Complexity and Systems Science*, 2nd ed.; Springer Nature: Basel, Switzerland, 2018.
19. Burgin, M.; Adamatzky, A. Structural Machines as a Mathematical Model of Biological and Chemical Computers. *Theor. Appl. Math. Comput. Sci.* **2017**, *7*, 1–30.
20. Burgin, M.; Adamatzky, A. Structural machines and slime mold computation. *Int. J. Gen. Syst.* **2017**, *45*, 201–224. [CrossRef]
21. Levine, I.N. *Quantum Chemistry*, 4th ed.; Prentice Hall: Englewood Cliffs, NJ, USA, 1991.
22. McQuarrie, D.A. *Quantum Chemistry*; University Science Books: Mill Valey, CA, USA, 1983.
23. What Does Operator Mean (in Computer Programming)? *Techopedia.* Available online: https://www.techopedia.com/definition/3485/operator-programming (accessed on 8 May 2020).
24. Burgin, M. Triadic Structures in Interpersonal Communication. *Information* **2018**, *9*, 283. [CrossRef]
25. Burgin, M.; Dodig-Crnkovic, G. A Multiscale Taxonomy of Information in the World. In *Theoretical Information Studies. Information in the World*; World Scientific Series in Information Studies; Burgin, M., Dodig-Crnković, G., Eds.; World Scientific: Singapore, 2019; Volume 11.
26. Dodig-Crnkovic, G. Cognition as Embodied Morphological Computation. In *Philosophy and Theory of Artificial Intelligence 2017*; PT-AI 2017. Studies in Applied Philosophy, Epistemology and Rational Ethics; Müller, V., Ed.; Springer: Cham, Switzerland, 2018; Volume 44.
27. Alcami, P.; El Hady, A. Axonal Computations. *Front. Cell. Neurosci.* **2019**, *13*, 413. [CrossRef]
28. Silver, A.R. Neuronal arithmetic. *Nat. Rev. Neurosci.* **2010**, *11*, 474–489. [CrossRef] [PubMed]
29. Burgin, M.; Schumann, J. Three Levels of the Symbolosphere. *Semiotica* **2006**, *160*, 185–202. [CrossRef]
30. Brooks, R.A. Intelligence without representation. *Artif. Intell.* **1991**, *47*, 139–159. [CrossRef]
31. Kolmogorov, A.N. On the Concept of Algorithm. *Rus. Math. Surv.* **1953**, *8*, 175–176.
32. Codd, E.F. *Cellular Automata*; Academic Press: New York, NY, USA, 1968.
33. Bar-Hillel, Y.; Carnap, R. Semantic Information. *Br. J. Philos. Sci.* **1958**, *4*, 147–157. [CrossRef]
34. Gärdenfors, P. *Conceptual Spaces: The Geometry of Thought*; MIT Press: Cambridge, MA, USA, 2000.
35. Gärdenfors, P. Conceptual Spaces as a Framework for Knowledge Representation. *Mind Matter* **2004**, *2*, 9–27.
36. Gärdenfors, P. Cognitive semantics and image schemas with embodied forces. In *Embodiment in Cognition and Culture*; Krois, J.M., Rosengren, M., Steidele, A., Westerkamp, D., Eds.; Benjamins: Amsterdam, The Netherlands, 2007; pp. 57–76.
37. Shreider, Y.A. On Semantic Aspects of Information Theory. *Inform. Cybern.* **1967**, 15–47. (In Russian)
38. Burgin, M.; Díaz-Nafría, J.M. Introduction to the Mathematical Theory of Knowledge Conceptualization: Conceptual Systems and Structures. In Proceedings of the Second International Conference on Applied Informatics (ICAI 2019), Communications in Computer and Information Science Book Series (CCIS, Volume 1051), Madrid, Spain, 7–9 November 2019; pp. 469–482.
39. Dodig-Crnkovic, G.; Giovagnoli, R. Computing Nature—A Network of Networks of Concurrent Information Processes. In *Computing Nature*; Springer: Heidelberg, Germany, 2013; pp. 1–22.
40. Dodig-Crnkovic, G. Dynamics of Information as Natural Computation. *Information* **2011**, *2*, 460–477. [CrossRef]

41. Burgin, M.; Dodig-Crnkovic, G. From the Closed Classical Algorithmic Universe to an Open World of Algorithmic Constellations. In *Computing Nature*; SAPERE Book Series; Springer: Heidelberg, Germany, 2013; pp. 241–253, arXiv:1211.4547.
42. Pfeifer, R.; Bongard, J. *How the Body Shapes the Way We Think: A New View of Intelligence*; MIT Press: Cambridge, UK, 2006.
43. Bohan Broderick, P. On Communication and Computation. *Minds Mach.* **2004**, *14*, 1–19. [CrossRef]
44. Dodig-Crnkovic, G. *Investigations into Information Semantics and Ethics of Computing*; Mälardalen University Press: Västerås, Sweden, 2006.
45. Burgin, M.; Dodig-Crnkovic, G. Information and Computation—Omnipresent and Pervasive. In *Information and Computation*; World Scientific: New York, NY, USA, 2011; pp. vii–xxxii.
46. Bournez, O. Achilles and the tortoise climbing up the hyper-arithmetical hierarchy. *Theor. Comput. Sci.* **1999**, *210*, 21–71. [CrossRef]
47. Gupta, V.; Jagadeesan, R.; Saraswat, V.A. Computing with Continuous Change. *Sci. Comput. Program.* **1999**, *30*, 3–49. [CrossRef]
48. Shannon, C. Mathematical Theory of the Differential Analyzer. *J. Math. Phys. MIT* **1941**, *20*, 337–354. [CrossRef]
49. Moore, C. Recursion Theory on the Reals and Continuous-time Computation: Real numbers and computers. *Theor. Comput. Sci.* **1996**, *162*, 23–44. [CrossRef]
50. Kelemen, J. On a Possible Future of Computationalism. In Proceedings of the 7th International Symposium of Hungarian Researchers on Computational Intelligence, HUCI'06, Budapest, Hungary, 24–25 November 2006; pp. 51–56.
51. Bera, R.K. *The Amazing World of Quantum Computing*; Springer: Singapore, 2020.
52. Karnani, M.; Pääkkönen, K.; Annila, A. The physical character of information. *Proc. R. Soc. A* **2009**, *465*, 2155–2175. [CrossRef]
53. Ehresmann, A.C. MENS, an Info-Computational Model for (Neuro-)cognitive Systems Capable of Creativity. *Entropy* **2012**, *14*, 1703–1716. [CrossRef]
54. Burgin, M. *Super-Recursive Algorithms*; Springer: New York, NY, USA, 2005.
55. Burgin, M.; Dodig-Crnkovic, G. A Taxonomy of Computation and Information Architecture. In Proceedings of the 2015 European Conference on Software Architecture Workshops, Dubrovnik/Cavtat, Croatia, 7–11 September 2015; ACM: New York, NY, USA. [CrossRef]
56. Burgin, M. Universal Limit Turing Machines. *Not. Russ. Acad. Sci.* **1992**, *325*, 654–658.
57. Burgin, M. Multiple computations and Kolmogorov complexity for such processes. *Not. Acad. Sci. USSR* **1983**, *27*, 793–797.
58. Zenil, H. *A Computable Universe. Understanding Computation & Exploring Nature as Computation*; World Scientific Publishing Company/Imperial College Press: Singapore, 2012.
59. Dodig-Crnkovic, G. Physical Computation as Dynamics of Form that Glues Everything Together. *Information* **2012**, *3*, 204–218, Special Issue on Information: Its Different Modes and Its Relation to Meaning. [CrossRef]
60. Burgin, M. *Measuring Power of Algorithms, Computer Programs, and Information Automata*; Nova Science Publishers: New York, NY, USA, 2010.
61. Burgin, M. Decidability and Universality in the Axiomatic Theory of Computability and Algorithms. *Int. J. Found. Comput. Sci.* **2012**, *23*, 1465–1480. [CrossRef]
62. Dodig-Crnkovic, G.; Burgin, M. Axiomatic Tools versus Constructive approach to Unconventional Algorithms. In Proceedings of the Symposium on Natural Computing/Unconventional Computing and its Philosophical Significance, AISB/IACAP World Congress, Birmingham, UK, 2–6 July 2012.
63. Burgin, M. Information Processing by Structural Machines. In *Theoretical Information Studies: Information in the World*; World Scientific: Singapore, 2020; pp. 323–371.
64. Blackadar, B. *Operator Algebras*; Springer: New York, NY, USA, 2005.
65. Kozen, D. A Completeness Theorem for Kleene Algebras and the Algebra of Regular Events. *Inform. Comput.* **1994**, *110*, 366–390. [CrossRef]


© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

Proceeding Paper

# Structural Connections: Algorithms as Rituals and Rituals as Algorithms †

**Mark Burgin**

Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA; markburg@cs.ucla.edu

† Presented at the International Workshop Habits and Rituals (H&R 2021), IS4SI Summit 2021, Online, 12–19 September 2021.

**Abstract:** In this paper, we analyze relations between rituals and algorithms with the goal of achieving better knowledge of both phenomena as systems of control information about processes. Similarities and dissimilarities between rituals and algorithms are explicated and explored.

**Keywords:** ritual; algorithm; action; operation; performance; orientation; result

**Citation:** Burgin, M. Structural Connections: Algorithms as Rituals and Rituals as Algorithms. *Proceedings* **2022**, *81*, 3. https://doi.org/10.3390/proceedings2022081003

Academic Editors: Raffaela Giovagnoli and Robert Lowe

Published: 9 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Rituals form an important class of social phenomena. As Gino and Norton write, "Despite the absence of a direct causal connection between the ritual and the desired outcome, performing rituals with the intention of producing a certain result appears to be sufficient for that result to come true. While some rituals are unlikely to be effective–knocking on wood will not bring rain–many everyday rituals make a lot of sense and are surprisingly effective." [1].

Examples of rituals are wedding ceremonies or systems of rites in organized religions.

Rituals of various kinds are a feature of almost all known human societies, past or present. They include not only the various worship rites and sacraments of organized religions and cults, but also the rites of passage of certain societies, atonement and purification rites, oaths of allegiance, dedication ceremonies, coronations and presidential inaugurations, marriages and funerals, school "rush" traditions and graduations, club meetings, sports events, Halloween parties, veterans parades, Christmas shopping, and more [2].

At the same time, algorithms have been utilized since the beginning of time. People have been using algorithms permanently. Moreover, algorithms as a technological phenomenon are playing a more and more important role in the contemporary society. First of all, algorithms form a pillar of information technology. Algorithms rule computers, these powerful devices for information processing. Algorithms are so important for computers that even mistakes of computers result mostly from mistakes of algorithms in the form of software. Consequently, the term "algorithm" has become a general scientific and technological concept used in a variety of areas. The huge diversity of algorithms and their mathematical models builds a specific "algorithmic universe."

The comparison of algorithms and rituals shows that there are intrinsic similarities between them. That is why, here, we explicate and explore similarities and dissimilarities between rituals and algorithms. To analyze relations between rituals and algorithms with the goal of attaining better understanding of both phenomena, we consider their definitions from monographs, textbooks, dictionaries, and encyclopedias.

## 2. The Concept Ritual

The word *ritual* originated from the Latin word *ritualis*, which meant a ceremonial rite.

There are several interrelated definitions of rituals:

- A ritual is a ceremony (a system of actions) consisting of a series of actions performed according to a prescribed order.
- A ritual is the prescribed order of performing a ceremony (a system of actions).
- A ritual is the established form for a ceremony (a system of actions).

Rituals are characterized, but not defined, by formalism, traditionalism, invariance, rule-governance, and often by sacral symbolism, especially in the case of religious rituals [3]. People engage in rituals with the intention of achieving a wide set of desired outcomes although, in some cases, people do not know whether they have achieved such outcomes, while there are also cases when it is clear that intended result is not achieved [1].

The purposes of rituals are varied. Rituals can fulfill religious obligations or ideals, satisfy spiritual or emotional needs of the practitioners, strengthen social bonds, provide social and moral education, demonstrate respect or submission, allow one to state one's affiliation, obtain social acceptance or approval for some event—or rituals are sometimes performed just for the pleasure of the ritual itself [2]. At the same time, psychologists found that rituals can be anchors of stability, helping people to achieve greater confidence and a sense of control in anxiety-provoking situations.

We see that the term *ritual* has two meanings:

- A sequence of actions which is repeated by people many times with a definite goal;
- A description of such a sequence of actions controlling their structure and organization.

Rituals involving a group of people are ceremonies, while ceremonies become rituals when they are repeated many times. At the same time, there are rituals performed by an individual, while there are ceremonies performed only once.

It is possible to consider rituals as systems of symbolic actions through which people articulate their attitudes and goals.

## 3. The Concept *Algorithm*

There are different approaches to the definition of algorithm. Being informal, the notion of algorithm allows a variety of interpretations and is modeled by dynamic mathematical structures such as partial recursive functions, Turing machines, or inductive Turing machines. Let us look at some of the suggested definitions.

- An algorithm is an unambiguous (definite) and adequately simple to follow (effective) prescription (e.g., organized set of instructions/rules) for deriving necessary results from given inputs (initial conditions) [4].
- An algorithm is a set of step by step instructions, to be carried out quite mechanically, so as to achieve some desired result [5].
- An algorithm is a well-ordered collection of unambiguous and effectively computable operations that, when executed, produces a result and halts in a finite amount of time [6].

It is possible to perceive that the first two definitions assume that algorithms are aimed at definite results, while the third one demands that algorithms always give the result. In essence, the first definition is the most general, while the third definition is the most restricted. As the result, the most popular model of algorithms, the Turing machine, is an algorithm by the first two definitions but is not an algorithm with respect to the third definition. In a similar way, the system of neuron weights and the activation functions form an algorithm in an artificial neuron network [7] by the first definition but this is not an algorithm with respect to the last two definitions.

## 4. Structural Connections

Comparing two groups of definitions, we can observe that if we treat a ritual as the established form, customary structure, or a prescribed order for a ceremony (a system of actions), which implies rule-governance, it becomes an algorithm for actions by people according to the first two definitions of algorithms. However, it is necessary to keep in

mind that what can be a mechanical action for some people, other people can comprehend as something that demands creativity and/or high intelligence.

In addition, the algorithmic perspective on rituals presents rituals as systems of structural and mental rules/instructions while a process organized according to these rules or instructions becomes performance of a ritual.

Comparing rituals to algorithms, we come to three types of rituals:

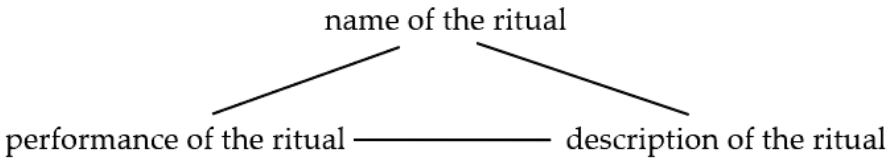
- Potential rituals;
- Accepted rituals;
- Performed rituals.

From the theory of algorithms, we know that an algorithm, performance (execution) of an algorithm, and description of an algorithm are different entities (Burgin, 2005). From the very beginning of the formation of the concept *algorithm*, therewas a clear distinction between an algorithm and its execution. In contrast to this, even now, many people identify algorithms with their descriptions. However, the programming practice persuasively demonstrates that one algorithm, as a rule, has many descriptions, for example, in different programming languages.

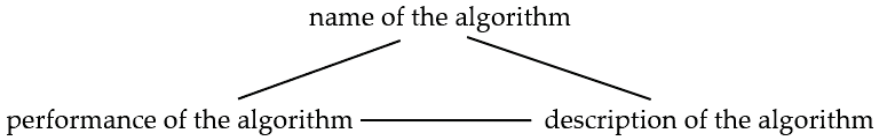
Projecting this situation on the realm of rituals, we see that, investigating rituals, it is useful to make a distinction between a ritual, performance of a ritual, and description of a ritual.

Thus, we come to the following structure.

A ritual can be represented by the following triad:

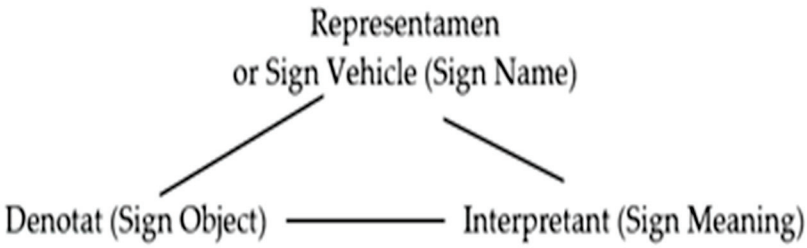



In a similar way, an algorithm can be described by the following triad:




Note that there is a difference between an algorithm and its description—the same algorithm can have different descriptions in different programming languages.

Both triads are special cases of the sign model introduced by Charles Sanders Peirce:




Let us contemplate algorithms from the perspective of rituals.

At first, we consider the goal-associated feature of algorithms comparing it with the same trait of rituals. According to this feature, it is possible to consider the two-dimensional goal-oriented stratification of algorithms.

In the first dimension, which classifies algorithms with respect to their results, we have:

1. Algorithms that always give the assigned/desired result.
2. Algorithms that always give the result, which is not always the assigned/desired result.
3. Algorithms that do not always give the result.

In the second dimension, which classifies algorithms with respect to their interaction with the user, we have:

1. Algorithms that always inform when the result is achieved and when they cannot achieve the result.
2. Algorithms that always inform when the result is achieved.
3. Algorithms that do not always inform when the result is achieved.

This gives us nine groups of algorithms: 1.1, 1.2, . . . , 3.2, 3.3. Taking conventional models of algorithms, we see that:

- Finite automata belong to the group 2.1.
- Turing machines belong to the group 3.2.
- Inductive Turing machines belong to the group 3.3.

The class of all rituals belongs to the group 3.3, that is, they do not always give the result and the performers of rituals do not always know when the result is achieved.

This feature of rituals shows that when people do not want to accept algorithms from the group 3.3, they unnecessarily hamper the abilities of algorithms to help people to solve different problems.

In spite of all similarities, there are essential differences between rituals and algorithms:

1. Each particular performance of a ritual is always finite and, as a rule, bounded in time, while even in one execution, an algorithm can work without stopping.
2. Algorithms include only mechanical steps (actions), while rituals can include some creative actions (steps), for example, preaching in religious rituals.
3. Algorithms, as a rule, are written for artificial devices while rituals are created for people.
4. Any algorithm can exist without even a single execution while a ritual must be performed many times to be an actual ritual. However, some algorithms simply include the necessity of repetition for some part of the algorithm.
5. Inputs of algorithms are well-defined data and knowledge, while often inputs of rituals are not sufficiently well known.

All this allows the treatment of rituals as humanized, locally finite, repetitive algorithms.

In a general case, any ritual is a ceremony, but not any ceremony is a ritual. To be a ritual, a ceremony must be repetitive, i.e., the rules must include repetition, or as it is called in the theory of algorithm, iteration of the whole process.

Thus, any iterative algorithm can be treated as a ritual (for a machine).

## 5. Conclusions

There are features of rituals that can be useful for understanding, design, and utilization of algorithms. For instance, algorithms in which the users are not informed whether results are already obtained can be effective in many situations.

There are features of algorithms that can be useful for understanding, organization, and performance of rituals. For instance, it is necessary to organize rituals in such a way that all their steps can be performed by the participants.

Further analysis of rituals as algorithms would allow us to achieving a better understanding of rituals as important social and psychological mechanisms. At the same time, the additional analysis of algorithms as rituals would contribute to the enhanced differentiation of different forms and types of algorithms in science, information technology, and beyond.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The author declares no conflict of interest.

## References


1. Gino, F.; Norton, M.I. Why Rituals Work: There are Real Benefits to Rituals, Religious or Otherwise. *Scientific American*. 14 May 2013. Available online: https://www.scientificamerican.com/article/why-rituals-work/ (accessed on 27 February 2022).
2. George, G.S.J. The Sociology of Ritual. *Am. Cathol. Sociol. Rev.* **1956**, *17*, 117–130. [CrossRef]
3. Bell, C. *Ritual: Perspectives and Dimensions*; Oxford University Press: New York, NY, USA, 1997.
4. Burgin, M. *Super-Recursive Algorithms*; Springer: New York, NY, USA; Heidelberg/Berlin, Germany, 2005.
5. Barbin, E.; Borowczyk, J.; Chabert, J.-L.; Guillemot, M.; Michel-Pajus, A.; Djebbar, A.; Martzloff, J.-C. *A History of Algorithms: From the Pebble to the Microchip*; Springer: New York, NY, USA; Heidelberg/Berlin, Germany, 1999.
6. Schneider, M.; Gersting, J. *An Invitation to Computer Science*; West Publishing Company: New York, NY, USA, 1995.
7. Hartley, R.; Szu, H. A Comparison of the Computational Power of Neural Networks. In Proceedings of the IEEE Conference on Neural Networks, San Diego, CA, USA, 21–24 June 1987; pp. 17–22.

proceedings

MDPI

*Proceedings*

# Processing Information in the Clouds †

**Mark Burgin [1], Eugene Eberbach [2] and Rao Mikkilineni [3,*]**

[1] Dept. of Math., University of California Los Angeles; Los Angeles, CA 90095, USA; mburgin@math.ucla.edu
[2] Dept. of Eng. and Sci., Rensselaer Polytechnic Institute; Hartford, CT 01609, USA; eeberbach@gmail.com
[3] School of Business, Golden Gate University, San Francisco, CA 94105, USA
* Correspondence: rmikkilineni@ggu.edu

† Conference Morphological, Natural, Analog and Other Unconventional Forms of Computing for Cognition and Intelligence (MORCOM), Berkeley, CA, USA, 2–6 June 2019.

Published: 7 May 2020

**Abstract:** Cloud computing makes the necessary resources available to the appropriate computation to improve scaling, resiliency, and the efficiency of computations. This makes cloud computing a new paradigm for computation by upgrading its artificial intelligence (AI) to a higher order. To explore cloud computing using theoretical tools, we use cloud automata as a new model for computation. Higher-level AI requires infusing features of the human brain into AI systems such as incremental learning all the time. Consequently, we propose computational models that exhibit incremental learning without stopping (sentience). These features are inherent in reflexive Turing machines, inductive Turing machines, and limit Turing machines.

**Keywords:** sentience; artificial intelligence; knowledge structure; oracle; structural machines

## 1. Introduction

When Eugene Wigner discussed the role of mathematics in physical theories, he emphasized the unreasonable effectiveness of mathematics in the natural sciences. He wrote:

> "*The first point is that mathematical concepts turn up in entirely unexpected connections. Moreover, they often permit an unexpectedly close and accurate description of the phenomena in these connections. Secondly, just because of this circumstance, and because we do not understand the reasons of their usefulness, we cannot know whether a theory formulated in terms of mathematical concepts is uniquely appropriate*." [1]

In this paper, we examine a few new theoretical insights based on advanced mathematical theories, and their application to understanding the information processing structures in cloud computing. The term "cloud computing" refers to a variety of Internet-based computing services. The difference between cloud-based and traditional software is that the cloud provides elastic, on-demand computing resources for executing information processing structures. Most cloud computing services are accessed through a web browser or a dedicated mobile app or through a browser on a smartphone or tablet. Therefore, cloud services do not require users to have sophisticated computers that can run specialized software. Despite its popularity and rich applications, cloud computing is not a de facto choice. Its formal model(s), and therefore, some properties of cloud computing, cannot be properly studied to exploit its full richness. In this paper, we discuss the ability of the cloud automata model [2] to harness the power of distributed computing in clouds. These results are new and original, and by stressing the emergence of cloud computing as a new computational paradigm, it is possible to use cloud computing for problems that either cannot be solved at all or not to be solved effectively by traditional computers.

Cloud automata are constructed using the mathematical theory of Oracles [3], which include Oracles of Turing machines as its special case. We develop a hierarchical approach based on Oracles [4–7] with different ranks, which encompass Oracle AI [8] as a particular case. In the context of the named-set approach, we describe an implementation of a high-performance edge cloud using hierarchical name-oriented networking and Oracle AI-based orchestration. We demonstrate how cloud automata with a control overlay allows for microservice network provisioning, monitoring, and reconfiguration to address non-deterministic fluctuations affecting their behavior without interrupting the overall evolution of computation. A high-level AI requires infusing features that mimic human functioning into AI systems. One of the central features is that humans learn all the time and the learning is incremental. Consequently, for AI, we need to use computational models, which reflect incremental learning without stopping (sentience). This means that conventional models of computation, such as Turing machines, are insufficient for AI—which needs more advanced models, such as inductive Turing machines [9]. In our model, we also use the Oracle AI agents as hierarchical cognizing agents. These assist in modeling, monitoring and executing computational structures that process information, and manage their evolution even in the face of non-deterministic fluctuations in the availability or demand of resource.

## 2. Oracles in Clouds and Cloud Automata

Many scientists are convinced that computations going beyond Turing machines are possible, while contemporary and future computers can compute, and will compute, beyond the Turing limit using—for example, reactive programs, such as operating systems or client/server Internet computing [10,11]. Cloud computing also belongs to this category, and cloud automata are aimed at defining more powerful models of computation and computing networks. Current cloud computing model is mainly addressing how to make the right resources available to the proper computation to improve scaling, resiliency and efficiency of the computation. It is possible to define three stages of networking a general and cloud technology, in particular:

1. The *address-oriented networking* when the connection to systems and resources is based on their addresses in the network;
2. The *name-oriented networking* when the connection to systems and resources is based on their names;
3. The *hierarchical name-oriented networking* when the connection to systems and resources is organized through connection to the relevant host, which provides necessary systems and resources by their names.

Currently, all three types of cloud computing and networking coexist using the Internet as their communication space. However, today, the Internet architecture supports and is exclusively oriented only at address-oriented networking. It is possible to use two approaches to achieve the transition from address-oriented networking to name-oriented networking in clouds. One of them demands a change in the basic Internet architecture. The other approach does not demand such big transformations, because it utilizes network agents, such as described in *DIME* (Distributed Intelligent Managed Element) *Network Architecture - DNA* cloud [2,5,6] or *middleware CORBA* (Common Object Request Broker Architecture) or *Java RMI* (Java Remote Method Invocation) [10,11], to convert names to addresses. Sets and operations are named using an analytical description of these procedures [12].

Cloud automata are designed using the hierarchical approach in artificial intelligence, based on the mathematical theory of oracles [3,4]. This approach includes Oracle AI [8] as a special case providing new tools for the exploration of artificial intelligence in general and Oracle artificial intelligence in particular. In computer science, oracles appeared in the first half of the 20$^{th}$ century when Turing introduced into computing, an oracle machine or o-machine. It is a Turing a-machine that pauses its computation at state "o", while to complete its calculation, it "awaits the decision" of "the oracle"—an unspecified entity "apart from saying that it cannot be a machine" [13]. Later, the concept of an oracle was utilized by different researchers. In his mathematical theory of Oracles [4],

Burgin describes an organizational hierarchy of Oracles, which consists of three levels of agent oracles:

1. Controlled agent oracles;
2. Regulated agent oracles;
3. Autonomous agent oracles.

A hierarchy of oracles is a useful structure for AI when machines (systems) are organized in layers so that a machine (system) in the layer $n$ can be an oracle only for machines (systems) in the layer $n - 1$ and lower. Naturally, when such a hierarchy exists, machines (systems) in the lowest layer are not oracles for machines at higher levels in this hierarchy. This structure ascribes ranks to oracles in the following way. An oracle of the rank 1 is an oracle for one or several basic systems [3,4]. Assuming that functioning of machines is organized according to the hierarchical structure, an oracle of the rank $n$ is an oracle for one or several oracles of the rank $n - 1$. Note that, in general, it is possible that a machine A plays the role of an oracle for a machine B in some situations, while in other situations the machine B plays the role of an oracle for a machine A. Hierarchic intelligence forms a hierarchy of oracles over the base information processing system. The oracles can be used as cognizing agents that configure, monitor and provide addressing, alerting, mediation and supervision functions for down-stream information processing systems based on global knowledge acquired from a peer or superior oracles. To build a high-level artificial intelligence, it would be useful to incorporate certain features of the human brain. One of the central features is that the brain learns all the time and learning is incremental. Consequently, for artificial intelligence, we need to use computational models, which reflect incremental learning without stopping (sentient systems). Oracles as cognizing agents coupled with knowledge structures [14] and structural machines [15] provide a framework to implement such sentient machines.

In this context, a *cloud automaton* is a three-component structure ($M$, $C$, $O$), in which $M$is the total user (basic) automaton, $C$ is the communication space and $O$ is the Oracle $O$ in a Cloud.

The *total user automaton* is a distributed system, which consists of *unit user automata*, each of which is utilized for computation or more general, for solving problems by the cloud user. The most convenient and flexible mathematical model of a user automaton is a grid automaton [9].

In a cloud automaton, the *Oracle O* can serve as a provider of information, services and/or virtual systems, e.g., computing and storage facilities, virtual software and hardware, data, and infrastructures, to the user automaton via the *communication space* [16].

When the unit user automata are either recursive, such as Turing machines or Minsky machines [12], or sub-recursive automata, such as finite automata, then a super-recursive automaton, such as an inductive Turing machine or inductive cellular automaton [17], can be the Oracle in a cloud automaton.

The structural machine framework relates knowledge structures (made up of named objects, their relationships and behaviors in the form of algorithms) and their processing to gleam information using computing processes [15]. The knowledge structures and their evolution can be thought of as executable microservices and their evolution in cloud computing. An implementation of non-stop web service using a set of oracles managing a web server, application server and a database on multiple clouds is described in Reference [6]. High availability using auto-failover, auto-scaling, and live migration of application components from one cloud to another, which are separated geographically is demonstrated without disrupting the service transactions. The oracles manage the downstream application components based on policies that are executed by oracle agents.

## 3. Conclusions

The foundation for an information processing entity (IPE) comes from the stored program implementation of the Turing machine. However, IPE networking architecture augments the information processing power by facilitating communication among IPEs and allowing shared resources for computation within a computing machine or a network of these machines. The first phase of information networking used their host (or logical) address (usually an IP address in private

or public networks). A domain name server (DNS) provides the mapping between the location and the IPE. Inside each host, multiple IPEs are distinguished by various ports. During the past five+ decades, IP networking using IP addresses has been the dominant protocol providing the information networking services establishing links between the IPE nodes.

More recently, the focus has been shifting to IPEs as microservices that provide specific business (or other) functions, with each acting as an "active" node—hiding the details of implementation (which itself could be a subnetwork of IPEs). This provides a new class of hierarchical name-oriented microservice networking, where each node executes a specific behavior based on the input. A control overlay allows for microservice network provisioning, monitoring, and reconfiguration, as required to address fluctuations in their behavior.

In essence, such a hierarchical named network of IPEs allows the composition of micro-services to create a managed process workflow by enabling dynamic configuration and reconfiguration of the micro-service network. Examples of this are Kubernetes orchestrated microservice networks and DIME (distributed intelligent managed element) networks, as discussed in the literature. Both these examples use hierarchical named micro-service networks (and subnetworks) to provision, monitor, and control information processing structures and their resources, while addressing non-deterministic fluctuations in the demand or availability of needed computing resources. In this context, cloud automata provide theoretical tools for modeling, exploration, and improvement of information processing in clouds.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Wigner, E. The Unreasonable Effectiveness of Mathematics in the Natural Sciences. *Commun. Pure Appl. Math*. **1960**, *13*, 1–14.
2. Burgin, M.; Eberbach, E.; Mikkilineni, R. Cloud Computing and Cloud Automata as A New Paradigm for Computation. *Comput. Rev. J.* **2019**, *4*, 113–134.
3. Burgin, M. Inaccessible Information and the Mathematical Theory of Oracles. In *Information Studies and the Quest for Transdisciplinarity: Unity through Diversity*; World Scientific: New York, NY, USA; London, UK, Singapore, 2017; pp. 59–114.
4. Burgin, M. On the power of oracles in the context of hierarchical intelligence. *J. Artif. Intell. Res. Adv.* **2016**, *3*, 6–17.
5. Mikkilineni, R.; Morana, G.; Burgin, M. Oracles in Software Networks: A New Scientific and Technological Approach to Designing Self-Managing Distributed Computing Processes. In Proceedings of the 2015 European Conference on Software Architecture Workshops, Dubrovnik/Cavtat, Croatia, 7–11 September 2015; ACM, pp. 11:1–11:8.
6. Burgin, M.; Mikkilineni, R. Cloud computing based on agent technology, super -recursive algorithms, and DNA. *Int. J. Grid Util. Comput.* **2018**, *9*, 193–204.
7. Mikkilineni, R.; Morana, G. Post-Turing Computing, Hierarchical Named Networks and a New Class of Edge Computing. In Proceedings of the 2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Napoli, Italy, 12–14 June 2019, pp. 82–87.
8. Armstrong, S.; Sandberg, A.; Bostrom, N. Thinking Inside the Box: Controlling and Using an Oracle AI. *Minds Mach.* **2012**, *22*, 299–324.
9. Burgin, M. *Superecursive Algorithms*; Springer: New York, NY, USA, 2005.
10. Tanenbaum, A.; Bos, H. *Modern Operating Systems*, 4th ed.; Pearson: London, UK, 2015.
11. Tanenbaum, A.; Wetherall, D. *Computer Networks*, 5th ed.; Pearson: London, UK, 2011.
12. Burgin, M. *Superecursive Algorithms*; NovaScience: New York, NY, USA, 2011.
13. Turing, A.M. Systems of logic defined by ordinals. *Proc. Lond. Math. Soc. Ser. 2*, **1939,** *45*, 161–228.
14. Burgin, M. *Theory of Knowledge*: *Structures and Processes*; World Scientific Books: Singapore,2016.
15. Burgin, M.; Adamatzky, A. Structural Machines as a Mathematical Model of Biological and Chemical Computers. *Theory Appl. Math. Comput. Sci.* **2017**, *7*, 1–30.

16. Burgin, M.; Liu, D.; Karplus, W. The Problem of Time Scales in Computer Visualization. In Proceedings of the International Conference on Computational Science, San Francisco, CA, USA, 28–30 May 2001; Lecture Notes in Computer Science; Volume 2074, part II, pp. 728–737.
17. Burgin, M. Inductive Cellular Automata. *Int. J. Data Struct. Algorithms* **2015**, *1*, 1–9.

© 2020 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

*Proceedings*

# Structural Machines as Unconventional Knowledge Processors †

**Rao Mikkilineni [1,*] and Mark Burgin [2]**

[1] Ageno School of Business, Golden Gate University, San Francisco, CA 94105, USA
[2] Department of Mathematics, University of California Los Angeles, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu
* Correspondence: rmikkilineni@ggu.edu
† Conference Morphological, Natural, Analog and Other Unconventional Forms of Computing for Cognition and Intelligence (MORCOM), Berkeley, CA, USA, 2–6 June 2019.

Published: 7 May 2020

**Abstract:** Knowledge systems often have very sophisticated structures depicting cognitive and structural entities. For instance, representation of knowledge in the form of a text involves the structure of this text. This structure is represented by a hypertext, which is networks consisting of linguistic objects, such as words, phrases and sentences, with diverse links connecting them. Current computational machines and automata such as Turing machines process information in the form of symbol sequences. Here we discuss based the methods of structural machines achieving higher flexibility and efficiency of information processing in comparison with regular models of computation. Being structurally universal abstract automata, structural machines allow working directly with knowledge structures formed by knowledge objects and connections between them.

**Keywords:** cognition; named set; knowledge structure; theory of oracles; structural machine

## 1. Introduction

Many philosophers, mathematicians and scientists have attempted to define knowledge and its relationship to information and its processing. Plato was perhaps the first to articulate that knowledge is always about something—some object or domain or mental concept [1]. In order to discern a knowledge object or domain, we have to name it. A name may be a label, number, idea, text, process, and even a physical object of a relevant nature. The named objects may be composed into knowledge structures which may have interobject and intraobject relationships and associated behaviors that may cause changes to their state, form or content. The role of information-processing structures is to discern the relationships and behaviors and evolve the state, form or content accordingly. Information in the strict sense includes a capacity to change structural elements in knowledge systems [2].

Knowledge systems exist in the world of structures, which belongs to the well-known existential triad of the world together with the physical world and the mental world. The knowledge objects or domains in the physical or mental world manifest themselves as structures. In the physical world, matter (containing physical, chemical and biological structures) contains energy. Similarly, structures contain information. Thus, "information is related to structures as energy is related to matter" [3]. Figure 1 shows the relationships between the physical world, the mental world, the digital world and the world of structures. Note that it is possible to treat the digital world as the mental world of digital devices—computers, computer networks and so on.

In the case of Turing machine implementation, the knowledge systems are represented by symbolic data structures and a central processor unit operates on them to process information. The

machine acts as a cognitive apparatus [4] with locality and the ability to form information processing structures where information flows from one apparatus to another with a velocity defined by the medium. Physical manifestation of information processing functions and provides structures.

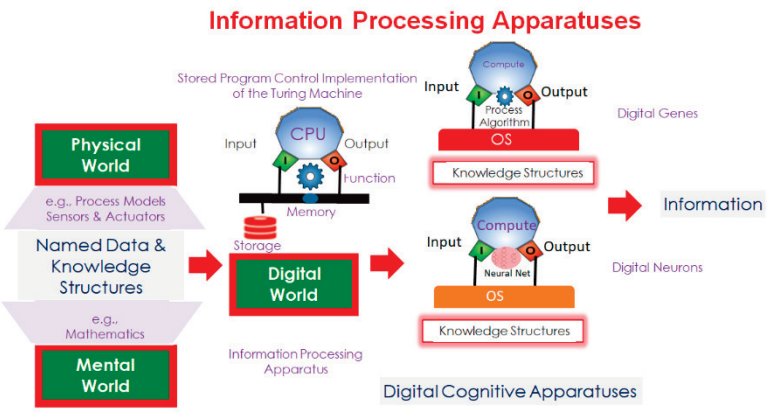



**Figure 1.** The world of structures and digital information processing.

A physical framework to address fluctuations and nondeterministic interactions of these processes using the concepts of entropy, energy and phase transitions through reconfiguration of structures. Any algorithm that can be specified is transformed into an executable function using CPU and Memory. Functions operate on data structures representing domain knowledge and the computation evolves their current state to a new state. As long as there are enough resources (CPU and memory), the computation will continue as encoded in the algorithm. This is equivalent to a digital gene (representing well-specified executable process evolutions) assisting the execution of business processes. Cognition comes from the ability to encode knowledge structures and their processing to transform them from one state to another just as genes in biology do.

It is interesting to note that the Turing computable functions allow representation of algorithms that define neural networks which are used to model processes that are usually treated as nonalgorithmic, such as voice recognition, video processing, etc. Here, cognition comes from the ability to encode how to mimic neural networks in the brain modeling and processing information just as neurons in biology do. The digital neuron in Figure 1 executes cognitive processes that are not specified in the form of conventional algorithms. It also imitates biological systems collaborating with an apparatus that processes algorithms analogous to a gene where knowledge is encoded in DNA structures.

By its organization, a standard Turing machine processes words letter by letter in the sequential mode. Thus, to work with knowledge using the standard Turing machines, it is necessary in advance to present knowledge as linear structures. This essentially decreases efficiency of computation creating unnecessary barriers for knowledge processing. To increase efficiency, Turing machines with many tapes and with multidimensional tapes were introduced. They perform computations with more advanced data structures such as arrays and collections of arrays. Nevertheless, this also was not enough, and to improve efficiency and allow processing of not only symbols but also links between them, more advanced automata, such as Kolmogorov algorithms storage modification machines and relational machines were developed. However, all these relations define only structures of the first order, while knowledge structures can have much higher orders. Structural machines eliminate this restriction and further advance efficiency [5]. In the next section, we discuss knowledge processing by structural machines. Structural machines work with knowledge structures of arbitrary order transforming, not only elements of these structures or the content of these elements as conventional models of computation do, but also relations of different orders in the processed structures. This allows achieving higher flexibility and efficiency in comparison with regular models of computation, including both conventional and unconventional computing systems. Structural machines can also simulate such advanced computational automata such as Kolmogorov algorithms,

limit Turing machines, storage modification machines, relational machines and other models of computation. Being structurally universal abstract automata, structural machines work directly with knowledge structures, molecular and atomic structures, with structures studied and utilized in the topological quantum field theory (TQFT), and with structures of quantum information such as qubits.

## 2. Named Sets, Knowledge Structures, Structural Machines, the Theory of Oracles and Digital Information Processing Structures

Structural relationships exist between data, which are entities observed in the physical world or conceived in the mental world. These structures define the knowledge of them in terms of their properties such as attributes, relationships and dynamics of their interaction. Information processing structures organize evolution of knowledge structures by an overlay of cognitive knowledge structures, which model, monitor and manage the evolution of the information processing system.

The most fundamental structure is called a *fundamental triad* or a *named set* [6]. It has the following visual representation shown in Figure 2:

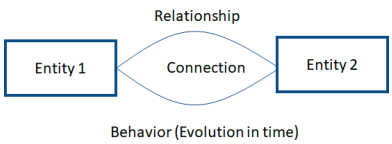



**Figure 2.** Visual representation of a fundamental triad (named set).

An entity or object with its own name is connected to another entity or object with its own name. The connection, which itself has a name, depicts the knowledge of the relationship and the behavioral evolution when a state change occurs in either object.

A knowledge structure [1,7] is composed of related fundamental triads, and any state change causes behavioral evolution based on the connections. The long and short of the theory of knowledge is that the attributes of objects in the form of data and the intrinsic, and ascribed knowledge of these objects in the form of algorithms and processes, make up the foundational blocks for information processing. Information processing structures utilize knowledge in the form of algorithms and processes that transform one state (determined by a set of data) of the object to another one with a specific intent. Information structures and their evolution using knowledge and data determine the flow of information. Living organisms have found a way not only to elaborate the knowledge of the physical objects, but also to create information processing structures that assist them in executing state changes.

The structural machine framework describes a process which allows information processing through transformation of knowledge structures. It involves a control device that configures and executes information processing operations on knowledge structures and manages the operations throughout its lifecycle using a processor. The processor uses the knowledge structures as input and delivers the processed information as knowledge structures in the output space.

In the special case where input knowledge structure and the output knowledge structures are words (symbols) and the process to be executed is an algorithm (a sequence of operations), then the structural machine becomes a Turing machine. The control is outside the Turing machine which provides the algorithm to be executed, assuring that the processor has the right resources to perform the operations and judge whether the computation is performed as expected. In essence, the functional requirements of the system that is under consideration such as business logic, sensor and actuator monitoring and control (the computed) etc., are specified as algorithms and are executed by the processor transforming the knowledge structures from the input space to the output space. Figure 3 shows the structural machine framework.

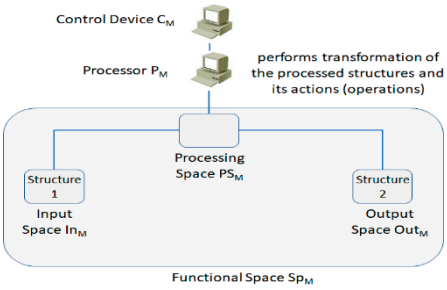



**Figure 3.** Structural machine framework for information processing structures.

It is important to observe that when the controller is an operator, the processor is a stored program implementation of a Turing machine, the processing space is the memory, and the knowledge structures are symbolic data structures, we obtain the current state-of-the-art information-processing structure.

A knowledge structure composed of the named sets is shown in Figure 4.

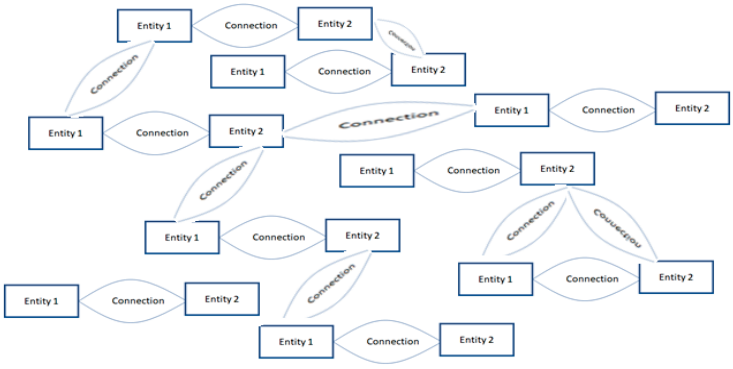



**Figure 4.** A network of knowledge structures showing the state vectors.

The knowledge structure depicted here is a graph that captures both cognitive and structural domain knowledge. The picture captures hierarchical, decoupled and concurrent structures whose evolutionary behavior is captured by the fundamental triads or named sets. The knowledge systems go beyond the knowledge captured by taxonomies or ontologies emerging, by adding the connection and depicting the relationships and evolutionary behaviors.

The knowledge structures composed of named sets are processed using the structural machines. The processor space deciphers the connection relationships and behaviors to evolve the state. The controller allows the implementation of a cognizing agent overlay that manages the downstream processors and associated knowledge-structure evolution. The cognizing agents shown in Figure 5 are defined using the mathematical theory of oracles proposed and developed by Mark Burgin [8]. According to this theory, an agent system can work as an oracle, collecting information for the basic machines or for a more complex information processing devices, such as a network or computer cluster. For instance, in the theory of superrecursive algorithms and computation, an important type of oracle contains information on whether a Turing machine halts given definite input or not. As we know in this case, even simple inductive Turing machines are able to compute this and other information that is incomputable by Turing machines [9]. Consequently, it is possible to use inductive Turing machines as oracles for bringing information that is incomputable by Turing machines and other recursive algorithms information to various computing devices such as Turing machines or neural networks.

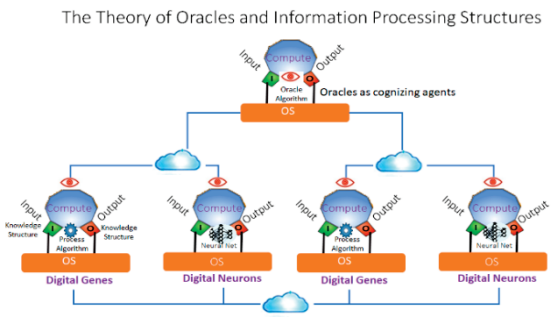



**Figure 5.** Oracles as cognizing agents managing the execution and evolution of knowledge structures.

The utilization of higher-order inductive Turing machines supplies Turing machines and other computing devices with even more powerful oracles, making the whole schema of oracle machines, for which the oracle Turing machine is only one particular case, a constructive device. When the agent that functions as an oracle collects information for the basic machine by search, it is a mining agent. When this agent obtains information for the basic machine by computation, it is a designing agent. The hierarchy of computational models determines a hierarchy of knowledge mining and designing agents. In this hierarchy, there are different types of agents with specific behavior. For instance, rigid agents transform only external information using the same program (algorithm) as the basic machine. At the same time, reflexive agents change their own program (algorithm) to achieve better results.

## 3. Conclusions

The introduction of structural machines has three objectives:

1. the elaboration of a more adequate model of contemporary computers, which work with and are directed by high-level programming languages;
2. the formation of mathematical tools for transition from computing with data to knowledge processing in the framework of artificial intelligence (AI);
3. the construction of a theoretical model for natural computation.

Structural machines can have many unit processors, which perform parallel and concurrent computations. There are many examples of structural machines.

1. Any individual can be treated as a structural machine with the brain as its processor and the functional space. Note that usually, people work in the inductive mode.
2. A group of people is a structural machine with several unit processors and the functional space, which comprises the brains of these individuals.
3. In essence, an advanced computer is also a structural machine.
4. A computer network, such as the World Wide Web, is a structural machine.
5. An individual with their computers is also a structural machine with several unit processors, which include utilized computers and the brain of the individual.
6. A cell phone is a structural machine.
7. An individual with their computers and cell phones is also a structural machine with several unit processors, which include utilized computers, cell phones and the brain of the individual.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. *Theory of Knowledge*: *Structures and Processes*; World Scientific Books: Singapore, 2016.
2. Burgin, M. Information: Problems, Paradoxes, and Solutions. *Triple C* **2003**, *1*, 53–70.
3. Burgin, M. Information in the structure of the world. *Int. J. Inf. Theor. Appl.* **2011**, *18*, 16–32.

4. Mikkilineni, R.; Morana, G. Post-Turing Computing, Hierarchical Named Networks and a New Class of Edge Computing. In Proceedings of the 2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Napoli, Italy, 12–14 June 2019; pp. 82–87.
5. Burgin, M.; Adamatzky, A. Structural Machines as a Mathematical Model of Biological and Chemical Computers. *Theory Appl. Math. Comput. Sci.* **2017**, *7*, 1–30.
6. Burgin, M. *Theory of Named Sets*; Nova Science Publisher Inc.: New York, NY, USA, 2011.
7. Burgin, M. Data, Information and Knowledge. *Information* **2004**, *7*, 47–57.
8. Burgin, M. Inaccessible Information and the Mathematical Theory of Oracles. In *Information Studies and the Quest for Transdisciplinarity: Unity through Diversity*; World Scientific: New York, NY, USA; London, UK; Singapore, 2017; pp. 59–114.
9. Burgin, M. *Superrecursive Algorithms*; Springer: New York, NY, USA, 2005.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

*Proceedings*

# Modeling Computing Devices and Processes by Information Operators †

**Mark Burgin [1,*] and Gordana Dodig-Crnkovic [2]**

[1] Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA

[2] Department of Computer Science and Engineering, Chalmers University of Technology, 41296 Gothenburg, Sweden; gordana.dodig-crnkovic@chalmers.se

* Correspondence: markburg@cs.ucla.edu

† Conference Theoretical Information Studies (TIS), Berkeley, CA, USA, 2–6 June 2019.

Published: 11 May 2020

**Abstract:** The concept of operator is exceedingly important in many areas as a tool of theoretical studies and practical applications. Here, we introduce the operator theory of computing, opening new opportunities for the exploration of computing devices, networks, and processes. In particular, the operator approach allows for the solving of many computing problems in a more general context of operating spaces. In addition, operator representation of computing devices and their networks allows for the construction of a variety of operator compositions and the development of new schemas of computation as well as network and computer architectures using operations with operators. Besides, operator representation allows for the efficient application of the axiomatic technique for the investigation of computation.

**Keywords:** computation; information; computing device; operator; network; computer architecture; composition

## 1. Introduction

The concept of an operator in mathematics stands for mapping or transformation from one space (set of elements) to another space. Linear maps are the basic operators on vector spaces. Linear operators (such as differentiation and integration) act on vector spaces of functions. The term operator is also used to denote a mathematical operation (like square root). In computer programming, a set of operators such as arithmetic-, comparison-, and logical-operators are supported by programming languages. The first textbook on operator theory [1] by Stefan Banach was published in 1932.

The concept of operator spans many disciplines as an important tool of theoretical studies and practical applications [2]. For instance, operators have become one of the most important tools in theoretical physics, especially quantum mechanics (see, e.g., [3,4]). In quantum mechanics, operators are represented as matrices, column vectors, and differential equations in an equivalent way. In mathematics, there is operator theory, which studies operators in Hilbert or Banach spaces (see, e.g., [5–7]). Operators are also becoming an important tool in information theory (see [8–11]). Many programming languages use operators. For instance, the programming language Python divides its operators into seven groups: arithmetic operators, assignment operators, comparison operators, logical operators, identity operators, membership operators, and bitwise operators [12]. The programming language Java divides its operators into five groups: arithmetic operators, assignment operators, comparison operators, logical operators, and bitwise operators [13].

Computing devices are information transformers and generators. That is why in this work, we develop operator models of computing devices and study their properties based on the unified operator theory [1].

## 2. Operator Modeling

In the context of this theory, we have the following definitions.

**Definition O1.** *An operator is an object (system) that operates, i.e., performs operations on some objects, systems, or processes, which are called operands of this operator.*

This brings us to the following definition.

**Definition O2.** *An operand is an object, system, or process operated by an operator.*

These definitions show that being an operator or an operand is a role and a characteristic of a system. One and the same system/object can be an operator in some situations and an operand in other situations. In a similar way, a system/object can be an operator with respect to some systems and not an operator with respect to other systems. All operators are systems, but not all systems are operators since subsequent to their formation, some systems can exist in substantial isolation from their environment to all intents and purposes.

Definitions O1 and O2 express the fundamental dyadic relation between operators and their operands, which is actualized in the form of the operator triad:

$$\text{Operator} \xrightarrow{\text{Operation/function}} \text{Operand}$$

The operator triad is a special case of the basic fundamental triad [14]. In the symbolic representation, the operator triad has the form

$$(\text{Op}, \textit{on}, \text{Od})$$

where Op is an operator, *on* is an operation, and Od is an operand.

To construct a general mathematical operator theory in some domain, for example, in the realm of computations, it is necessary to organize the multiplicity of relevant operands in the form of an operating space, i.e., the space that is transformed by an operator.

Different types of operators function in distinct operating spaces. For instance, operators of quantum mechanics use Hilbert spaces as their operating space. Information operators function in information spaces. As there are diverse types of information, representation of them by operators demands different types of information spaces. Based on the general theory of information, we can differentiate syntactic, semantic, pragmatic, algorithmic, cognitive, emotional, and effective information spaces.

Consequently, modeling computing devices by information operators, we can treat them as tools for transformation and generation of distinct kinds of information. Here, we consider syntactic, semantic, and pragmatic information and corresponding information spaces.

In the case of digital computing devices, both physical and abstract, syntactic information spaces are systems of formal, artificial, or natural languages. For instance, taking such an abstract computing device as a Turing machine, we come to the syntactic information space, which consists of all formal languages with the alphabet of this Turing machine. In the case of finite automata, it is possible to utilize syntactic information spaces of three types. A syntactic information space of the first type consists of all symbols from the alphabet of the finite automaton. A syntactic information space of the second type consists of all symbols denoting states of the finite automaton. A syntactic information space of the third type consists of all formal languages with the alphabet of the finite automaton.

Digital computing devices process information in symbolic form, transforming words of some languages. Analog/physical computational processes are based on mapping between physical spaces. Consequently, both physical and abstract, syntactic information spaces are systems of formal, artificial, or natural languages. This implies that treating these devices as operators, we encounter operands of five types: (physical) data, separate symbols, words, texts, and languages and families of languages.

For instance, taking such an abstract computing device as a Turing machine, we come to the syntactic operating space, which consists of all formal languages with the alphabet of this Turing

machine. At the same time, it is possible to take the space of all strings in some alphabet as the syntactic operating space of a Turing machine.

In the case of finite automata, it is possible to utilize syntactic information spaces of four types. A syntactic information space of the first type consists of all symbols from the alphabet of the finite automaton. A syntactic information space of the second type consists of all symbols denoting states of the finite automaton. A syntactic information space of the third type consists of all words from the formal languages with the alphabet of the finite automaton. A syntactic information space of the fourth type consists of all formal languages with the alphabet of the finite automaton.

The majority of abstract automata (computing devices) work with linear (i.e., one-dimensional) languages. However, there are also abstract automata (computing devices) that work with more sophisticated structures. For instance, Kolmogorov algorithms work with arbitrary graphs [15], Turing machines work with two-dimensional tapes, two-dimensional cellular automata work with two-dimensional structures, while structural machines work with arbitrary structures [16].

As the result, operators representing different abstract automata (computing devices) have different syntactic operating spaces.

Operators that represent Kolmogorov algorithms are Kolmogorov computation operators, the syntactic operating space of which is a collection of formal graph languages, i.e., languages words of which are graphs, while the definability domain consists of enumerable (recursively computable) graph languages [15].

Operators that represent two-dimensional cellular automata [17] are two-dimensional cellular computation operators, the syntactic operating space of which is the collection of two-dimensional array languages, i.e., whose language words are two-dimensional arrays, while the definability domain consists of all enumerable (recursively computable) two-dimensional array languages.

Operators that represent structural machines are structural computation operators, the syntactic operating space of which is the collection of structural languages, i.e., whose language words are structures, while the definability domain consists of all enumerable (recursively computable) structural languages.

Operators that represent Turing machines with one-dimensional tapes are one-dimensional Turing computation operators, the syntactic operating space of which is the collection of formal languages, while the definability domain consists of all recursively enumerable (recursively computable) languages.

Operators that represent Turing machines with two-dimensional tapes are two-dimensional Turing computation operators, the syntactic operating space of which is the collection of two-dimensional formal languages, while the definability domain consists of all recursively enumerable (recursively computable) two-dimensional languages.

Semantic information spaces can also have a different nature. For instance, in the semantic information theory of Bar–Hillel and Carnap, a semantic information space consists of possible worlds [18]. Often a semantic information space is a conceptual space, as in reference [19]. In the theory of epistemic information, a semantic information space is a conceptual space [9,10]. In the semantic information theory of Shreider, a semantic information space is a thesaurus as a system of texts and semantic relations between these texts [20]. Utilization of semantic information spaces in modeling of computing devices by information operators allows for the studying of semantic aspects of computation, computing systems, and networks.

Natural computation is physical computation performed through the dynamics of physical bodies [21], and involves information spaces of physical states (morphologies) of the systems undergoing spatiotemporal transformations [22–25]. In robotics, specific morphological computation that offloads computational tasks of control to the natural physical behavior and morphology of the robot body itself is presented, for example, by the authors of [26,27]. The conceptual model connecting subsymbolic (physical) and symbolic (formal language-based) levels of computation is given by Ehresmann [28].

## 3. Conclusions

To conclude, it is necessary to remark that operator modeling of computing devices—both abstract and physical—allows for the application of various constructions and results from the mathematical operator theory to automata, algorithms, and computations, presenting the following possibilities.

*First possibility*: Operator representation of computing devices allows for the formulating and solving of many computing device and network problems in a more general context in operating spaces of operators.

*Second possibility*: Operator representation of computing devices allows for the construction of a variety of operator compositions (operations) and the development of new schemas of computation as well as new network and computer architectures using operations with (composition of) operators.

*Third possibility*: Operator representation of computing devices allows for the efficient application of the axiomatic technique for the investigation of algorithms and computations.

Operator modeling of computing devices also presents various problems for further research. For instance, an interesting problem is to build a general mathematical description of operator spaces and to study operators in these spaces. One more interesting problem is the construction and exploration of operator algebras with nonstandard operations. There are different models of quantum computing involving dissimilar quantum theories [29]. Thus, an important task is to develop a unified operator theory of quantum computing.

**Funding:** One of the authors (GDC) has her research funded by Swedish Research Council grant.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Banach, S. *Théorie des Opérations Linéaires;* Monografie Matematyczne: Warszawa, Poland, 1932; Volume 1.
2. Burgin, M.; Brenner, J. Operators in Nature, Science, Technology, and Society: Mathematical, Logical, and Philosophical Issues. *Philosophies* **2017**, *2*, 21, doi:10.3390/philosophies2030021.
3. Dirac, P. *The Principles of Quantum Mechanics*; Oxford University Press: Oxford, UK, 1930.
4. Exner, P.; Havlíček, M. *Hilbert Space Operators in Quantum Physics*; Springer: New York, NY, USA, 2008.
5. Brown, A.; Pearcy, C. *Introduction to Operator Theory I: Elements of Functional Analysis*; Springer Verlag: New York, NY, USA; Heidelberg/Berlin, Germany, 1977.
6. Ball, J.A.; Bolotnikov, V.; Helton, J.W.; Rodman, L. (Eds.) *Topics in Operator Theory (Operator Theory: Advances and Applications)*; Birkhäuser Verlag: Basel, Switzerland, 2010.
7. Burgin, M. *Semitopological Vector Spaces: Hypernorms, Hyperseminorms and Operators*; Apple Academic Press: Toronto, NA, Canada, 2017.
8. Harris, Z. *A Theory of Language and Information: A Mathematical Approach*; Oxford University Press: Oxford, UK, 1991.
9. Burgin, M. Epistemic Information in Stratified M-Spaces. *Information* **2011**, *2*, 697–726.
10. Burgin, M. Weighted E-Spaces and Epistemic Information Operators. *Information* **2014**, *5*, 357–388.
11. Brenner, J.; Burgin, M. Information as a Natural and Social Operator. *Inf. Theor. Appl.* **2011**, *18*, 33–49.
12. Downey, A.B. *How to Think Like a Computer Scientist: Learning with Python*; Green Tea Press: Needham, MA, USA, 2002.
13. Eckel, B. *Thinking in Java*; Prentice Hall: Upper Saddle River, NJ, USA, 2006.
14. Burgin, M. *Theory of Named Sets*; Nova Science Publishers: New York, NY, USA, 2011.
15. Kolmogorov, A.N. On the Concept of Algorithm. *Russian Math. Surv.* **1953**, *8*, 175–176.
16. Burgin, M.; Adamatzky, A. Structural machines and slime mold computation. *Int. J. Gener. Syst.* **2017**, *45*, 201–224.
17. Codd, E.F. *Cellular Automata*; Academic: New York, NY, USA, 1968.
18. Bar-Hillel, Y.; Carnap, R. Semantic Information. *Brit. J. Phil. Sci.* **1958**, *4*, 147–157.
19. Gärdenfors, P. *Conceptual Spaces: The Geometry of Thought*; MIT Press: Cambridge, MA, USA, 2000.
20. Shreider, Y.A. On Semantic Aspects of Information Theory. *Inf. Cybern. (Moscow, Radio)* **1967**, *10*, 15–47. (in Russian).
21. Rozenberg, G.; Kari, L. The many facets of natural computing. *Commun. ACM* **2008**, *51*, 72–83.

22. Dodig-Crnkovic, G. Knowledge Generation as Natural Computation. *J. Syst. Cybern. Inf.* **2008**, *6*, 12–16.
23. Dodig-Crnkovic, G. Dynamics of Information as Natural Computation. *Information* **2011**, *2*, 460–477.
24. Dodig-Crnkovic, G. Nature as a Network of Morphological Infocomputational Processes for Cognitive Agents. *Eur. Phys. J.* **2017**, *226*, 181–195, doi:10.1140/epjst/e2016-60362-9.
25. Dodig-Crnkovic, G. Cognition as Embodied Morphological Computation. *Phil. Theory Artif. Intell.* **2017**, 19–23, doi:10.1007/978-3-319-96448-5.
26. Paul, C. Morphology and Computation. In Proceedings of the International Conference on the Simulation of Adaptive Behavior Los Angeles, CA, USA, 16–18 September 2004; pp. 33–38.
27. Pfeifer, R.; Iida, F. Morphological computation: Connecting body, brain and environment. *Jpn. Sci. Mon.* **2005**, *58*, 48–54.
28. Ehresmann, A.C. MENS, an Info-Computational Model for (Neuro-)cognitive Systems Capable of Creativity. *Entropy* **2012**, *14*, 1703–1716.
29. Freedman, M.H.; Kitaev, A.; Larsen, M.J.; Wang, Z. Topological Quantum Computation. *Bull. Am. Math. Soc.* **2002**, *40*, 31–38.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

*Proceedings*

# Information Processing by Symmetric Inductive Turing Machines †

**Mark Burgin**

Department of Mathematics, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

† Conference Morphological, Natural, Analog and Other Unconventional Forms of Computing for Cognition and Intelligence (MORCOM), Berkeley, CA, USA, 2–6 June 2019.

Published: 13 May 2020

**Abstract:** Traditional models of computations, such as Turing machines or partial recursive functions, perform computations of functions using a definite program controlling these computations. This approach detaches data, which are processed, and the permanent program, which controls this processing. Physical computers often process not only data but also their software (programs). To reflect this peculiarity of physical computers, symmetric models of computations and automata were introduced. In this paper, we study information processing by symmetric models, which are called symmetric inductive Turing machines and reflexive inductive Turing machines.

**Keywords:** information; induction; computation; symmetry; computer; efficiency

## 1. Introduction

Theoretical models of automata and computation have a threefold goal:

- To model physical automata and their functioning by theoretical tools;
- To study physical automata and their functioning by theoretical tools;
- To delineate the further development of physical automata and their functioning by theoretical tools.

To reflect important properties of computers, Marcin Schroeder introduced a new theoretical model of computation—symmetric Turing machines or S-machines [1,2]. In a conventional Turing machine, the head (processor) performs operations with data in the memory (tape) using a fixed system of instructions—its program. In a symmetric Turing machine, information processing goes not only from the head to the memory but also backward. On the one hand, the head (processor) performs operations with data in the memory using a fixed system of instructions—its program. On the other hand, the memory performs operations with instructions from the head (processor).

It is also possible to carry out this approach using two types of memory—data memory and program memory—and having a processor that performs two kinds of operations—operations with data based on information stored in the program memory and operations with the program based on information stored in the data memory.

As we know, physical computers do not only process data but also perform operations with their programs using special tools, such as interpreters, compilers, and translators. There are also program optimizers, which improve characteristics of programs transforming these programs.

Automata that perform transformations with their programs, such as reflexive Turing machines, were explored in [3]. It was proved that these machines have the same computing power as Turing machines. This result disproved Kleene's conjecture [4], which suggested that algorithms that change their programs while computing would be more powerful than Turing machines.

However, it was also proved that reflexive Turing machines could be much more efficient than conventional Turing machines [3].

Using a technique similar to the one employed in [3], it is possible to prove that in a general case, functioning of a symmetric Turing machine working in parallel mode can be simulated by a conventional Turing machine with five tapes and five heads. It means that symmetric Turing machines have the same computing power as Turing machines.

To achieve higher computing power, here we introduce and study symmetric inductive Turing machines, which further develop the structure and possibilities of inductive Turing machines allowing modeling of natural computations in various situations.

## 2. Symmetry in Computations

Physical computers and networks process information using various programs. The system of these programs is called the software of a computer or network, of which the processed information is called infware [5].

There is a definite symmetry between software and infware. To explicate this symmetry, we need a broader understanding of a program. Namely, we define a *program of computation* as a structure that determines (controls) a computational process.

A computational process transforms, transmits, and stores structures, which form infware of the computing system. Usually these structures represent data, although the goal of computing devices is working with knowledge.

Thus, we have two types of structures—controlling and processed structures. However, the roles of these two types of structures can be exchanged—computers can process their programs, e.g., translate them from one programming language to another or optimize them, using data structures for controlling this processing.

This situation displays symmetry between software and infware. Utilization of this symmetry can serve for better organization and efficient optimization of computational processes.

When a computational process involves transformation of not only infware but also software, it becomes symmetric. Consequently, automata and machines that process their infware and software are called *symmetric*.

Note that automata and machines can process their own software or software of other automata and machines. In the first case, corresponding automata and machines are called *reflexive*. In the second case, corresponding automata and machines are called *exterior*. For instance, translators and interpreters are exterior symmetric programs. Automata and machines that can process both their own software and software of other automata and machines are called *coalescent*.

These peculiarities are reflected in some directions in computer programming, such as reflective programming or metaprogramming. They allow computer programs to be observing, reading, generating, analyzing, and modifying other computer programs or themselves at runtime [6–9]. The goal is to allow software developers to minimize the length of code to express a solution, in turn reducing the development time. Reflective programming and metaprogramming also provide means for higher flexibility of programs to efficiently handle new situations without recompilation.

Metaprogramming and reflective programming techniques were popular in the 1970s and 1980s when they were based on list processing languages such as LISP. In particular, LISP hardware machines were employed in the 1980s enabling applications that could process code and were useful for artificial intelligence applications.

There are three modes of temporal organization of symmetric information processing:

- *Program preprocessing,* when at first, the program of the S-machine is processed and then it is used for input data processing;
- *Interchangeable processing,* when the whole computational process is divided into intervals such that an interval of data processing is followed by an interval of program processing, which in turn is followed by an interval of data processing, and so on;
- *Concurrent processing,* when data processing and program processing are performed at the same time.

The latter mode is used in reflexive Turing machines [3]. To organize concurrent processing, a reflexive Turing machine uses two copies of the same program—the execution copy and the processed copy. One is used for data processing, while the second one is transformed, for example, optimized, then at some step of computation, a copy of the processed copy changes the execution copy and is used for data transformation, while the machine continues to transform the processed copy.

## 3. Reflexive Inductive Turing Machines

Thus, we have three classes of symmetric automata and machines:

→ *Reflexive symmetric automata and machines*, which process their own software in addition to their infware;
→ *Exterior symmetric automata and machines*, which process software of other automata and machines in addition to their own infware;
→ *Coalescent symmetric automata and machines*, which process software of other automata and machines in addition to their own software and infware.

Here, we are mostly interested in reflexive symmetric automata and machines, describing how to upgrade inductive Turing machines, enhancing them with symmetric computations.

We remind a definition of an inductive Turing machine [5].

The *hardware* of an inductive Turing machine *M* consists of three abstract devices:

- The *control device* (*controller*) *A* is a finite automaton, which controls functioning of *M;*
- The *operating device* (*processor*) *H* can include any number of processors;
- The memory *E*.

The *software* of an inductive Turing machine *M* consists of rules according to which the operating device *H* functions.

The *infware* of an inductive Turing machine *M* consists of those symbols, words, and languages, which are processed by *M*.

The memory *E* of an inductive Turing machine of the first and higher orders is a network of cells structured by a system of relations that provide connections between cells. This structuring determines input registers, the working memory, and output registers of *M*. Cells can be of different types: *binary cells* store bits of information represented by symbols 1 and 0; *byte cells* store information represented by strings of eight binary digits; *symbol cells* store symbols of the alphabet(s) of the machine *M* and so on.

The *operating device* (*processor*) *H* of an inductive Turing machine consists of one or several heads, similar to those which are used in Turing machines.

An inductive Turing machine gives the result either when it comes to a final state or when the word in the output register stops changing and this word becomes the result of computation. This is the *output-stabilizing mode* of computation. There are also other ways to determine results of inductive computations.

A reflexive inductive Turing machine utilizes two types of memory—data memory and program memory. Namely, to enhance an inductive Turing machine with the possibility to perform symmetric computations, we separate the memory *E* into two parts—$E_d$, which is used for working with the infware (e.g., data) and $E_d$, which is used for working with the software (i.e., instructions) of the machine *M*.

In addition, a reflexive inductive Turing machine has two processors—a data processor and a program (instruction) processor. The data processor performs operations with data based on information in the form of a program stored in the program memory. Similarly, the program (instruction) processor performs operations with the program based on information stored in the data memory. This information is obtained, interpreting definite data as a program.

Both processors function under the control of the same control device *C*, which organizes their interaction, which can be:

- *Separated*, implying program preprocessing when at first, the program of the S-machine is processed and then it is used for input data processing;
- *Interchangeable*, when the whole computational process is divided into intervals such that an interval of data processing is followed by an interval of program processing, which in turn is followed by an interval of data processing, and so on;
- *Parallel*, when processing of programs go parallel with processing data under the regulation of the same control device.

Note that utilization of the common control device for data and program processors does allow the achievement of true concurrency, which demands separate control devices.

Using techniques similar to ones elaborated in [3], it is possible to prove the following results.

**Theorem 1**. *Functioning of a reflexive inductive Turing machine can be simulated by an inductive Turing machine of the same order*.

**Theorem 2**. *For any natural number k, any universal inductive Turing machine U of the first order and any program p for U with time complexity* $T_{U,p}(n) \geq n$, *there is a reflexive inductive Turing machine M of the first order that can perform the same computations with time complexity* $T_{M,p}(n)$, *which is asymptotically less than* $(½)^{-k}\, T_{U,p}(n)$.

These results show that although reflexive inductive Turing machines have the same computing power as inductive Turing machines, they can compute much faster.

## 4. Conclusions

Symmetric inductive Turing machines were defined and their special case—a reflexive inductive Turing machine—was studied. It was demonstrated that introduction of symmetric computations allows a decreasing time of computation although it does not change the computing power of the machines working with one control device.

It was proved that reflexive inductive Turing machines have the same computing power as inductive Turing machines when they process data and programs under the regulation of the same control device. Utilization of separate control devices for data and program processing allows the achievement of true concurrency, which, in a general case, can bring forth computation of recursively incomputable data. Thus, there is an open problem in whether symmetric inductive Turing machines in general, and reflexive inductive Turing machines in particular, can be more computationally powerful than inductive Turing machines.

An interesting problem for future research is elaboration of algorithmic (Kolmogorov) complexity based on symmetric Turing machines and symmetric inductive Turing machines. It has been proved [5] that inductive Turing machines can essentially decrease algorithmic (Kolmogorov) complexity for infinitely many constructive objects in comparison with Turing machines. It would be interesting to find whether symmetric inductive Turing machines can essentially decrease algorithmic (Kolmogorov) complexity for infinitely many constructive objects in comparison with symmetric Turing machines.

**Funding:** This research received no external funding

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Schroeder, M.J. Dualism of Selective and Structural Manifestations of Information in Modelling of Information Dynamics. In *Computing Nature*; SAPERE 7; Springer: Berlin, Germany, 2013; pp. 125–137.
2. Schroeder, M.J. From Proactive to Interactive Theory of Computation. In Proceedings of the 6th AISB Symposium on Computing and Philosophy: The Scandal of Computation—What is Computation? Exeter, UK, 2–5 April 2013; pp. 47–51.

3. Burgin, M. Reflexive Calculi and Logic of Expert Systems. In *Creative Processes Modeling by Means of Knowledge Bases*; Institute of Mathematics: Sofia, Bulgaria, 1992; pp. 139–160. (In Russian)
4. Kleene, S. Constructive and Non-Constructive Operations. In Proceedings of the International Congress of Mathematicians, Edinburg, UK, 14–21 August 1958; Cambridge University Press: Cambridge, UK, 1960.
5. Burgin, M. *Superrecursive Algorithms*; Springer: New York, NY, USA, 2005.
6. Smith, B.C. Procedural Reflection in Programming Languages. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1982.
7. Demers, F.-N.; Malenfant, J. Reflection in logic, functional and object-oriented programming: a Short Comparative Study. In Proceedings of the IJCAI'95 Workshop on Reflection and Metalevel Architectures and their Applications in AI, Montreal, QC, Canada, 20–25 August 1995; pp. 29–38.
8. Chlipala, A. Ur: statically-typed metaprogramming with type-level record computation. In Proceedings of the ACM SIGPLAN 2010 Conference on Programming Language Design and Implementation (PLDI'10), Phoenix, AZ, USA, 22–28 June 2010; Volume 45, pp. 122–133.
9. Ibrahim, M.H. Reflection in Object-Oriented Programming. In Proceedings of the IJCAI'95 Workshop on Reflection and Metalevel Architectures and their Applications in AI, Montreal, QC, Canada, 20–25 August 1995; pp. 29–38.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

information

MDPI

Article

# Triadic Automata and Machines as Information Transformers

**Mark Burgin**

Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

Received: 12 December 2019; Accepted: 31 January 2020; Published: 13 February 2020

**Abstract:** Algorithms and abstract automata (abstract machines) are used to describe, model, explore and improve computers, cell phones, computer networks, such as the Internet, and processes in them. Traditional models of information processing systems—abstract automata—are aimed at performing transformations of data. These transformations are performed by their hardware (abstract devices) and controlled by their software (programs)—both of which stay unchanged during the whole computational process. However, in physical computers, their software is also changing by special tools such as interpreters, compilers, optimizers and translators. In addition, people change the hardware of their computers by extending the external memory. Moreover, the hardware of computer networks is incessantly altering—new computers and other devices are added while other computers and other devices are disconnected. To better represent these peculiarities of computers and computer networks, we introduce and study a more complete model of computations, which is called a triadic automaton or machine. In contrast to traditional models of computations, triadic automata (machine) perform computational processes transforming not only data but also hardware and programs, which control data transformation. In addition, we further develop taxonomy of classes of automata and machines as well as of individual automata and machines according to information they produce.

**Keywords:** information; automaton; machine; hardware; software; modification; process; inductive; recursive; superrecursive; equivalence

## 1. Introduction

It is well known that computers are processing information. At the same time, they are also containers or carriers of information. That is why it is so important to study properties of computers and computer networks from the information perspective. To efficiently do this, researchers need adequate models of computers and computer networks. Many properties of computers, computer networks and computation are well presented in conventional models of automata and algorithms. However, there are still properties that demand new models.

Traditionally, computation is treated as data transformation, which modifies information contained in data and creates new information. At the same time, there were important traits of computers and computer networks, which were missed in traditional models. For instance, the majority of these models do not have input and output systems. The absence of these systems prevented finding and formalizing inductive computations for a long time [1,2].

Other ignored characteristics are related to hardware and software transformations, which take place in computers and computer networks. As a matter of fact, in physical computers, their programs are changing by special software tools such as interpreters, compilers, optimizers and translators. Besides, by using external memory, people change the hardware of their computers. The hardware of computer networks is permanently altering—new computers and other devices are added while other computers and other devices are disconnected.

The goal of this paper is to build models of computation reflecting these characteristics of physical information processing systems and to study their properties.

It is necessary to remark that in our study, we make a distinction between automata, which works autonomously, and machines, which can involve participation of people.

Looking at the history of theoretical computer science, we can see that in the simplest form, hardware modification was present in models of computation from the very beginning. Indeed, to preserve finiteness of its memory, a Turing machine had to modify its hardware or more exactly, its memory in the process of computation because in a general case, the length of the used tape could become larger than any number as the length of the computed words increased. To achieve this, a Turing machine had to add new cells to its tape, i.e., the memory was changing.

This property of Turing machines essentially increases their power because with a finite tape (memory) such a machine would be equivalent to, i.e., not more computationally powerful than, a finite automaton.

In such a natural way, hardware modification emerged at the very beginning of the theory of computation. However, the first model, in which hardware was essentially changing in the process of computation, was *hardware modification machine* [3–5]. Such a machine is a collection of units—each of which is a multiple-input finite-state transducer—and a system of arbitrary connections between units. These connections can be changed during computation, and new units can be activated as the computation proceeds.

Even more essential software modification is performed in inductive Turing machines, where its memory is fundamentally constructed by a special agent in the form of a Turing machine or another inductive Turing machine [6]. Treating this agent as a component of the main machine, we obtain an abstract automaton with hardware self-modification because memory is a part of the machine hardware.

While physical computers also perform operations with their programs using special software tools such as interpreters, compilers, optimizers and translators, the concept of software modification in abstract automata came to computer science much later. The idea of such modification for increasing power of computations belongs to Stephen Kleene, who formulated a conjecture that it might be possible that algorithms that change their programs while computing would be more powerful than Turing machines [7].

To test this conjecture, Mark Burgin constructed reflexive Turing machines [8]. This was the first theoretical model of algorithms that change their programs while computing. Using this model, Burgin proved that the class of reflexive Turing machines is computationally equivalent to the class of Turing machines, i.e., both classes of abstract machines have the same computing power [8]. In such a way, Kleene's conjecture was disproved but, at the same time, it was proved that reflexive Turing machines can be essentially more efficient than Turing machines. Namely, a relevant reflexive Turing machine can effectively outperform any Turing machine that computes the same function [8,9].

A more general model of automata modifying their software—symmetric Turing machines or S-machines—was later suggested by Marcin Schroeder [10,11]. The concept of a symmetric inductive Turing machine was introduced in [12].

There are also directions in computer programming, such as reflective programming or metaprogramming, which allow observing and modifying computer programs at runtime [13–15].

Here, we synthesize hardware and software modification in one theoretical model of a triadic automaton or triadic machine, which processes and changes not only data (infware) but also the software and the hardware of itself. Thus, in general, a triadic automaton (triadic machine) is able to transform all three of its basic components.

Thus, triadic automata (triadic machines) transform information of three types because the hardware of the automaton (machine) is physical information container (carrier), its software is textual information container (carrier) and infware is symbolic information container (carrier).

This paper has the following structure. In the second section, after Introduction, we consider relations between algorithms, automata and machines. In the third section, we discuss the inner

structure of abstract automata, which consists of its hardware, software and infware, which can be treated as data processed by this automaton. In the fourth section, we introduce and formally describe the concept of a triadic automaton. In the fifth section, we study the dynamics of triadic automata and machines. In Conclusion, we summarize obtained results and suggest directions for the future research.

## 2. Algorithms, Automata and Machines

It would be no exaggeration to say that the basic concepts of computer science are algorithms, automata, machines and computation. Nevertheless, there are no unanimously accepted definitions of these important concepts. That is why, in this section, we analyze these concepts, explicate relations between them and suggest informal definitions of these concepts in the context of information processing.

**Definition 1.** *An algorithm is a constructive finite description of a set of processes aimed at solving some problem with the exact definitions of its input and result.*

Here, constructive means that all described operations are comprehensible, executable and finite.

There are different types, categories, kinds, forms and classes of algorithms. Let us consider some of them.

According to the world structuration described in [16], the following types of algorithms are used by people:

1. *Physically represented algorithms*, e.g., the hardware of computers;
2. *Structurally represented algorithms*, e.g., structures of computer programs or of transition rules of finite automata;
3. *Mentally represented algorithms*, e.g., mental schemas of addition or multiplication [17–19].

In turn, physically represented algorithms have the following forms:

1. *Instrumental algorithms*, e.g., algorithms in the form of an automaton;
2. *Textual algorithms*, e.g., a system of instructions or rules;
3. *Numeric algorithms*, e.g., algorithms as weights in neural networks.

Textual and numeric algorithms together form the class of *symbolic algorithms*.

In this context, textual algorithms can be divided into three subclasses:

1. *Operationally expressed* textual algorithms, e.g., systems of instructions;
2. *Functionally expressed* textual algorithms, e.g., partially recursive functions;
3. *Intentionally expressed* textual algorithms giving only descriptions of what is necessary to do, e.g., programs in functional or relational programming languages.

This classification reflects levels of explicitness of algorithm representations.

According to their forms, the following categories of algorithms are used by people:

1. *Parametric algorithms*, e.g., weights in neural networks;
2. *Instruction algorithms*, e.g., rules in Turing machines;
3. *Description algorithms*, e.g., programs in functional or relational programming languages.

This classification reflects levels of symbolism in algorithm representations.

There are also levels of algorithms, which reflect grades of explicit descriptions of computational processes by algorithms [20,21]. For simplicity, we describe these levels for algorithms that have the form of a system of instructions or rules.

*Algorithms of the first level* contain only instructions (rules) for data transformation.

*Algorithms of the second level* contain instructions (rules) for data transformation and execution instructions (or metarules), which describe how to apply data transformation instructions (rules).

For instance, in finite automata, instructions (rules) that define how to select an appropriate transition are execution instructions (or metarules).

Note that metarules of instruction selection can essentially influence the functioning of systems where they are used.

*Algorithms of the third level* contain instructions (rules) for data transformation, execution instructions (metarules) of the first level, which describe how to apply data transformation instructions (rules) and execution instructions (metarules) of the second level, which describe how to apply execution instructions (metarules) of the first level.

It is possible to continue this construction considering algorithms of any level $n$.

Algorithms are utilized to control and direct the functioning of automata and machines, which are systemic devices. That is why, at first, we consider the general concept of a device.

**Definition 2.** *A device is a structure that performs actions and generates processes.*

There are different types, categories, kinds and classes of devices. Let us consider some of them.

According to the world structuration described in [16], the following types of devices are used by people:

1. *Physical devices*, e.g., computers or calculators;
2. *Abstract devices*, e.g., abstract automata such as finite automata or Turing machines;
3. *Mental devices*, e.g., mental schemas of neural systems, which perform addition or multiplication [22,23].

It is important to understand that only instrumental algorithms are devices. Two other classes—textual and numeric algorithms—are not devices because they do not perform actions themselves. For instance, a Turing machine is a device while a partial recursive function is not a device. Algorithms that are not devices need devices or people to be performed.

According to their origin, the following classes of devices are created and used by people:

1. *Artificial devices* are created by people, e.g., computers;
2. *Natural devices* exist in nature, e.g., an organism of a living being or the Earth, considered as a device;
3. *Combined devices* are combinations of artificial and natural devices, e.g., a car with a driver or a plane with a pilot.

People construct and use a diversity of artificial devices—computers, cars, planes, cell phones, ships and so on. At the same time, people use also different natural devices. The system of a sundial is an example of such a device. It consists of three subsystems: the Sun, the Earth and the sundial itself.

According to the classification of natural systems, the following kinds of devices exist:

1. *Animate devices* are or include living individuals,
2. *Inanimate devices* do not include living individuals, and
3. *Hybrid natural devices* are combinations of animate and inanimate devices.

In some generalized sense, it is possible to consider all living beings as animate devices. The Solar System is an example of inanimate device.

Automata and machines are important special cases of devices.

**Definition 3.** *An automaton is a device that autonomously performs actions prescribed by an algorithm when a relevant input is given.*

The functioning of automata is controlled by algorithms.

Similar to devices in general, there are three basic classes of automata:

1. *Artificial automata* are created by people, e.g., electronic clocks;
2. *Natural automata* exist in nature, e.g., the Solar system;
3. *Combined automata* are combinations of artificial and natural devices, e.g. a sundial, which consists of two natural systems - the Sun and the Earth – combined with one artificial system - the sundial itself.

In essence, all classifications of devices are applicable to automata.

Very often people treat theoretical automata and machines as the same class of objects. For instance, Turing machines are considered the most popular class of abstract automata. However, in our study, we make a distinction between automata and machines.

**Definition 4.** *A machine is a device such that when a relevant input is given, performs actions prescribed by an algorithm which can involve participation of people.*

For instance, cell phones are machines while clocks are automata.

Note that according to these definitions, any automaton is a machine, but it is not true that any machine is an automaton.

In essence, all classifications of devices induce corresponding classes of automata and machines because automata and machines are devices. To avoid repetition, we do not consider those types, categories, kinds and classes of automata and machine classification that are induced by classifications of devices. At the same time, there are classifications of automata and machines brought about by classification of algorithms. It is possible to take the following classification as an example.

According to the classification of algorithms that control automata, the following categories of automata are used in computer science:

1. *Parametric automata*, e.g., neural networks, are controlled by parametric algorithms;
2. *Instruction automata*, e.g., Turing machines, are controlled by instruction algorithms;
3. *Description automata*, e.g., symmetric Turing machines in the sense of [10,11,24], are controlled by description algorithms.

Devices in general and automata and machines, in particular, can perform various functions and generate diverse processes. Here we are interested in information processing in the form of computation.

Although computation pervades the contemporary society, there is no unanimously accepted definition of computation [25]. At the same time, there is a variety of different definitions and description—some of which we consider here.

After Turing machine was accepted as the uppermost model of algorithms, computation has been interpreted as what a Turing machine is doing. When it became clear that computation can go beyond Turing machines, the opposite trend appeared, in which it was supposed that any system in nature or society is computing. This approach is called pancomputationalism (cf., for example, [26–28]).

As it is explained in [29], there are three basic levels of generality in understanding the phenomenon of computation:

On the top level, *computation* is perceived as any transformation of information and/or information representation.

On the middle level, *computation* is distinguished as a discrete process of transformation of information and/or information representation.

On the bottom level, *computation* is recognized as a discrete process of symbolic transformation of information and/or symbolic information representation.

Here, we take on the engineering approach to computation according to which computation is a process performed by information processing devices. Besides, in what follows, we consider only computing automata and machines.

## 3. Inner Structure of Abstract Automata

On analyzing a computer, we see that it has various devices, e.g., one or several processors, memory of different types and input/output devices [30]. All these devices and connections between them constitute the *hardware* of this computer.

In addition, a computer has various programs, which direct its functioning. All these programs cast the *software* of this computer.

Besides, a computer works with diverse data. These data are unified under the name the *infware* of this computer.

In a similar way, the inner structure of an abstract automaton consists of three components—hardware, software and infware—and relations between them [1].

**Definition 5.** *The hardware of an abstract automaton comprises theoretical (abstract) devices, such as a control device, processor or memory, which play a part in computations performed by this automaton, and connections between these devices.*

In a general situation, the hardware of a real information processing system has three key components: the input subsystem, output subsystem and processing subsystem. Thus, to properly represent information processing systems, which are actually utilized by people, the hardware of an abstract automaton must have three basic components: (abstract) *input device*, (abstract) *information processor*, and (abstract) *output device* [1]. In many theoretical models of computation, input and output devices are either not specified or represented by components of the common memory and/or of the processor. For example, in a typical Turing machine, operations of input and output utilize the working memory—one or several tapes. In contrast to this, inductive Turing machines contain special input and output registers, e.g., tapes [6]. The same is true for pointer machines. Indeed, a pointer machine receives input—finite sequences of symbols (words) from its "read-only tape" (or an equivalent storage device) – and it writes output sequences of symbols on an output "write-only" tape (or an equivalent storage device).

Neural networks also have these core components: the input subsystem that comprises all input neurons, output subsystem that consists of all output neurons, and it is possible to treat all neurons of the network or only all hidden neurons as its information processing subsystem [31]. In some cases, input and output neurons are regarded as one group of visible neurons.

**Definition 6.** *Infware of an abstract automaton consists of objects processed by this automaton including input and output data.*

Here are some examples.

The majority of abstract automata (computing devices) work with linear (i.e., one-dimensional) languages. Consequently, their infware consists of words in some alphabet.

Kolmogorov algorithms and storage modification machines work with arbitrary graphs [32,33]. Consequently, their infware consists of graphs.

There are also many practical algorithms that work with graphs (cf., for example, [34]). Consequently, their infware also consists of graphs.

Turing machines with two-dimensional tapes and two-dimensional cellular automata work with two-dimensional arrays of words. This means that their infware consists of two-dimensional arrays of words.

Turing machines with $n$-dimensional tapes and $n$-dimensional cellular automata work with $n$-dimensional arrays of words. This means that their infware consists of $n$-dimensional arrays of words.

Structural machines work with arbitrary structures [35,36]. Consequently, their infware consists of arbitrary structures.

Note that such advanced abstract automata as structural machines can process not only data but also knowledge [37]. In this case, their infware consists of knowledge.

It is necessary to remark that the word *infoware*, which is used in the field of networks and computers, has a very different meaning in comparison with the term *infware*.

**Definition 7.** *Software of an abstract automaton consists of texts, which control the functioning of this automaton.*

Here are some examples.

Many kinds of algorithms and abstract automata, such as finite automata, pushdown automata, register machines, Kolmogorov algorithms, random access machines (RAM), and Turing machines, use systems of instructions, for example, in the form of transition rules, to control computational processes. Such systems of instructions constitute software of these automata and machines.

The system of weights, activation functions, threshold functions and output functions form software of neural networks. It is possible to treat these systems as algorithms although their form is different from traditional algorithms, which are described as sets of instructions.

In contrast to neural networks, software of the majority of algorithms and abstract automata consists of systems of instructions. These instructions or rules determine computational processes, which are controlled by algorithms and are going in these automata. All these classes of algorithms and abstract automata are unified by the comprehensive concept of an instruction machine.

**Definition 8.** *(a) An instruction machine or instruction automaton M is an automaton functioning of which is determined by a system of instructions (rules).*
*(b) A pure instruction machine or pure instruction automaton M is an automaton functioning of which is determined only by a system of instructions (rules) and its input.*

Note that the functioning of an instruction automaton is not necessarily uniquely determined by its system of instructions. For instance, its functioning can also depend on the states of its control device as in Turing machines. Besides, in nondeterministic instruction machines, e.g., in nondeterministic Turing machines, there are metarules of instruction selection, which can essentially influence its functioning [20,21].

At the same time, an important dynamic characteristic of the majority of abstract automata is their state. This brings us to another important class of automata.

**Definition 9.** *(a) A state machine or state automaton M has a control device and is an automaton functioning of which is determined by the states of its control device.*
*(b) A pure state machine or pure state automaton M has a control device and is an automaton functioning of which is determined only by the states of its control device and its input.*

Note that the control device of an automaton can coincide with the whole automaton. In this case, the functioning of the automaton is determined by its states. However, when the machine has a control device makes this automaton more flexible.

Often state machines (state automata) are also instruction machines (automata) with systems of instructions. However, implicitly any state machine (automaton) $M$ is an instruction machine (automaton). Indeed, if we take descriptions of how the functioning of the machine (automaton) $M$ depends on the state, we obtain instructions (rules) of its functioning.

We observe this situation in the case of finite automata. A finite automaton is a pure state machine (automaton) but its transition function (relation) makes it also an instruction machine (automaton).

Let us consider the structure of a state instruction machine (state instruction automaton) in a general case.

In a general case, a state instruction machine (state instruction automaton) $M$ has three components:

- The *control device* $C_M$, which is a finite automaton and represents states of the machine (automaton) $M$;

- The *memory* $W_M$, which stores data;
- The *processor* $P_M$, which transforms (processes) information (data) from the input and the memory $W_M$.

The memory $W_M$ consists of cells and connections between them. Each cell can be empty or contain a symbol from the alphabet $A_M$ of the machine (automaton) $M$.

On each step of computation, the processor $P_M$ observes one cell from the memory $W_M$ at a time, and can change the symbol in this cell and go to another cell using connections in the memory $W_M$. These operations are performed according to the instructions $R_M$ for the processor $P_M$. These instructions $R_M$ can be stored in the processor $P_M$ or in the memory $W_M$.

However, it is possible that an instruction machine consists of a single processor as its particular case—a finite automaton.

**Example 1.** *A finite automaton G is an instruction machine, which has the following representation. Namely, a finite automaton (FA) G consists of three structures:*

- *The linguistic structure* $L = (\sum, Q, \Omega)$ *where* $\sum$ *is a finite set of input symbols, Q is a finite set of states, and* $\Omega$ *is a finite set of output symbols of the automaton G;*
- *The state structure* $S = (Q, q_0, F)$ *where* $q_0$ *is an element from Q that is called the start state and F is a subset of Q that is called the set of final (in some cases, accepting) states of the automaton G;*
- *The action structure, which is traditionally called the transition function of G and has the following form*

$$\delta\colon \sum \times Q \rightarrow Q \times \Omega$$

*It can also be represented as two relation/functions—the state transition relation/function*

$$\delta_1\colon \sum \times Q \rightarrow Q$$

*and the output relation/function*

$$\delta_2\colon \sum \times Q \rightarrow \Omega$$

*Thus, a FA is a triad* $G = (L, S, \delta)$.

*The transition relation/function* $\delta$ *is portrayed by descriptions of separate transitions and each of these descriptions is an instruction (rule) for the automaton functioning.*

*Note that a finite automaton does not have a memory.*

**Example 2.** *A Turing machine T is also an instruction machine because its functioning is defined by a system of rules (instructions), which have the following form for a Turing machine with one head*

$$qa \rightarrow pbQ$$

*Here, a is the symbol, which is observed by the head of T and changed to the symbol b, and q is a state of the Turing machine, or more exactly, of its control device, which is changed in this operation to the state p, while Q is direction of the move of the head after performing the writing operation.*

**Example 3.** *An inductive Turing machine K is also an instruction machine because its functioning is defined by a system of rules (instructions), which are similar to rules of Turing machines [1,6].*

There are numerous kinds of instruction machines with various types of instructions. However, it is possible to distinguish three classes of instructions:

- *Straightforward* or *prescriptive instructions* directly tell what is necessary to do.
- *Descriptive instructions* describe what result it is necessary to obtain.
- *Implicit instructions* have a form of data that can be interpreted as instructions.

Let us consider some examples.

**Example 4.** *The descriptions of transitions of a finite automaton G are straightforward instructions.*

**Example 5.** *A function in functional programming is a descriptive instruction.*

**Example 6.** *Weights of artificial neurons in artificial neural networks are implicit instructions.*

While all these examples are conventional models of computation, in the next section, we introduce and study more advanced models.

## 4. Structure of Triadic Automata and Machines

Triadic automata (machines) transform data (infware), instructions (software) and memory (hardware). Before describing their structure, we make an inventory of the types of triadic machines (triadic automata).

**Definition 10.** *A triadic machine (automaton) is called:*

- *a hardware modification machine (automaton) if it transforms only infware and hardware,*
- *a software modification or symmetric machine (automaton) if it transforms only infware and software,*
- *a transducer if it transforms only infware and has input and output,*
- *a generator if it transforms only infware and has only output,*
- *an acceptor if it transforms only infware and only input,*
- *a hardware expansion machine (automaton) if it only expands its hardware,*
- *a software expansion machine (automaton) if it only expands its software, and*
- *a symmetric expansion machine (automaton) if it only expands its hardware and software.*

Besides, there are different ways to perform hardware/software modifications. With respect to the source of modification, it is possible to consider three types of hardware/software modifications in an automaton (machine) $M$:

*External modification* is performed by another system.

*Internal modification* is performed by the automaton (machine) $M$.

*Combined modification* is performed by both the automaton (machine) $M$ and another system.

What modifications are possible and permissible depends on the structure of a triadic machine or triadic automaton. Its mandatory components are input and output systems working together with one or more processors. Usually, input and output components are specific registers in the memory of the machine (automaton) [1]. At the same time, in neural networks, input and output are organized using specified neurons [31].

However, adding memory and other components to automata allows increasing their flexibility, interoperability and efficiency. These changes are reflected in the structure of triadic machines (automata), which have different types. Here we consider two types state and instruction triadic automata (machines).

**Definition 11.** *A triadic state machine or triadic state automaton A with memory has seven core hardware components:*

- *The control device $C_A$, which is a finite automaton and represents states of the machine (automaton) A;*
- *The data memory $W_A$, which stores data and includes input and output registers;*
- *The software memory $V_A$, which stores software of the machine (automaton) A;*
- *The data processor $P_M$, which transforms (processes) information (data) from the memory $W_M$;*
- *The software processor $D_M$, which transforms (processes) software of A stored in the memory $V_M$;*

- *The metaprocessor $P_A$, which transforms (e.g., builds or deletes connections in) the hardware $H_A$ and/or changes the control device $C_A$.*

In the standard form, both memories consist of cells, which are connected by transition links. Processors have their programs of functioning, which constitute the software of the automaton.

In the same way as triadic state machines, triadic instruction machines constitute a special class of triadic machines. In a general case, it is possible that the functioning of a triadic instruction machine does depend on its state. However, we include the state system in the general description of triadic instruction machines because when the functioning of a triadic instruction machine does depend on its state, it is possible to treat this as a machine with only one state.

**Definition 12.** *A triadic instruction machine or triadic instruction automaton H with memory has seven core hardware components:*

- *The control device $C_H$, which is a finite automaton and represents states of the machine (automaton) H;*
- *The data memory $W_H$, which stores data;*
- *The instruction memory $V_H$, which stores instructions;*
- *The data processor $P_M$, which transforms (processes) information (data) from the memory $W_M$;*
- *The instruction processor $D_M$, which transforms (processes) information (instructions) from the memory $V_M$;*
- *The memory processor $P_W$, which transforms (builds or deletes connections and/or cells in) the memory $W_M$;*
- *The memory processor $P_V$, which transforms (e.g., builds or deletes connections and/or cells in) the memory $V_M$.*

Memory processors are hardware transformers and it is also possible to include a control device processor in the structure of a triadic instruction machine. This additional processor changes the control device $C_A$.

There are different classes of triadic instruction machines (automata).

**Definition 13.** *Triadic instruction machines (automata) that:*

- *do not have processors that transform memory are called symmetric instruction machines,*
- *have only processor(s) that transform data are called instruction machines,*
- *have only processor(s) that transform instructions are called translation machines or translators,*
- *have only processor(s) that transform memory are called construction machines or constructors,*
- *do not have processors that transform data are called constructors (construction machines) with translators, and*
- *do not have processors that transform instructions are called generative instruction machines.*

Machines from each class have their specific functions. For instance, construction machines (constructors) can be used to construct memory for other machines. This technique is employed in inductive Turing machines of the second and higher orders use inductive Turing machines of lower orders as their constructors [1,6].

Besides, there are different methods to organize program formation with the help of computing/constructing agents. If the memory of the automaton has connections between any pair of cells, then the program can use these connections. Thus, it is possible to organize the inductive mode of computing by inductive computation (compilation) of the program for the main computation.

In the simplest approach called the *sequential strategy*, it is assumed that given some schema, for example, a description of the structure of the memory $E$ of an inductive Turing machine $M$, an automaton $A$ builds the program and places it in the memory $E$ before the machine $M$ starts its computation. When $M$ is an inductive Turing machine of the first order, its constructor $A$ is a Turing machine, which, for example, puts the names of the connections of the memory of $M$ into instructions (rules) of $M$. When $M$ is

an inductive Turing machine of the second or higher order, its constructor $A$ is also an inductive Turing machine—the order of which is less than the order of $M$ and which modifies instructions (rules) of $M$. For instance, the program of inductive Turing machines of the second order is constructed by Turing machines of the first order.

According to another methodology, which is called the *concurrent strategy*, program formation by the automaton $A$ and computations of the machine $M$ go concurrently, that is, while the machine $M$ computes, the automaton $A$ constructs the program in the memory $E$.

It is also possible to use the *mixed strategy* when some parts of the program $E$ are assembled before the machine $M$ starts its computation, while other parts are formed parallel to the computing process of the machine $M$.

These three strategies determine three kinds of the constructed program (software):

- In the *static program* (*static software*) of the machine $M$, everything is constructed before $M$ starts working.
- In the *growing program* (*growing software*) of the machine $M$, parts are constructed while $M$ is working but no parts are deleted.
- In the *dynamic program* (*growing software*) of the machine $M$, when it necessary, some parts are constructed and when it necessary, some parts are deleted while $M$ is working.

It is possible to use similar strategies for hardware modification. This approach determines three types of the constructed hardware of a triadic automaton/machine:

- In the *static hardware* of the machine $M$, everything is constructed before $M$ starts working.
- In the *growing hardware* of the machine $M$, parts are constructed while $M$ is working but no parts are deleted.
- In the *dynamic hardware* of the machine $M$, when it necessary, some parts are constructed and some parts are deleted while $M$ is working.

Now, we can analyze the functioning of triadic automata and machines in more detail.

## 5. The Dynamics of Triadic Automata and Machines

To describe the dynamics of triadic automata and machines, we need some concepts from the theory of computation and automata.

There are different equivalence relations between automata, machines and algorithms. Three basic ones are determined by properties of computations [1,38].

**Definition 14.** *(a) Two automata, machines or algorithms A and B are operationally equivalent if given the same input, they perform the same operations.*
*(b) Two classes of automata, machines or algorithms* ***H*** *and* ***K*** *are operationally equivalent if each automaton in* ***H*** *is operationally equivalent to an automaton in* ***K*** *and vice versa.*

For instance, a pushdown automaton, which does not use its stack, is operationally equivalent to a nondeterministic finite automaton. Consequently, the class of all pushdown automata, which do not use their stack, is operationally equivalent to the class of all nondeterministic finite automata.

However, operational equivalence does not completely characterize automata, machines and algorithms with respect to their functioning. To achieve this goal, we need a stronger equivalence.

**Definition 15.** *(a) Two operationally equivalent automata, machines or algorithms A and B are strictly operationally equivalent if for both of them the result is determined by the same rules.*
*(b) Two classes of automata, machines or algorithms* ***H*** *and* ***K*** *are strictly operationally equivalent if each automaton in* ***H*** *is strictly operationally equivalent to an automaton in* ***K*** *and vice versa.*

By definition, strictly operationally equivalent automata, machines or algorithms are operationally equivalent. However, properties of computation show that two automata (machines or algorithms) can be operationally equivalent but not strictly operationally equivalent. For instance, a Turing machine with one working tape, one input read-only tape, one output write-only tape and with three corresponding heads is operationally equivalent to a simple inductive Turing machine [1]. However, these machines are not strictly operationally equivalent because their results are defined by different rules.

In a similar way, strictly operationally equivalent classes of automata, machines or algorithms are operationally equivalent. However, the previous example shows that two classes of automata (machines or algorithms) can be operationally equivalent but not strictly operationally equivalent.

In addition to two kinds of operational equivalence, there are other forms of equivalence of automata, machines and algorithms. In particular, it is known that automata can perform different operations but give the same result. This observation brings us to two more types of automata/machines equivalence.

**Definition 16.** *(a) Two classes of automata, machines or algorithms* ***H*** *and* ***K*** *are functionally equivalent if the automata, machines or algorithms from* ***H*** *and* ***K*** *compute the same class of functions.*
*(b) Two automata, machines or algorithms A and B are functionally equivalent if the classes {A} and {B} are functionally equivalent.*

For instance, the class of all Turing machines with one tape and the class of all one-dimensional cellular automata are functionally equivalent because they compute the same class of partially recursive functions [1].

Properties of automata machines and algorithms imply the following result.

**Lemma 1.** *If the results of computations are defined in the same way for two (classes of) automata machines or algorithms, then their operational equivalence implies their functional equivalence.*

In other words, strict operational equivalence implies functional equivalence.

However, when the results of computations are defined in a different way, two operationally equivalent (classes of) automata machines or algorithms can be not functionally equivalent. For instance, the class of all Turing machines with three tapes and heads and the class of all simple inductive Turing machines are operationally equivalent but they are not functionally equivalent because inductive Turing machines can compute much more functions than Turing machines [6].

Let us assume that all considered below automata (machines) work with words in some alphabet. Naturally, these automata (machines) compute some languages.

**Definition 17.** *(a) Two classes of automata, machines or algorithms* ***H*** *and* ***K*** *are linguistically equivalent if they compute (or accept) the same class of languages.*
*(b) Two automata, machines or algorithms A and B are linguistically equivalent if the classes {A} and {B} are linguistically equivalent.*

For instance, the class of all deterministic finite automata and the class of all nondeterministic finite automata are linguistically equivalent [1].

Properties of functions imply the following result.

**Lemma 2.** *[38]. Functional equivalence of two (classes of) automata machines or algorithms implies their linguistic equivalence.*

Lemmas 1 and 2 imply the following result.

**Corollary 1.** *Strict operational equivalence implies linguistic equivalence.*

However, operational equivalence implies neither functional nor linguistic equivalence. Indeed, when the results of computations are defined in a different way, two operationally equivalent (classes of) automata machines or algorithms can be not linguistically equivalent. For instance, the class of all Turing machines with three tapes and heads and the class of all simple inductive Turing machines are operationally equivalent but they are not linguistically equivalent because inductive Turing machines can compute much more languages than Turing machines [6].

All classes of automata, machines or algorithms studied and utilized in computer science and mathematics are usually divided into three substantial categories: subrecursive, subrecursive and superrecursive classes of automata, machines or algorithms [1]. Here we introduce a more detailed classification.

**Definition 18.** *A class of automata, machines or algorithms* ***R*** *is called linguistically (functionally) recursive if it is linguistically (functionally) equivalent to the class of all Turing machines.*

**Example 7.** *The class of all Random Access Machines (RAM) is linguistically and functionally recursive [39].*

**Example 8.** *The class of all cellular automata is linguistically recursive [40].*

**Example 9.** *The class of all storage modification machines is linguistically and functionally recursive [33].*

**Example 10.** The class of all Minsky machines is linguistically and functionally recursive [41].

Any language computed by some Turing machine is also computed by a universal Turing machine [42]. This gives us the following result.

**Theorem 1.** *A class of automata or machines* ***H*** *is linguistically recursive if and only if it is linguistically equivalent to the class {U} where U is a universal Turing machine.*

Recursive classes of automata, machines or algorithms form a theoretical threshold for separating two basic classes—subrecursive and superrecursive automata, machines or algorithms. Namely, Turing machines are also used to define two more classes of automata, machines or algorithms.

**Definition 19.** *A class of automata, machines or algorithms* ***W*** *is called linguistically (functionally) subrecursive if not all languages (functions) computable/acceptable in the class of all Turing machines are computable/acceptable in* ***W****.*

**Example 11.** *The class of all deterministic finite automata is linguistically and functionally subrecursive.*

**Example 12.** *The class of all pushdown automata is linguistically and functionally subrecursive.*

**Example 13.** *The class of all resource restricted Turing machines is linguistically and functionally subrecursive.*

For a long time, it was believed that functionally recursive algorithms/automata in general and Turing machines in particular form the most computationally powerful class of algorithms/automata. In spite of this different classes of more powerful algorithms/automata have been constructed. They form one more basic class of algorithms/automata.

**Definition 20.** *A class of automata, machines or algorithms* ***U*** *is called linguistically (functionally) superrecursive if not all languages (functions) computable/acceptable in* ***U*** *are computable/acceptable in the class of all Turing machines.*

**Example 14.** *The class of all inductive Turing machines is linguistically and functionally superrecursive [1,6].*

**Example 15.** *The class of all periodic Turing machines is linguistically and functionally superrecursive [43].*

**Example 16.** *The class of all inductive cellular automata is linguistically and functionally superrecursive [44].*

However, the possibility to compute a function (language) noncomputable by Turing machines does not guaranty that that in this class of algorithms/automata, all functions (languages) computable by Turing machines will be computable. In other words, it is possible that a linguistically and functionally superrecursive class of automata, machines or algorithms does not contain a linguistically and functionally recursive subclass of automata, machines or algorithms. That is why it is reasonable to consider one more important category of automata (machines) related to their functioning.

**Definition 21.** *A superrecursive class of automata, machines or algorithms* $\boldsymbol{U}$ *is called strictly linguistically (functionally) superrecursive if all languages (functions) computable/acceptable in the class of all Turing machines are also computable/acceptable in* $\boldsymbol{U}$*.*

**Example 17.** *The class of all simple inductive Turing machines is strictly linguistically and functionally superrecursive [1,6].*

**Example 18.** *The class of all inductive cellular automata is strictly linguistically and functionally superrecursive [44].*

**Example 19.** *The class of all neural networks with real number weights is strictly linguistically and functionally superrecursive [45].*

**Example 20.** *The class of all general machines working with real numbers is strictly linguistically and functionally superrecursive [46,47].*

**Lemma 3.** *Any strictly linguistically (functionally) superrecursive class of automata, machines or algorithms is linguistically (functionally) superrecursive.*

The introduced stratification of classes of automata or machines into three groups allows obtaining similar classification of individual automata or machines.

**Definition 22.** *(a) An automaton (machine) A is linguistically (functionally) recursive if the class {A} is linguistically (functionally) recursive.*
*(b) An automaton (machine) A is nominally linguistically (functionally) recursive if it belongs to a class of linguistically (functionally) recursive automata (machines).*

In other words, automaton (machine) $A$ is linguistically (functionally) recursive if the class $\{A\}$ is linguistically (functionally) equivalent to the class of all Turing machines.

**Example 21.** *A universal Turing machine is linguistically and functionally recursive.*

**Example 22.** *Any Turing machine is nominally linguistically and functionally recursive.*

Note that any Turing machine is nominally linguistically and functionally recursive, but it is not always linguistically and functionally recursive.

Properties of universal Turing machines imply the following result.

**Theorem 2.** *An automaton (machine) A is linguistically recursive if and only if it is linguistically equivalent to a universal Turing machine.*

**Definition 23.** *An automaton (machine) A is linguistically (functionally) subrecursive if the class {A} is linguistically (functionally) subrecursive.*

**Example 23.** *Any finite automaton is linguistically and functionally subrecursive.*

Note that any Turing machine is nominally linguistically and functionally recursive but in the majority of cases, it is linguistically and functionally subrecursive.

**Proposition 1.** *A nominally linguistically (functionally) recursive automaton (machine) A is also nominally linguistically subrecursive if it is not linguistically (functionally) recursive and superrecursive.*

**Definition 24.** *(a) An automaton (machine) A is linguistically (functionally) superrecursive if the class {A} is linguistically (functionally) superrecursive.*
*(b) An automaton (machine) A is nominally linguistically (functionally) superrecursive if it belongs to a class of linguistically (functionally) superrecursive automata (machines).*

**Example 24.** *A universal inductive Turing machine is linguistically and functionally superrecursive.*

Note that any inductive Turing machine is nominally linguistically and functionally superrecursive but in the majority of cases, it is linguistically and functionally recursive or subrecursive. For instance, inductive Turing machines that do not use memory are linguistically and functionally subrecursive.

This clearly shows the importance of the memory for the computing power of automata and machines. Structured memory is an important concept introduced in the theory of inductive Turing machines [48]. We remind that a structured memory *E* consists of cells and is structured by a system of relations and connections (ties) that organize memory functioning and provide connections between cells. This structuring delineates three components of the memory E: input registers, the working memory, and output registers. In a general case, cells may be of different types: binary cells store bits of information represented by symbols 1 and 0; byte cells store information represented by strings of eight binary digits; symbol cells store symbols of the alphabet(s) of the machine M and so on. Thus, the memory is a network of cells and it is possible to interpret these cells not only as containers of symbols but also as information processing systems, such as neurons in the brain of a human being or computers in the World Wide Web or abstract computing devices in a grid automaton [49].

**Definition 25.** *The software (hardware) of a triadic automaton is recursive if it and its transformation rules determine recursive algorithms.*

**Example 25.** *The software (hardware) of a Turing machine is recursive.*

Let *A* be an instruction automaton (machine) with a structured memory.

**Theorem 3.** *If the instructions (rules) of the automaton (machine) A are recursive, i.e., they define a recursive or subrecursive algorithm, but the structuring of its memory is superrecursive, then A can also be superrecursive.*

The structuring of the automaton (machine) memory can be performed as hardware modification similar to the construction of inductive Turing machines of the second and higher orders. As inductive Turing machines of any order form a superrecursive class of automata, Theorem 3 give us the following result.

**Corollary 2.** *It is possible to achieve superrecursivity by hardware modification.*

**Corollary 3.** *A triadic automaton with recursive software can be superrecursive.*

An example of such an automaton is given in the studies of interactive hypercomputation [50].

**Definition 26.** *Information processing, e.g., computation, is recursive if it can be realized by a recursive algorithm.*

Properties of recursive computations imply the following result [1].

**Theorem 4.** *A triadic automaton with recursive software and hardware and their parallel recursive data processing and software/hardware modification is recursive.*

This result shows that parallel to data processing recursive software/hardware modification cannot increase computing power. However, even recursive rules of hardware modification can essentially improve efficiency of automata and machines decreasing their time complexity. Namely, results from [51] allow proving the following result.

**Theorem 5.** *For any recursively computable language L (function f), there is a recursive hardware modification automaton A with a priory structured memory, which computes the language L (the function f), with linear time complexity.*

As recursive hardware modification automata are also triadic automata, we have the following corollary.

**Corollary 4.** *For any recursively computable language L (function f), there is a recursive triadic automaton A, which computes L (f) with linear time complexity.*

Inductive Turing machines form an important class of automata formalizing scientific induction and inductive reasoning in general. That is why we can use inductive Turing machines for developing a new classification of automata, machines or algorithms. Here we do this for inductive Turing machines of the first order.

**Definition 27.** *A class of automata, machines or algorithms* ***D*** *is called linguistically (functionally) plainly inductive if it is linguistically (functionally) equivalent to the class* $\boldsymbol{IM}_1$ *of all inductive Turing machines of the first order.*

**Example 26.** *The class of all simple inductive Turing machines is linguistically and functionally inductive [1,6].*

**Example 27.** *The class of all periodic Turing machines is linguistically and functionally inductive [43].*

**Example 28.** *The class of all inductive cellular automata is linguistically and functionally inductive [44].*

Properties of universal inductive Turing machines of the first order (cf. [1,6]) give us the following result.

**Theorem 6.** *A class of automata or machines* ***H*** *is linguistically plainly inductive if and only if it is linguistically equivalent to the class {W} where W is a universal inductive Turing machine of the first order.*

**Definition 28.** *A class of automata, machines or algorithms* ***W*** *is called linguistically (functionally) plainly superinductive if not all languages (functions) computable/acceptable in* ***W*** *are computable/acceptable in the class of all inductive Turing machines of the first order.*

Properties of universal inductive Turing machines of the second or higher orders (cf. [1,6]) give us the following result.

**Theorem 7.** *A class of automata or machines* $\boldsymbol{H}$ *is linguistically (functionally) plainly superinductive if it is linguistically (functionally) equivalent to the class* $\boldsymbol{IM}_2$ *of all inductive Turing machines of the second or higher order.*

This brings us to the following problem.

**Problem 1.** *Are there proper subclasses of the class* $\boldsymbol{IM}_2$ *of all inductive Turing machines of the second order, which are computationally weaker than* $\boldsymbol{IM}_2$ *but are still linguistically or functionally plainly superinductive?*

## 6. Conclusions

We introduced concepts of triadic automata and machines. Relations between different types of triadic automata and machines and their properties were obtained. In addition, we further developed taxonomy of classes of automata and machines as well as of individual automata and machines. All this opens new directions for further research.

Inductive Turing machines form a powerful class of algorithms [6]. Thus, formalization and exploration of triadic inductive Turing machines is a motivating problem for future research. It is possible to study the same problem for periodic Turing machines [43], which are intrinsically related to inductive Turing machines.

Structural machines provide an extremely flexible model of computation [35,36]. It would be interesting to introduce and study triadic structural machines.

It would also be interesting to formalize triadic cellular automata, triadic inductive cellular automata [44] and triadic neural networks and study their properties.

Traditional recursive and superrecursive models of automata and algorithms are used for the development of algorithmic information theory. Thus, an important direction for the future research is the development of triadic information theory based on triadic automata.

**Funding:** This research received no external funding.

**Acknowledgments:** The author would like to express gratitude to Robert Prentner and another reviewer unknown to the author for their useful remarks.

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Burgin, M. *Superrecursive Algorithm*; Springer: New York, NY, USA, 2005.
2. Burgin, M. Inductive Turing Machines. In *Unconventional Computing—A volume in the Encyclopedia of Complexity and Systems Science*; Adamatzky, A., Ed.; Springer: Berlin/Heidelberg, Germany, 2018; pp. 675–688.
3. Dymond, P.W.; Cook, S.A. Hardware complexity and parallel computation. In Proceedings of the 21st Annual Symposium on Foundations of Computer Science, Syracuse, NY, USA, 13–15 October 1980; pp. 360–372.
4. Cook, S.A. Towards a complexity theory of parallel computation. *L'Enseignement Math.* **1981**, *XXVII*, 99–124.
5. Dymond, P.W. On nondeterminism in parallel computation. *Theor. Comput. Sci.* **1986**, *47*, 111–120. [CrossRef]
6. Burgin, M. Nonlinear Phenomena in Spaces of Algorithms. *Int. J. Comput. Math.* **2003**, *80*, 1449–1476. [CrossRef]
7. Kleene, S. Constructive and Non-constructive Operations. In Proceedings of the International Congress of Mathematicians, Edinburg, UK, 14–21 August 1958; Cambridge University Press: Cambridge, UK, 1960.
8. Burgin, M. Reflexive Calculi and Logic of Expert Systems. In *Creative Processes Modeling by Means of Knowledge Bases*; Institute of Mathematics: Sofia, Bulgaria, 1992; pp. 139–160. (In Russian)
9. Burgin, M. Reflexive Turing Machines and Calculi. *Vychislitelnyye Syst. (Log. Methods Comput. Sci.)* **1993**, *148*, 94–116, 175–176.
10. Schroeder, M.J. Dualism of Selective and Structural Manifestations of Information in Modelling of Information Dynamics. In *Computing Nature*; SAPERE 7; Springer: Berlin, Germany, 2013; pp. 125–137.

11. Schroeder, M.J. From Proactive to Interactive Theory of Computation. In Proceedings of the 6th AISB Symposium on Computing and Philosophy: The Scandal of Computation—What is Computation? Exeter, UK, 2–5 April 2013; pp. 47–51.
12. Burgin, M. Processing information by symmetric inductive machines. In Proceedings of the IS4SI Summit Berkeley 2019—Where is the I in AI and the Meaning in Information, Berkeley, CA, USA, 2–7 June 2019.
13. Smith, B.C. Procedural Reflection in Programming Languages. Ph.D. Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, USA, 1982.
14. Demers, F.-N.; Malenfant, J. Reflection in logic, functional and object-oriented programming: A Short Comparative Study. In Proceedings of the IJCAI'95 Workshop on Reflection and Metalevel Architectures and their Applications in AI, Montreal, QC, Canada, 20–25 August 1995; pp. 29–38.
15. Chlipala, A. Ur: statically-typed metaprogramming with type-level record computation. In Proceedings of the ACM SIGPLAN 2010 Conference on Programming Language Design and Implementation (PLDI'10), Phoenix, AZ, USA, 22–28 June 2010; Volume 45, pp. 122–133.
16. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
17. Mandler, J.M. *Stories, Scripts, and Scenes: Aspects of Schema Theory*; Lawrence Erlbaum Associates: Hillsdale, NJ, USA, 1984.
18. Arbib, M.A. *Modules, Brains and Schemas, Formal Methods*; LNCS 3393; Springer: Berlin/Heidelberg, Germany, 2005; pp. 153–166.
19. Burgin, M. Mathematical Schema Theory for Modeling in Business and Industry. In Proceedings of the 2006 Spring Simulation Multi Conference (SpringSim'06), Huntsville, AL, USA, 2–6 April 2006; pp. 229–234.
20. Burgin, M.; Debnath, N. Reusability as Design of Second-Level Algorithms. In Proceedings of the ISCA 25th International Conference "Computers and their Applications" (CATA-2010), ISCA, Honolulu, HI, USA, 22–24 October 2010; pp. 147–152.
21. Burgin, M.; Gupta, B. Second-level Algorithms, Superrecursivity, and Recovery Problem in Distributed Systems. *Theory Comput. Syst.* **2012**, *50*, 694–705. [CrossRef]
22. Dehaene, S.; Changeux, J.P. Development of elementary numerical abilities: A neuronal model. *J. Cogn. Neurosci.* **1993**, *5*, 390–407. [CrossRef]
23. Dehaene, S. *The Number Sense: How the Mind Creates Mathematics*; Oxford University Press: New York, NY, USA, 1997.
24. Schroeder, M.J. Computing with Nature. *Proceedings* **2017**, *1*, 178. [CrossRef]
25. Dodig-Crnkovic, G.; Burgin, M. *Information and Computation*; World Scientific: New York, NY, USA, 2011.
26. Zuse, K. *Rechnender Raum*; Friedrich Vieweg&Sohn: Braunschweig, Germany, 1969.
27. Fredkin, E. Digital Mechanics. *Phys. D* **1990**, *45*, 254–270. [CrossRef]
28. Dodig-Crnkovic, G. Significance of Models of Computation from Turing Model to Natural Computation. *Minds Mach.* **2011**, *21*, 301–322. [CrossRef]
29. Burgin, M.; Dodig-Crnkovic, G. Information and Computation—Omnipresent and Pervasive. In *Information and Computation*; World Scientific: New York, NY, USA, 2011; pp. vii–xxxii.
30. Shiva, S.G. *Advanced Computer Architectures*; CRC Press: Boca Raton, FL, USA, 2005.
31. Haykin, S. *Neural Networks: A Comprehensive Foundation*; Macmillan: New York, NY, USA, 1994.
32. Kolmogorov, A.N. On the concept of algorithm. *Uspehi Mat. Nauk.* **1953**, *8*, 175–176.
33. Shönhage, A. Storage Modification Machines. *SIAM J. Comput.* **1980**, *9*, 490–508. [CrossRef]
34. Kleinberg, J.; Tardos, E. *Algorithm Design*; Pearson-Addison-Wesley: Boston, MA, USA, 2006.
35. Burgin, M.; Adamatzky, A. Structural machines and slime mold computation. *Int. J. Gen. Syst.* **2017**, *45*, 201–224. [CrossRef]
36. Burgin, M.; Adamatzky, A. Structural Machines as a Mathematical Model of Biological and Chemical Computers. *Theory Appl. Math. Comput. Sci.* **2017**, *7*, 1–30.
37. Burgin, M.; Mikkilineni, R.; Mittal, S. Knowledge processing as structure transformation. *Proceedings* **2017**, *1*, 212. [CrossRef]
38. Burgin, M. *Measuring Power of Algorithms, Computer Programs, and Information Automata*; Nova Science Publishers: New York, NY, USA, 2010.
39. Shepherdson, J.C.; Sturgis, H.E. Computability of Recursive Functions. *J. ACM* **1963**, *10*, 217–255. [CrossRef]

40. von Neumann, J. *Theory of Self-Reproducing Automata*; University of Illinois Lectures on the Theory and Organization of Complicated Automata, Edited and completed by Arthur W. Burks; University of Illinois Press: Urbana, IL, USA, 1949.
41. Minsky, M. *Computation: Finite and Infinite Machines*; Prentice-Hall: New York, NY, USA, 1967.
42. Hopcroft, J.E.; Motwani, R.; Ullman, J.D. *Introduction to Automata Theory, Languages, and Computation*; Addison Wesley: Boston, MA, USA, 2007.
43. Burgin, M. Periodic Turing Machines. *J. Comput. Technol. Appl. (JoCTA)* **2014**, *5*, 6–18.
44. Burgin, M. Inductive Cellular Automata. *Int. J. Data Struct. Algorithms* **2015**, *1*, 1–9.
45. Siegelman, H.T. *Neural Networks and Analog Computation: Beyond the Turing Limit*; Birkhauser: Berlin, Germany, 1999.
46. Blum, L.; Shub, M.; Smale, S. On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines. *Bull. Am. Math. Soc.* **1989**, *21*, 1. [CrossRef]
47. Blum, L.; Cucker, F.; Shub, M.; Smale, S. *Complexity and Real Computation*; Springer: New York, NY, USA, 1998.
48. Burgin, M.S. Inductive Turing Machines. *Not. Acad. Sci. USSR* **1983**, *270*, 1289–1293.
49. Burgin, M. Cluster Computers and Grid Automata. In Proceedings of the ISCA 17th International Conference "Computers and their Applications", International Society for Computers and their Applications, Honolulu, HI, USA, 13–15 August 2003; pp. 106–109.
50. Burgin, M. Interactive Hypercomputation. In Proceedings of the 2007 International Conference on Foundations of Computer Science (FCS'07), Las Vegas, NV, USA, 25–28 June 2007; CSREA Press: Las Vegas, NV, USA, 2007; pp. 328–333.
51. Burgin, M. Super-recursive Algorithms as a Tool for High Performance Computing. In Proceedings of the High Performance Computing Symposium, San Diego, CA, USA, 12–15 April 1999; pp. 224–228.


© 2020 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

big data and cognitive computing

MDPI

Article

# From data Processing to Knowledge Processing: Working with Operational Schemas by Autopoietic Machines

**Mark Burgin [1] and Rao Mikkilineni [2,*]**

[1] Department of Mathematics, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu
[2] Ageno School of Business, Golden Gate University, San Francisco, CA 94105, USA
* Correspondence: rmikkilineni@ggu.edu

**Abstract:** Knowledge processing is an important feature of intelligence in general and artificial intelligence in particular. To develop computing systems working with knowledge, it is necessary to elaborate the means of working with knowledge representations (as opposed to data), because knowledge is an abstract structure. There are different forms of knowledge representations derived from data. One of the basic forms is called a schema, which can belong to one of three classes: operational, descriptive, and representation schemas. The goal of this paper is the development of theoretical and practical tools for processing operational schemas. To achieve this goal, we use schema representations elaborated in the mathematical theory of schemas and use structural machines as a powerful theoretical tool for modeling parallel and concurrent computational processes. We describe the schema of autopoietic machines as physical realizations of structural machines. An autopoietic machine is a technical system capable of regenerating, reproducing, and maintaining itself by production, transformation, and destruction of its components and the networks of processes downstream contained in them. We present the theory and practice of designing and implementing autopoietic machines as information processing structures integrating both symbolic computing and neural networks. Autopoietic machines use knowledge structures containing the behavioral evolution of the system and its interactions with the environment to maintain stability by counteracting fluctuations.

**Keywords:** knowledge; information; schema; knowledge processing; automaton; autopoietic machines; structural machines

**Citation:** Burgin, M.; Mikkilineni, R. From data Processing to Knowledge Processing: Working with Operational Schemas by Autopoietic Machines. *Big Data Cogn. Comput.* **2021**, *5*, 13. https://doi.org/10.3390/bdcc5010013

Received: 7 January 2021
Accepted: 2 March 2021
Published: 10 March 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Data are objects known or assumed as facts, forming the basis of reasoning or calculation. Data also contain factual information (such as the results of measurements or statistics) used as a basis for reasoning, discussion, or calculation. In the digital computing world, data are the carriers of information in the digital form, which can be transmitted or processed. Data can be naturally represented by named sets where the name may be a label, number, idea, text, process, and even a physical object. The named objects may be decomposed into knowledge structures with inter-object and intra-object relationships and associated behaviors, which can cause changes to their state, form, or content. The role of information-processing structures is to discern the relationships and behaviors of various entities composed of data and evolve their state, form, or content accordingly. Information in the strict sense includes a capacity to change structural elements in knowledge systems [1].

Conventional computers process data and provide a way to handle information stored in these data. However, a system cannot be intelligent if it cannot operate with knowledge [2]. The degree of intelligence depends on the ability to operate with knowledge. This is true both for natural and artificial systems. At the same time, intelligent systems work not directly with knowledge but employ knowledge representations of different

types, because knowledge per se is an abstract structure [3]. One of the most powerful and flexible forms of knowledge representations is schema.

The first known utilization of the term schema is ascribed to the outstanding philosopher Immanuel Kant, who did this in his philosophical works [4]. As an example, Kant defined the "dog" schema as a mental pattern of the figure of a four-footed animal, without restricting it to any particular instance from experience, or any possible picture that an individual can have [5]. Later, Head and Holmes employed the notion of a schema in neuroscience using body schemas in their studies of brain damage [6]. Bartlett applied the notion of a schema to the exploration of remembering [7]. One more principal approach to utilization of schemas in psychology was instigated by Jean Piaget. He studied cognitive development from the biological perspective describing it in terms of operation with schemas [8].

Informally, a *mental schema* is both a mental representation (descriptive knowledge) of the world and operational knowledge that determines action in the world.

Schemas are extensively utilized by people and computers for a diversity of purposes. Cognitive schemas are structures used to organize, enhance, and simplify knowledge of the world around us. For instance, behavior of a businessperson or a manager is based on a variety of cognitive schemas that suggest what to do in this or that situation. An important tool in database theory and technology is the notion of the database schema. It gives a general description of a database, is specified during database design, and is not expected to change frequently [9]. The arrival of the Internet and introduction of XML started the development of programming schema languages. There schemas are represented by machine-processable specifications defining the structure and syntax of metadata arrangements in programming languages. Researchers elaborated various XML schema languages [10,11]. XML schemas, in particular, serve as tools for describing the structure, content, and semantics of XML metadata. Specific schemas were elaborated for energy simulation data representation [12]. A particular concept of a schema has been quite common in programming, where it was formalized and widely utilized in theoretical and practical areas [13–16]. Dataflow schemas have been used for studies of parallel computations (cf., for example, [17]). One more kind of schemas, interactive schemas, is extensively used in psychology, theories of learning, and brain theory [18–20].

Schemas are studied and used in a variety of areas including neurophysiology, psychology, computer science, Internet technology, databases, logic, and mathematics. The reason for such high-level development and application is the existence of different types of schemas (in brain theory, cognitive psychology, artificial intelligence, programming, networks, computer science, mathematics, databases, etc.). Many knowledge structures, for example, algorithmic skeletons [21,22] or strategies, are special cases of operational schemas. To better understand human intellectual and practical activity (thinking, decision-making, and learning) and to build artificial intelligence, we have to be able to work with a variety of schema types. Such opportunities are provided by the mathematical schema theory developed by one of the authors [23–25], elements of which are presented in Section 2 of this paper.

There are three classes of schemas: operational, descriptive, and representation schemas. Here we consider only operational schemas. Other types of schemas are studied elsewhere. Schemas in general and operational schemas, in particular, are related to multigraphs. That is, the structure of an operational schema is an oriented multigraph.

In Section 3, we describe operations aimed at construction, transformation, and adaptation of schemas. Presenting some properties of these operations in the form of theorems, we give only outlines of proofs not to overload the work with an unnecessary formalism. In Section 4, we present a powerful theoretical model for operation with schemas, which is called a structural machine [26,27]. In Section 5, we discuss computing structures for operation with schemas. In Section 6, we present some conclusions.

In our exposition, we give simple examples to help the reader to better understand highly abstract structures studied in this paper, because our goal is the development

of flexible theoretical tools for the study and advancement of existing computing and network systems.

## 2. Schemas and Elements of Their Mathematical Theory

The mathematical theory of schemas unifies the variety of approaches to schemas in different areas. It provides a general mathematical concept of a schema on the level of generality that makes it possible to model by mathematical tools virtually any type of schema.

According to the typology of knowledge [3], there are three basic schema types:

- Operational schemas
- Descriptive or categorical schemas
- Representation schemas

Here we consider only operational schemas, which are simply called schemas for simplicity, and their processing, while the two other types and their processing are studied elsewhere.

Note that database schemas of all three types—conceptual, logical, and physical schemas—are descriptive schemas and are not considered in this paper.

There are three types of schema elements: *objects* (also called *nodes* or *vertices*), *ports*, and *ties* (also called *links*, *connections*, or *edges*). They belong to three classes:

- Object/node, port, and connection/edge *constants*.
- Object/node, port, and connection/edge *variables*.
- Objects/nodes, ports, and connections *with variables*.

Ports are devices through which information/data arrive (*output ports* or *outlets*) and is transmitted from the schema (*input ports* or *inlets*). There are constants, variables, and constants with variables ports. For instance, a constant port P can have the variable defining its capacity.

Operational schemas also contain such objects as automata and/or automata-valued variables as their nodes. For instance, it is possible to consider an object variable $T$ that can take any Turing machine as its value.

There are different links between objects/nodes. For instance, there are one-way links and two-way links. Each constant link has a fixed type, while link variables can take values in sets of links having different types, and some link variables can also take a void value. For instance, let elements $A$ and $B$ in a schema $Q$ are connected only by a link variable $L$. If in the instantiation $R$ of the schema $Q$, this variable takes the void value, it means that there are no links between $A$ and $B$ in $R$.

**Definition 2.1.** *A port schema B (also called a schema with ports) is the system of three sets, three multisets, and three mappings having the following form*:

$$B = (A_B, \mathrm{V}_{\mathrm{N}B}, P_B, \mathrm{V}_{\mathrm{P}B}, C_B, \mathrm{V}_{\mathrm{C}B}, p_{IB}, c_B, p_{EB})$$

Here, $A_B$ is the set of all objects (e.g., automata) from the schema $B$; the multiset $\mathrm{V}_{\mathrm{N}B}$ consists of all object variables (e.g., automaton variables) from $B$; the set $C_B$ is the set of all connections/links between the objects and object variables in the schema $B$; the multiset $\mathrm{V}_{\mathrm{C}B}$ consists of all link variables, i.e., variables that take values in the links between the objects and object variables in the schema $B$; the set $P_B = P_{IB} \cup P_{EB}$ (with $P_{IB} \cap P_{EB} = \emptyset$) is the set of all ports of the schema $B$, $P_{IB}$ is the set of all ports (called *internal ports*) of the automata from $A_B$, and $P_{EB}$ is the set of *external ports* of $B$, which are used for interaction of $B$ with different external systems and are divided into the input and output ports; the multiset $\mathrm{V}_{\mathrm{P}B}$ consists of all port variables from $B$ and is divided into two disjunctive sub-multisets $\mathrm{V}_{\mathrm{P}Bin}$ that consists of all variable inlets from $B$, and $\mathrm{V}_{\mathrm{P}Bout}$ consists of all outlets from the schema $B$; $p_{IB}$: $P_{IB} \cup \mathrm{V}_{\mathrm{P}B} \to A_B \cup \mathrm{V}_{\mathrm{N}B}$ is a (variable) total function, called the *internal port assignment function*, that assigns ports to automata; $c_B$: $C_B \cup \mathrm{V}_{\mathrm{C}B} \to ((P_{Ibout} \cup \mathrm{V}_{\mathrm{P}Bout}) \times (P_{Ibin} \cup \mathrm{V}_{\mathrm{P}Bin})) \cup (P'_{IBin} \cup \mathrm{V}'_{\mathrm{P}Bin}) \cup (P'_{IBout} \cup \mathrm{V}'_{\mathrm{P}Bout})$ is a (variable) function, called the *port-link adjacency*

*function*, that assigns connections to ports, where $P'_{IGin}$, $P''_{Igout}$, $V'_{PBin}$ , and $V'_{PBout}$ are disjunctive copies of $P'_{IGin}$, $P''_{Igout}$, $V'_{PBin}$, and $V'_{PBout}$, correspondingly; and $p_{EB}$: $P_{EB} \cup V_{PB} \to A_B \cup P_{IB} \cup C_B \cup V_{NB} \cup V_{PB} \cup V_{CB}$ is a (variable) function, called the *external port assignment function*, that assigns ports to different elements from the schema $B$.

Operational schemas are intrinsically related to automata, because in a general case, an instantiation of an operational schema is a grid automaton. In some sense, operational schemas are grid automata with variables. Grid automata are studied in [28,29]. However, for the sake of easier understanding, here we provide examples of relations between schemas and automata, not for grid automata, but for simpler and better-known automata, which are particular cases of grid automata.

**Example 2.1.** *A schema of transducer (information processing device) hardware from* [30] *describes an arbitrary transducer*.

In a general case, the information processing device hardware consists of three main components: an input device, information processor, and output device. This is described by the following schema:

$$Tr = (V_{NTr}, P_{Tr}, V_{PTr}, C_{Tr}, V_{CTr}, p_{ITr}, c_{Tr}, p_{ETr}) \quad (1)$$

$V_{NTr}$ = {ID, IP, OD}, where ID is a variable that takes input devices as values, IP is a variable that takes information processors as values, and OD is a variable that takes output devices as values.

$P_{Tr}$ consists of ports from the three device variables, each of which has one input port and one output port.

$V_{PTr}$ consists of port variables attached to the three device variables.

$C_B$ is the set of all connections/links between ID, IP, and OD.

$V_{CTr}$ is the set of all connection/link variables, which connect ID, IP, and OD.

$c_{Tr}$ is the adjacency function.

An informal visual representation of this schema $Tr$ is given in Figure 1.

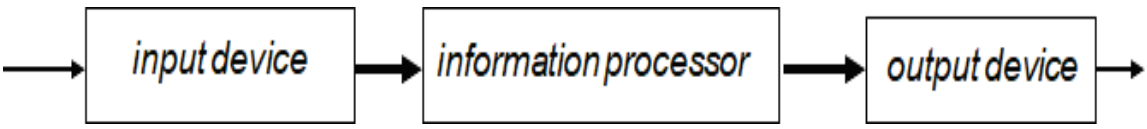



**Figure 1.** A schema of transducer hardware.

Here the variable ID (input device) can take such values as a keyboard, microphone, camera, mouse, or several of these devices together. The variable OD (output device) can take such values as a screen, monitor, printer, plotter, projector, speaker, TV set, headphones, or several of these devices together. The variable IP (information processor) can take such values as a physical system, such as a computer or cell phone, or an abstract automaton such as a Turing machine or register machine.

**Example 2.2.** *A schema of parallel information processing device hardware describes two independent information processing devices, which can work in the parallel mode*.

In a general case, parallel information processing hardware consists of the following main components: two or more input devices, two or more information processors, and two or more output device. The simplest case of such hardware, with two of each device, is described by the following schema:

$$PTr = (V_{NPTr}, P_{PTr}, V_{PTr}, C_{PTr}, V_{CPTr}, p_{IPTr}, c_{PTr}, p_{EPTr})$$

$V_{NPTr}$ = {ID1, IP1, OD1, ID2, IP2, OD2}, where ID1 and ID2 are variables that take input devices as values, IP1 and IP2 are variables that take information processors as values, and OD1 and OD2 are variables that takes output devices as values. These variables are similar to the variables from Example 2.1 and can take similar values.

$P_{PTr}$ consists of ports from the device variables, each of which has one input port and one output port.

$V_{PTr}$ consists of port variables attached to the three device variables.

$C_{PTr}$ is the set of all connections/links between ID1, IP1, OD1, ID2, IP2, and OD2.

$V_{CPTr}$ is the set of all connection/link variables, which connect ID1, IP1, OD1, ID2, IP2, and OD2.

$c_{PTr}$ is the adjacency function.

An informal visual representation of this schema *PTr* is given in Figure 2.

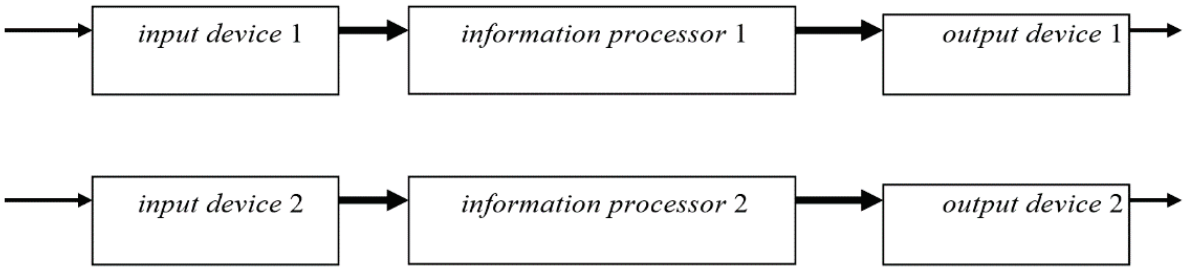



**Figure 2.** A schema of parallel information processing device hardware.

**Definition 2.2.** *A basic schema R (also called a schema without ports) has the same organization as the port schema, only without ports. Thus, it consists of two sets, two multisets, and one mapping, i.e.,* $R = (A_R, V_{NR}, C_R, V_{CR}, c_R)$.

There is a natural correspondence between basic schemas and port schemas. On the one hand, taking a port schema *R*, it is possible to delete all ports and to change all links between ports by the links between the elements of the schema *R*, to which these ports belong. On the other hand, taking a basic schema *Q*, it is possible to attach ports to the elements of the schema *Q*, to which links are attached in *Q*.

**Example 2.3.** *A schema of a Turing machine describes the hardware of an arbitrary Turing machine.*

A conventional Turing machine consists of three main components: a *control device*, *information processor*, and *memory* (*tape*). This is described by the following schema:

$$Tm = (V_{NTm}, C_{Tm}, c_{Tm})$$

$V_{NTm}$ = {cd, h, m}, where cd is a variable that takes values in accepting finite automata, h is a variable that takes computing finite automata as values, and m is a variable that takes different types of tapes, e.g., one-dimensional, two-dimensional, or *n*-dimensional tapes, as values.

$C_{Tm}$ is the set of all connections/links between cd, h, and m. That is, cd is connected to h, while h is connected to one cell in m.

$c_{Tm}$ is the adjacency function between elements/variables and links.

An informal visual representation of this schema *Tm* is given in Figure 3.

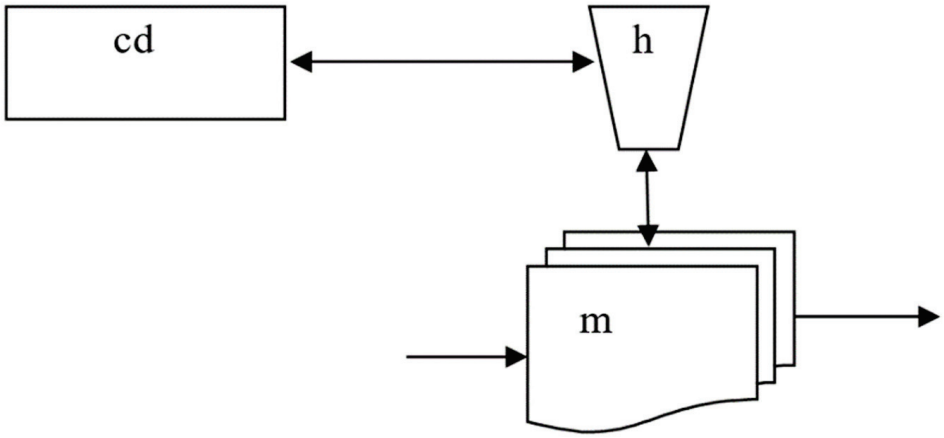



**Figure 3.** A schema of a Turing machine.

**Example 2.4.** *A schema of a simple inductive Turing machine describes the hardware of an arbitrary simple inductive Turing machine* [30].

A simple inductive Turing machine consists of five main components: a *control device*, *information processor*, *input register*, *output register*, and *working memory*. This is described by the following schema:

$$ITm = (V_{NITm}, C_{ITm}, c_{ITm})$$

$V_{NTr}$ = {**ir, or, cd, h, m**}, where **cd** is a variable that takes values in accepting finite automata, **h** is a variable that takes computing finite automata as values, **ir** denotes the input register, **or** denotes the output register, and **m** is a variable that takes different types of tapes, e.g., one-dimensional, two-dimensional, or *n*-dimensional tapes, as values.

$C_{ITm}$ is the set of all connections/links between **cd**, **h**, and **m**. That is, the element **cd** is connected to the element **h**, while **h** is connected to one cell in m.

$c_{ITm}$ is the adjacency function between elements/variables and links.

Note that in the Turing machine schema and the simple Turing machine schema, the memory is not changing in the process of computation. However, in the higher order inductive Turing machine schema, the memory can be transformed in the process of computation [30]. This difference is not displayed in the schema considered here but can be exhibited in a more detailed schema.

An informal visual representation of this schema *ITm* is given in Figure 4.

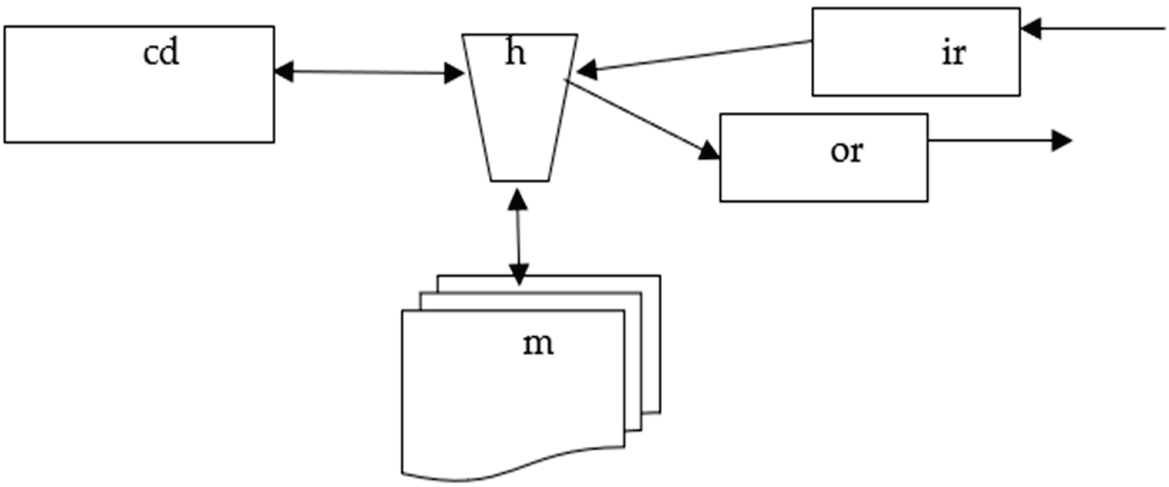
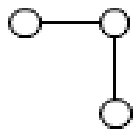



**Figure 4.** A schema of an inductive Turing machine.

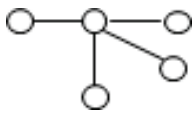
The structure of schema is an oriented multigraph, which is called the *grid*.

**Definition 2.3.** *The grid G(P) of a (basic or port) schema P is the generalized oriented multigraph that has exactly the same vertices and edges as P with the adjacency function* $c_{G(B)}$ *equal to* $c_B$.

**Example 2.5.** *The grid of the schema Tm of a Turing machine.*



**Example 2.6.** *The grid of the schema ITm of an inductive Turing machine.*



It is possible to read more about different types of schemas, their properties, and mathematical models in [23–25] and other publications on the mathematical theory of schemas.

## 3. Operations with Schemas

We discern three types of operations with schemas—large-scale, middle-scale, and local operations.

Types of large-scale operations with schemas:

1. Schema processing
2. Schema utilization
3. Schema transmission

Large-scale operations are formed from middle-scale operations. For schema processing, there are the following types of middle-scale operations with schemas:

1. Creation/elaboration of schemas
2. Transformation of schemas
3. Decomposition/termination of schemas

Utilization of schemas includes the following middle-scale operations (stages):

1. Creation of a schema from the existing material
2. Formation of a schema instance (instantiation)
3. Application of the schema instance

Middle-scale operations are formed from local operations. For schema transformation, there are the following types of local operations with schemas:

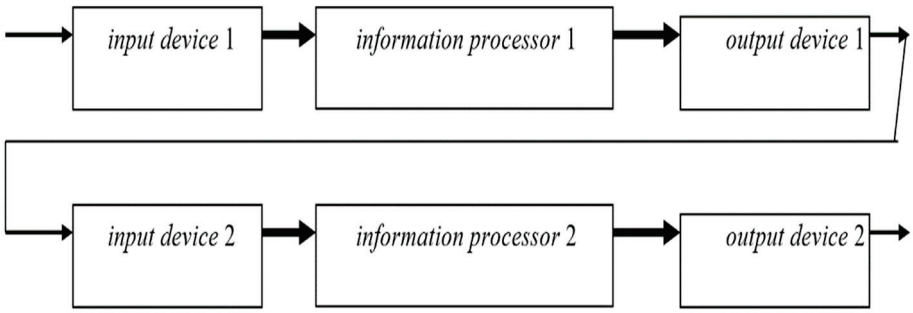
1. *Composition/aggregation* of several schemas
2. *Monotransformation* when one schema is changed
3. Coordinated transformation of several schemas—*polytransformation*

Let us consider some of them:

1. *Outside clutching* (also called external composition) con(A, B) of schemas A and B is composed by correct attaching some external ports of the schemas A and B to one another.

An example of outside clutching is sequential composition, which is the most popular operation in computing, where the output port of one computing system is connected to the input port of another computing system. As a result, the output of the first computing system is used as the input of the second computing system.

**Example 3.1.** *Outside clutching (external composition) con(Tr1, Tr2) of the schemas Tr1 and Tr2, each of which is isomorphic to the schema Tr (cf. Example 2.1).*




**Theorem 1.** *If the connection of the external ports remains the same, then the outside clutching (external composition) of schemas is a commutative operation, i.e., con(A, B) = con(B, A) for any schemas A and B.*

Proof is done by induction on the number of the new connections and is based on the following properties of schemas. The external ports of the schemas A and B become internal ports in their outside clutching con(A, B). If *p* and *q* are internal ports of a schema, then the connection (*p*, *q*) is equal to (the same as) the connection (*q*, *p*).

2. *Mixed clutching* (also called incorporation) inc(A, B) of schemas A and B is composed by correct attaching the external ports of the schema B to the internal ports of the schema A.

An example of mixed clutching is adding an additional chip to a computer or embedded device.

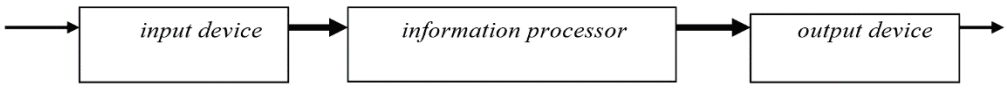
3. *Inside clutching* icl(A, B) of schemas A and B is composed by correct attaching the internal ports of schemas A and B.

An example of mixed clutching is adding a feedback link, which goes from the output registers to the input registers.

4. *Substitution:*
   a. *Node substitution* sub(A, *a*; B) of a node *a* in the schema A by the schema B is constructed by eliminating the node *a* from the schema A and connecting the external links of the node *a* to the appropriate external ports of the schema B.
   b. *Link substitution* sub(A, l; B) of a link l in the schema A by the schema B is constructed by eliminating l from the schema A and connecting the input ports of the schema B to the source of l and the output ports of B to the target of l.
   c. *Component substitution* sub(A, C; B) of a component C in the schema A by the schema B is constructed by connecting external links of the component C to appropriate external ports of the schema B.

**Example 3.2.** *The node substitution sub(Tr, IP; Tm) of the variable IP in the schema Tr of a transducer by the schema Tm of a Turing machine (cf. Example 2.3).*

The schema *Tr*




An informal visual representation of the result of the substitution sub(*Tr*, IP; *Tm*) is given in Figure 5.

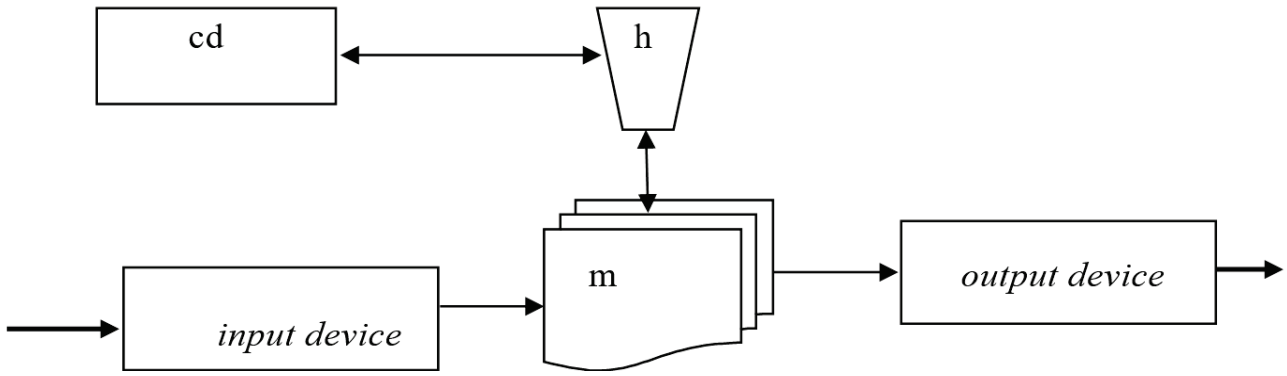



**Figure 5.** A schema of the transducer hardware.

Let us consider two schemas $A$ and $B$. Properties of substitution imply the following result:

**Lemma 1.** *All elements in the schemas A and B but the element a are not changing in the substitution sub(A, a, B).*

In contrast to clutching, substitution is not commutative. At the same time, under definite conditions, substitution is associative.

**Theorem 2.** *Substitution of different elements in a schema is an associative operation, i.e., if a and b are different elements of a schema A, then sub(sub(A, a, B), b, C) = sub(sub(A, b, C), a, B) for any schemas C and B.*

Proof is done by induction on the number of the new connections and is based on the following properties of schemas. When the substitution of the schema $C$ in the schema sub($A$, $a$, $B$), then only the element $b$ is changed, and this element does not belong to the schema $B$. When the substitution of the schema $B$ in the schema sub($A$, $b$, $C$), then only the element $a$ is changed, and this element does not belong to the schema $C$.

5. *Free composition* $A{*}B$ of schemas $A$ and $B$ is constructed by taking unions of their elements, i.e., $A_{A*B} = A_A \cup A_B$, $V_{\mathrm{N}(A*B)} = V_{\mathrm{N}A} \cup V_{\mathrm{N}B}$, $P_{A*B} = P_A \cup P_B$, $V_{\mathrm{P}(A*B)} = V_{\mathrm{P}A} \cup V_{\mathrm{P}B}$, $C_{A*B} = C_A \cup C_B$, and $V_{\mathrm{C}(A*B)} = V_{\mathrm{C}A} \cup V_{\mathrm{C}B}$.

**Example 3.3.** *The schema PTr (cf. Example 2.2) is a free composition of the schemas Tr1 and Tr2, each of which is isomorphic to the schema Tr (cf. Example 2.1).*

**Theorem 3.** *Free composition of schemas is a commutative and associative operation.*

Proof of commutativity is similar to the proof of Theorem 1 and proof of associativity is similar to the proof of Theorem 2.

## 4. Structural Machines as Schema Processors

To be able to efficiently process knowledge, a computer or network system must have knowledge-oriented architecture and assembly of operations. The most powerful and at the same time, flexible, model of computing automata is a structural machine [27]. It provides architectural and operational means for operation with schemas.

For simplicity, we consider structural machines of the first order, which work with first-order structures.

**Definition 4.1.** *A first-order structure is a triad of the following form*:

$$\boldsymbol{A} = (A, r, R)$$

In this expression, we have the following:

- Set $A$, which is called the *substance* of the structure $\boldsymbol{A}$ and consists of elements of the structure $\boldsymbol{A}$, which are called *structure elements* of the structure $\boldsymbol{A}$
- Set $R$, which is called the *arrangement* of the structure $\boldsymbol{A}$ and consists of relations between elements from $A$ in the structure $\boldsymbol{A}$, which have the first order and are called *structure relations* of the structure $\boldsymbol{A}$
- The *incidence relation* $r$, which connects groups of elements from $A$ with the names of relations from $R$

Lists, queues, words, texts, graphs, directed graphs, mathematical and chemical formulas, tapes of Turing machines, and Kolmogorov complexes are particular cases of structures of the first order that have only unary and binary relations. Note that labels, names, types, and properties are unary relations.

**Definition 4.2.** *A structural machine M is an abstract automaton that works with structures of a given type and has three components*:

- The *unified control device* $C_M$ regulates the state of the machine $M$.
- The *unified processor* $P_M$ performs transformation of the processed structures and its actions (operations) depend on the state of the machine $M$ and the state of the processed structures.
- The *functional space* $Sp_M$ is the space where operated structures are situated.

The functional space $Sp_M$, in turn, consists of three components:

- The *input space* $In_M$, which contains the input structure.
- The *output space* $Out_M$, which contains the output structure.
- The *processing space* $PS_M$, in which the input structure(s) is transformed into the output structure(s).

We assume that all structures—the input structure, the output structure, and the processed structures—have the same type. In particular, the functional, input, output, and processing spaces have definite topologies defined by the systems of neighborhoods [26].

Computation of a structural machine $M$ determines the *trajectory of computation*, which is a tree in a general case and a sequence when the computation is deterministic and performed by a single processor unit.

There are two forms of functional spaces, $Sp_M$ and $USp_M$:

- $Sp_M$ is the set of all structures that can be processed by the structural machine $M$ and is called a *categorical functional space.*
- $USp_M$ is a structure for which all structures that can be processed by the structural machine $M$ are substructures and is called a *universal functional space.*

There are three basic types of unified control devices:

- It can be one *central control device*, which controls all processors of the structural machine.
- It can consist of *cluster control devices*, each of which controls a cluster of processors in the structural machine.
- It can consist of *individual control devices*, each of which controls a single processor in the structural machine.

There are three basic types of unified processors:

- A *localized processor* is a single abstract device, which consists of one or several *processor units* or *unit processors* functioning as a unified whole.
- A *distributed cluster processor* consists of unit processors (processor units) from a cluster, which performs definite functions in a structural machine.

- A *distributed total processor* consists of a system of all unit processors (processor units) from a structural machine.

It is possible to treat a localized processor as a singular unit processor although it can be constructed of several processor units, which are moving together processing information. Examples of distributed cluster processors are systems of processors that perform parallel or pipeline transformations of data in the memory.

Structural machines can process information in different modes [31]. This brings us to three kinds of structural machines:

- *Recursive structural machines*, in which all processors work in the recursive mode.
- *Inductive structural machines*, in which all processors work in the inductive mode.
- *Combined structural machines*, in which some processors work in the recursive mode, while other processors work in the inductive mode.

It is proved that for any Turing machine and thus for any recursive algorithm $A$, there is an inductive Turing machine $M$ that simulates functioning of $A$ [30]. Similar reasoning gives us the following result:

**Theorem 4.** *For any recursive or combined structural machine R, there is an inductive structural machine Q that simulates functioning of R.*

Properties of inductive Turing machines [30] also imply the following result:

**Theorem 5.** *Inductive and combined structural machines are essentially more powerful than recursive structural machines.*

Architectural and functional features of structural machines provide diverse possibilities to perform operations with schemas. For instance, a singular unit processor can move from one location to another location in a schema, acting upon the operated schema. Another option is parallel or concurrent operation of several unit processors performing the necessary transformations of the operated schema.

## 5. Computing Structures for Operation with Schemas

Digital information processing systems of the current generation are composed of computing structures that are stored program implementations of the Turing machines. As mentioned in this paper, they are designed to transform one state of the data structure, which is a representation of the arrangement, relationships, and contents of data, to another state based on a specified algorithm. The data structure consists of the knowledge about various data elements and the relationships between them. The behaviors of how events alter the data structures are captured in the algorithm (encoded as a program) and are executed to change the state of the knowledge structure using the CPU. Digital computing structures use programming languages, which operate on a variety of data structures such as characters, integers, floating-point real number values, enumerated types (i.e., a small set of uniquely named values), arrays, records (also called tuples or structs), unions, lists, streams, sets, multisets, stacks, queues, double-ended queues, trees, general graphs, etc. In addition, word processors, such as Word or TeX, work with various geometrical shapes, figures, and pictures.

Figure 6 shows the schema and the data structure evolution of a Turing machine stored program control implementation. The schema represents the control processor that executes [30] the infware, the information processor which provides the hardware (CPU and memory) and the software (the algorithms operating on the data structures.) The memory provides both the program and the data for processing.

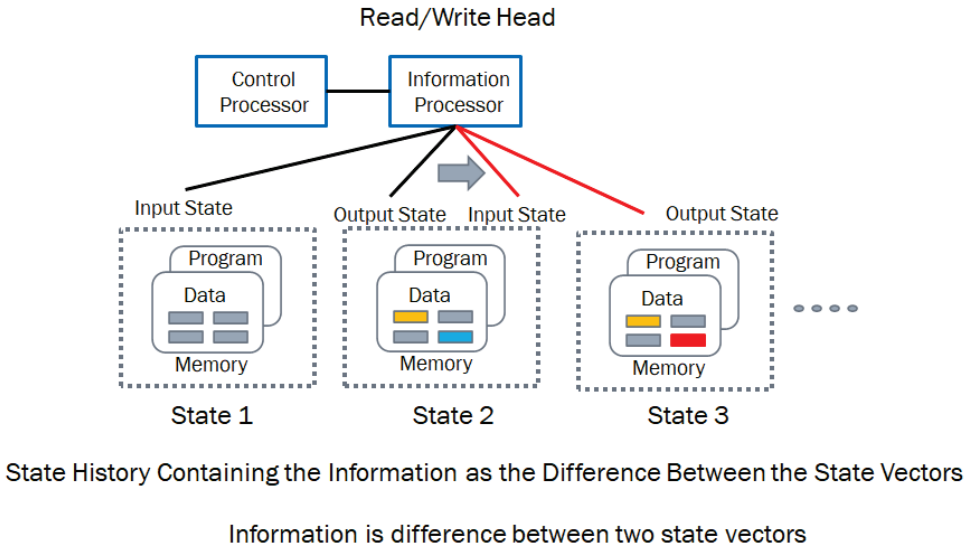



**Figure 6.** Data and the program stored in the computer memory are processed by the CPU in the information processor.

In essence, information is the change between the data structures from one instant to another, and information processing consists of physical structures that execute the behaviors transforming the data structures from one state to another with a read – compute – write cycle of operations. The digital computing structure requires hardware (the processor and memory) and software (which provides the algorithm and the data structure). In addition, we need the knowledge to configure, monitor, and operate the digital computing structure to execute the algorithm, which is called the infware [30]. The current state of the art requires a third party to execute the infware (an operator or other digital automata), which is given this knowledge to configure, monitor, and operate both hardware and software. The limitations of this architecture are threefold:

- Reacting to large fluctuations in the demand for resources or the availability of resources in a widely distributed computing structure executing on different provider hardware and software increases complexity and cost of end-to-end operational visibility and control while increasing reaction latency.
- When the distributed components are communicating asynchronously, data consistency, availability, and partitioning cause problems for executing non-stop high-reliability applications at scale without service disruption.
- Insufficient scalability, especially in processing so-called big data, and the widely distributed nature of the access to both the sources and consumers of data necessitate pushing information processing closer to the edge.

In this section we discuss how the use of new theory about knowledge structures, triadic automata, and structural machines allow us to design a new class of distributed information processing structures that use infware containing hierarchical intelligence to model, manage, and monitor distributed information processing hardware and software components as an active graph representing a network of networked processes. The infware, i.e., the processed information in the form of data structures, uses the operations on the schema representing the hardware, software, and their evolutionary behaviors as dynamic knowledge structures. An autopoietic system [32,33] implemented using triadic structural machines, i.e., structural machines working as triadic automata, is capable "of regenerating, reproducing, and maintaining itself by production, transformation, and destruction of its components and the networks of processes downstream contained in them." The autopoietic machines discussed in this paper, which operate on schema containing knowledge structures, allow us to deploy and manage non-stop and highly reliable computing structures at scale independent of whose hardware and software are used. Figure 7

shows the structural machine operating on the knowledge structure in the form of an active graph in contrast to a data structure in the Turing machine implementation, which is a linear sequence of symbols.

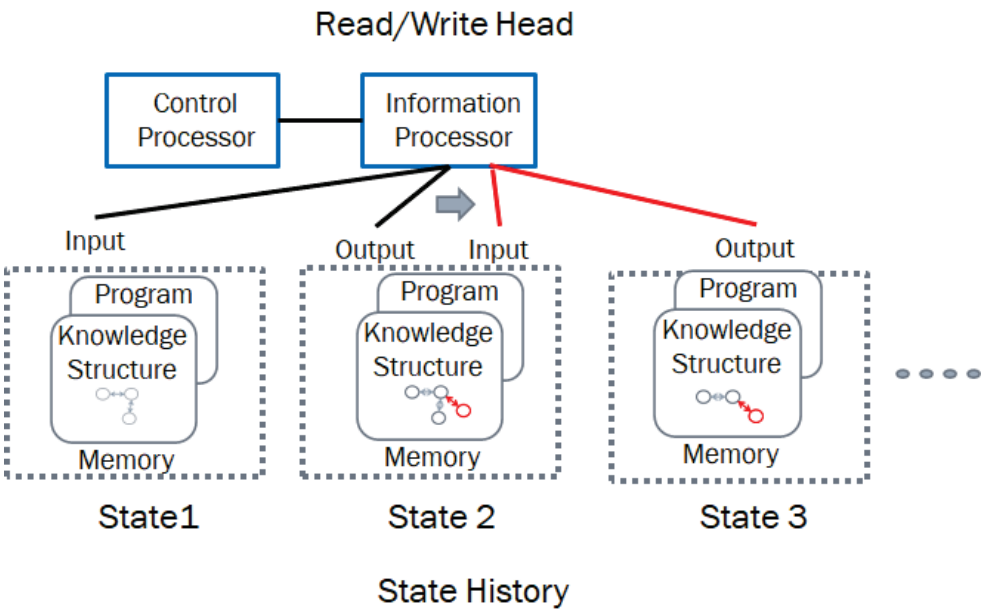



**Figure 7.** The schema with a triadic automaton represents a knowledge structure containing various object, inter-object, and intra-object relationships and behaviors, which emerge when an event occurs, changing the objects or their relationships.

It is important to emphasize the differences between data, data structures, knowledge, and knowledge structures. Data are mental or physical "observables" represented as symbols. Data structures define patterns of the relationships between data items. Knowledge is a system that includes data enhanced by relations between the data and the represented domain. Knowledge structures include data structures abstracted to various systems and inter-object and intra-object relationships and behaviors, which emerge when an event occurs, changing the objects or their relationships. The corresponding state vector defines a named set of knowledge structures, a representation of which is illustrated in Figure 8.

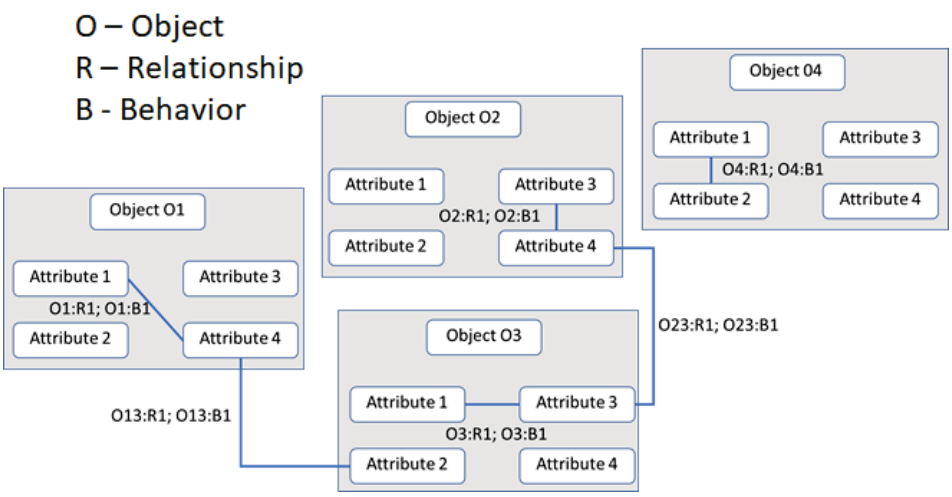



**Figure 8.** A knowledge structure modeling intra-object and inter-object behaviors.

Information processing in triadic structural machines is accomplished through operations on knowledge structures, which are graphs representing nodes, links, and their

behaviors. Knowledge structures contain named sets and their evolution containing named entities/objects, named attributes, and their relationships. Ontology-based models of domain knowledge structures contain information about known knowns, known unknowns, and processes for dealing with unknown unknowns through verification and consensus. Inter-object and intra-object behaviors are encapsulated as named sets and their chains. Events and associated behaviors are defined as algorithmic workflows, which determine the system's state evolution.

A named set chain of knowledge structures provides a genealogy representing the system state history. This genealogy can be treated as the deep memory and used for reasoning about the system's behavior, as well as for its modification and optimization. In Figure 8, the behaviors are executed when an event induces a change. For instance, when Attribute 1 in Object 1 changes the behavior when B1 is executed, it in turn may cause another behavior workflow. In this paper, we use the knowledge structures to represent the schema that models how both hardware and software are configured, monitored, and managed during the evolution of computations, executing an information processing structure such as a business application. Figure 9 shows the implementation of the autopoietic machine with hardware, software, and infware, which is used to execute a business application.

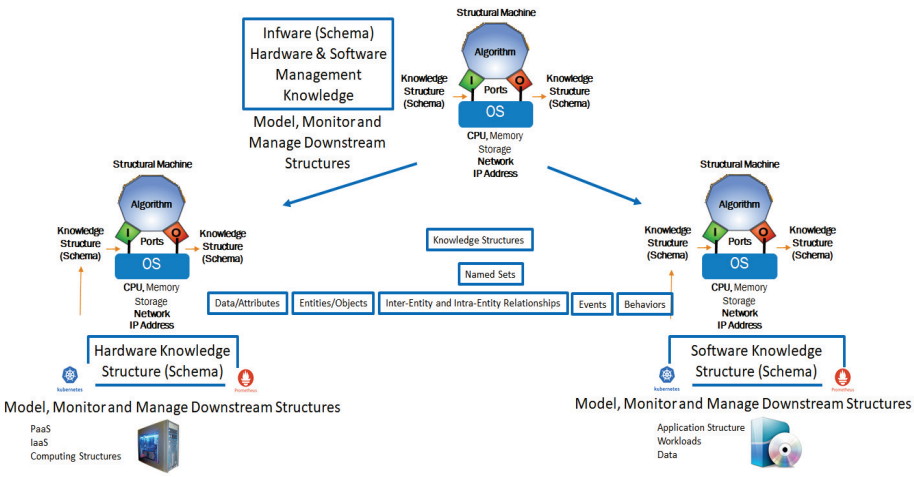



**Figure 9.** Schema managing infware, hardware, and software for the deployment, configuring, monitoring, and managing distributed application workloads on cloud resources.

The structural machine's control processor operates on the downstream knowledge structures to evolve their states based on the event flow. At the leaf node, the conventional Turing automata operate on the data structures to evolve traditional computing processes. The important innovation here is the regulatory overlay to discover/configure, monitor, and manage the traditional computing process evolution using the local knowledge of how the local IaaS and PaaS are configured, monitored, and managed while coordinating with global knowledge to optimize the end-to-end system behavior in the face of fluctuations.

The schema in Figure 9 describes the hardware components, software components, and their management characteristics defined by knowledge structures, forming an autopoietic automaton. The knowledge structures represent this schema in the form of a system of named sets containing various data elements, objects, or entities that are composed of the data, and their inter-object and intra-object relationships and behaviors associated with events that cause changes to the instances as time progresses [27].

## 6. Conclusions

In this paper, we describe theoretical and practical computational tools for working with knowledge structures, such as schemas, taking into account their inter- and intra-relationships and associated behaviors when events change the state. We use triadic structural machines described in this paper to perform operations with schemas. The result

is the design and implementation of autopoietic automata that are capable of regenerating, reproducing, and maintaining themselves by production, transformation, and destruction of its components and the networks of processes generated by them. Based on autopoietic automata and triadic structural machines working with schemas, we are currently building a system that models, monitors, and manages a distributed application deployed over multiple cloud resources to respond to large fluctuations in the demand for or the availability of computing resources in the system without disturbing the end to end transaction being delivered by the computing structure.

It is important to point out the difference between the current state of the art based on the classical computer science (constrained by the Church Turing thesis boundaries), which process information in the form of symbol sequences, and the current advancement of the field based on structural machines, which process information in the form of a dynamically-evolving network of networks represented by various schemas. As a result, we obtain autopoietic machines, which are capable of regenerating, reproducing, and maintaining themselves. The control processors in an autopoietic machine operate on the downstream graphs where a transaction can span across multiple distributed graphs, reconfiguring their nodes, links, and topologies based on well-defined pre-condition and post-condition transaction rules to address fluctuations, for example in resource availability or demand. The information processor of the machine, on the other hand, evolves the application workloads using the algorithms specified as programs operating on symbolic data structures.

Triadic structural machines provide the theoretical means for the design of autopoietic automata, whereas Turing machines cannot do this. The hierarchical control process overlay allows for the implementation of 4E (embedded, embodied, enactive, and extended) cognitive processes with downstream autonomous components interacting with each other and with their environment using system-wide knowledge-sharing, which allows global regulation to optimize the stability of the system as a whole based on memory and historical experience-based reasoning. This observation points out a major breakthrough in the way we will design future information processing systems with hierarchical intelligence and resilience at scale. We conclude this paper with the following observation from Leonardo da Vince: "He who loves practice without theory is like the sailor who boards ship without a rudder and compass and never knows where he may cast".

Other applications of the constructions and results of this paper are described in other papers of the authors and other researchers.

**Author Contributions:** Both authors have equally contributed to the article, its conceptualization, methodology, investigation, writing, original draft preparation, review, and editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The study does not report any data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
2. Minsky, M. *The Society of Mind*; Simon and Schuster: New York, NY, USA, 1986.
3. Burgin, M. *Theory of Knowledge: Structures and Processes*; World Scientific: New York, NY, USA; London, UK; Singapore, 2016.
4. Arbib, M.A. Modules, Brains and Schemas. In *Formal Methods in Software and Systems Modeling. Lecture Notes in Computer Science*; Kreowski, H.J., Montanari, U., Orejas, F., Rozenberg, G., Taentzer, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2005; Volume 3393.
5. Kant, I. *Critique of Pure Reason*; Smith, N.K., Translator; Macmillan: London, UK, 1929.
6. Head, H.; Holmes, G. Sensory Disturbances from Cerebral Lesions. *Brain* **1911**, *34*, 102–254. [CrossRef]
7. Bartlett, F.C. *Remembering*; Cambridge University Press: Cambridge, UK, 1932.
8. Piaget, J. *The Language and Thought of the Child*; Kegan Paul, Trench &Trubner: London, UK, 1926.

9. Elmasri, R.; Navathne, S.B. *Fundamentals of Database Systems*; Addison-Wesley: Reading, MA, USA, 2000.
10. Daum, B. *Modeling Business Objects with XML Schema*; Morgan Kauffman: Burlington, MA, USA, 2003.
11. Duckett, J.; Ozu, N.; Williams, K.; Mohr, S.; Cagle, K.; Griffin, O.; Francis Norton, F.; Stokes-Rees, I.; Tennison, J. *Professional XML Schemas*; Wrox Press Ltd.: Birmingham, UK, 2001.
12. Gowri, K. EnrXML–A Schema for Representing Energy Simulation Data. In Proceedings of the 7th International IBPSA Conference, Rio de Janeiro, Brazil, 13–15 August 2001; pp. 257–261.
13. Ershov, A.P. Mixed Computation in the Class of Recursive Program Schemata. *Acta Cybern.* **1980**, *4*, 19–23.
14. Ianov, Y.I. On the Equivalence and Transformation of Program Schemes. *Commun. ACM* **1958**, *1*, 8–12. [CrossRef]
15. Karp, R.M.; Miller, R.E. Parallel Program Schemata. *J. Comput. Syst. Sci.* **1969**, *3*, 147–195. [CrossRef]
16. Keller, R.M. Parallel Program Schemata and Maximal Parallelism. *JACM* **1973**, *20*, 514–537. [CrossRef]
17. Dennis, J.B.; Fossen, J.B.; Linderman, J.P. *Data Flow Schemes*; LNCS, 19; Springer: Berlin/Heidelberg, Germany, 1974.
18. Arbib, M.A. Schema Theory. In *The Encyclopedia of Artificial Intelligence*, 2nd ed.; Shapiro, S., Ed.; Wiley-Interscience: Hoboken, NJ, USA, 1992; pp. 1427–1443.
19. Price, E.; Driscoll, M. An Inquiry into the Spontaneous Transfer of Problem-Solving Skill. *Contemp. Educ. Psychol.* **1997**, *22*, 472–494. [CrossRef] [PubMed]
20. Solso, R.L. *Cognitive Psychology*; Allyn and Bacon: Boston, MA, USA, 1995.
21. Cole, M.I. *Algorithmic Skeletons*; MIT Press: Cambridge, MA, USA, 1989.
22. Schank, R.C.; Abelson, R.P. Scripts, Plans. In *Goals and Understanding*; Lawrence Erlbaum Associates: Hillsdale, NJ, USA, 1977.
23. Burgin, M. Mathematical Models in Schema Theory. *arXiv* **2005**, arXiv:cs/0512099.
24. Burgin, M. From Craft to Engineering: Software Development and Schema Theory. In Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering (CSIE 2009), WRI, Los Angeles, CA, USA, 31 March–2 April 2 2009; Volume 7, pp. 787–791.
25. Burgin, M. Mathematical Schema Theory for Network Design. In Proceedings of the ISCA 25th International Conference on Computers and their Applications (CATA-2010), ISCA, Honolulu, HI, USA, 24–26 March 2010; pp. 157–162.
26. Burgin, M. *Information Processing by Structural Machines, in Theoretical Information Studies: Information in the World*; World Scientific: New York, NY, USA; London, UK; Singapore, 2020; pp. 323–371.
27. Mikkilineni, R.; Burgin, M. Structural Machines as Unconventional Knowledge Processors. *Proceedings* **2020**, *47*, 26. [CrossRef]
28. Burgin, M. From Neural Networks to Grid Automata. In Proceedings of the IASTED International Conference Modeling and Simulation, Palm Springs, CA, USA, 9–11 April 2003; pp. 307–312.
29. Burgin, M. Cluster Computers and Grid Automata. In Proceedings of the ISCA 17th International Conference on Computers and Their Applications (CATA-2002), San Francisco, CA, USA, 4–6 April 2002.
30. Burgin, M. *Super-Recursive Algorithms*; Springer: New York, NY, USA; Berlin/Heidelberg, Germany, 2005.
31. Burgin, M. Super-Recursive Algorithms and Modes of Computation. In Proceedings of the 2015 European Conference on Software Architecture Workshops, Dubrovnik/Cavtat, Croatia, 7–11 September 2015; pp. 10:1–10:5.
32. Burgin, M.; Mikkilineni, R.; Phalke, V. Autopoietic Computing Systems and Triadic Automata: The Theory and Practice. *Adv. Comput. Commun.* **2020**, *1*, 16–35. [CrossRef]
33. Maturana, H.R.; Varela, F.J. *Autopoiesis and Cognition. The Realization of the Living*; Reidel: Dordrecht, The Netherlands, 1980.

proceedings

MDPI

Proceeding Paper

# Designing New Forms of Digital Autopoietic Machines †

**Rao Mikkilineni [1,*] and Mark Burgin [2]**

[1] Ageno School of Business, Golden Gate University, San Francisco, CA 94105, USA
[2] Mathematics Department, University of California, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu
* Correspondence: rmikkilineni@ggu.edu
† Presented at the Conference on Theoretical and Foundational Problems in Information Studies, IS4SI Summit 2021, online, 12–19 September 2021.

**Abstract:** The general theory of information (GTI) tells us that information is represented, processed and communicated using physical structures. The physical universe is made up of structures combining matter and energy. According to GTI, "information is related to knowledge as energy is related to matter." GTI also provides tools to deal with transformation of information and knowledge. We present here the application of these tools for the design of digital autopoietic machines with higher efficiency, resiliency and scalability than the information processing systems based on the Turing machines. We discuss the utilization of these machines for building autopoietic and cognitive applications in a multi-cloud infrastructure.

**Keywords:** general theory of information; named set; knowledge structure; structural machine; autopoietic machine; multi-cloud infrastructure

**Citation:** Mikkilineni, R.; Burgin, M. Designing New Forms of Digital Autopoietic Machines. *Proceedings* **2022**, *81*, 59. https://doi.org/10.3390/proceedings2022081059

Academic Editor: H. Paul Zellweger

Published: 21 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Over millennia, our view of the world has changed from a void with a large number of invisible and indivisible particles, which were called atoms, to a vacuum with vibrating strings in it. We still are trying to grasp Plato's ideas of forms/ideas with respect to physical reality. Our quest for a better worldview continues.

The general theory of structures (GTS) [1] offers a vision of the world as a whole in the form of the existential triad. It provides means to model and analyze the physical and mental systems in terms of abstract and ideal structures. Information and knowledge about these structures are related as energy and matter are related to each other, while the transformation rules help us to understand the behaviors of both physical and mental systems. Figure 1 shows the relationships between these structures.

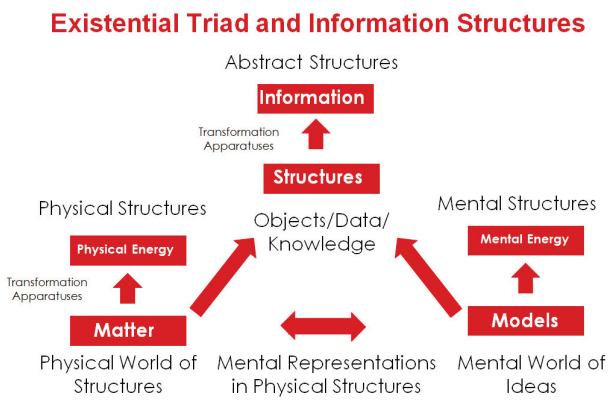



**Figure 1.** General theory of information and information structures.

The primary goal of the general theory of information (GTI) [2] is "to obtain a definition of information with the following properties. It has to be (1) sufficiently wide to

encompass a diversity of phenomena that exist under the common name *information*, (2) sufficiently flexible to reflect and organize all properties people ascribe to information, and (3) sufficiently efficient to provide a powerful tool for scientific exploration and practical usage".

In this paper, we present a scientific exploration and practical use of the tools provided by GTI by examining a complex system where function, structure and fluctuations play key roles in how the system behaves determined by the interactions among its constituent components and their environment. Digital computing structures composed of distributed and communicating software and hardware components fall into this category. A computing structure acts as a complex system where fluctuations in the demand for, and the availability of, resources required to execute the computations disturb its stability and performance. The fluctuations impact on the resiliency and efficiency of the structure grows as the scale of components increases. In addition, when the system is distributed, where the components are concurrent, asynchronous, and locally managed by autonomous infrastructure providers, the emergence of global behavior depends upon the nature and the strength of the fluctuations, and in the case of autopoiesis, is prone to instability. In this paper, we apply GTI tools to a system in which an application deployed in a heterogeneous multi-cloud environment with resources used by the application components is managed by local autonomous infrastructure management systems. We show a way to infuse autopoietic and cognitive behaviors into the application management to predict and manage instabilities by reconfiguring the structure without disrupting the stability of the system. Autopoiesis refers to a system with a well-defined identifier, which is capable of reproducing and maintaining itself. Cognition, on the other hand, is the ability to acquire and process information, apply knowledge, and change the circumstances. Figure 2 shows a distributed application in a multi-cloud infrastructure.

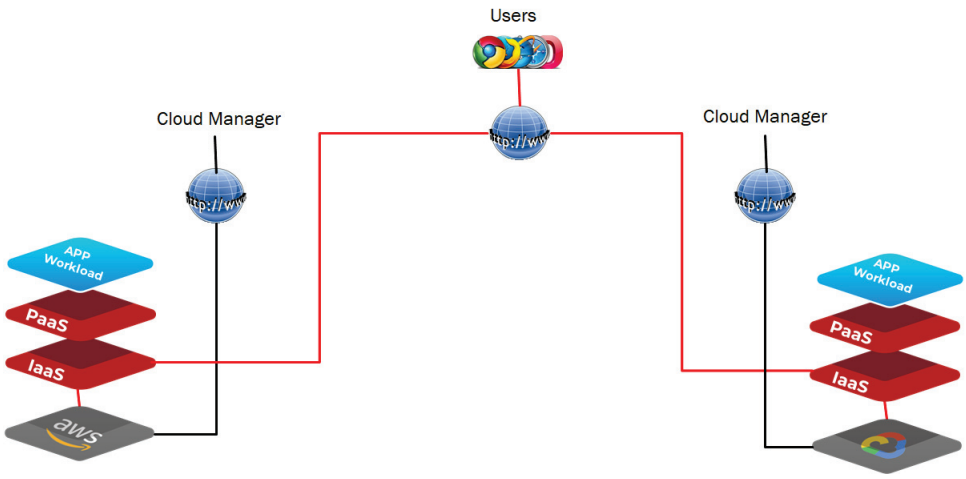



**Figure 2.** An application deployed in different clouds using different cloud managers. Cloud services infrastructure as a service (IaaS) and platform as a service (PaaS) are used by the cloud manager to deploy application workloads.

This paper has the following structure. In Section 2, we discuss GTI tools and their use for infusing autopoiesis and cognition into application management where the application acts as a self-managing unit with an identifier and manages its own stability with the knowledge of its distributed component deployment and their state of health along with the knowledge of best practices to deal with fluctuations. In Section 3, we discuss the evolution of application from Turing machine-based algorithms operating on strings of symbols to the structural machines operating on knowledge structures. In Section 4, we present some conclusions and suggestions for implementing the autopoietic machines with cognitive behaviors.

## 2. Tools Offered in the GTI

GTI and its various tools that assist in transforming information and knowledge are discussed in detail in the references mentioned here. Here we will briefly review the required tools and discuss their application. An information unit is conventionally described by the existence or non-existence (1 or 0) of a class or object that is physically observed or mentally conceived. The class is an abstract concept with attributes while an object is an instance of a class with an identifier, defined by two components—the object-state and object-behavior. An entity is either an object or a class. An attribute is a key value pair with an identifier (name) and a value associated with it. The attribute state is defined by its value. Information is related to knowledge and is defined by the relationships between various entities and their interactions (behaviors) when the values of the attributes change. A named set or a fundamental triad can represent epistemic connections between two entities as a knowledge structure derived from information (Figure 3).

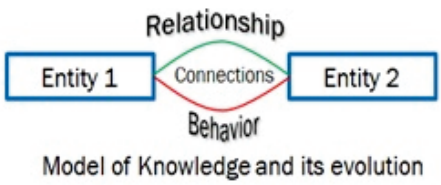



**Figure 3.** The fundamental triad as a knowledge structure derived from information.

A knowledge structure defines various triadic relationships between all the entities that are contained in a system. The knowledge structure is a state or instance of a knowledge schema and various operations are used to evolve the schema from one state to another. Specific instances of the knowledge structure schema are utilized for modeling the domain knowledge and process information changes as they evolve with changes in their entities and their attributes and behaviors.

The structural machine is an information processing mechanism that works with schemas of knowledge structures and performs operations on them to evolve information changes in the system from one instant to another when any of the attributes of any of the objects changes [3].

The structural machines surpass the Turing machines, which work only with such primitive structures as strings of symbols, by their representations of knowledge and the operations that process information. Triadic structural machines with an assortment of general and mission-oriented processors and other triadic automata enable autopoietic behaviors [4–6].

### *2.1. From Turing Machines to Structural Machines*

Structural machines process all kinds of structures including knowledge structures that incorporate domain knowledge in the form of entities, their relationships and process evolution behaviors as a network of networks with each node defining functional behaviors and links defining the information exchange (or communication). The operations on the knowledge structure schema define the creation, deletion, connection and reconfiguration operations based on control knowledge structures. They are agnostic to what the functions of the nodes are or what information is exchanged between them. This provides the composability of knowledge structures across domains in processing information. In contrast, the Turing machines process only strings of symbols, which can encode knowledge for the price of diminishing the efficiency of information processing. Therefore, the Turing machine operations are too simple to support composability across domains causing high complexity in processing information and evolving the knowledge.

### *2.2. Changing Systems Behaviors Using Functional Communication*

The architectural, instructional and behavioral changes are regulated by the knowledge structures and therefore by their impact on knowledge structures, functional communication or information exchange induces the architectural, instructional and behavioral changes. Changes are propagated through knowledge structures, enabling self-regulation of the system.

### *2.3. Triadic Automata and Autopoietic Behavior*

A triadic structural machine with hierarchical control processors provides the efficient and flexible theoretical means for the design of autopoietic automata allowing transformation and regulation of all three dimensions of information processing and system behavior—the physical, mental and structural dimensions. The control processors operate on the downstream information processing structures, where a transaction can span across multiple distributed components by reconfiguring their nodes, links and topologies based on well-defined pre-condition and post-condition transaction rules to address fluctuations, such as fluctuations in resource availability or demand.

### *2.4. Providing Global Optimization Using Shared Knowledge and Predictive Reasoning to Deal with Large Fluctuations*

The hierarchical control process overlay in the design of the structural machine, allows implementing 5E (embedded, embodied, enacted, elevated and extended) cognitive processes with downstream autonomous components interacting with each other and with their environment using system-wide knowledge-sharing, which allows global regulation to optimize the stability of the system as a whole based on memory and historical experience-based reasoning [7]. Downstream components provide sensory observations and control using both neural network and symbolic computing structures.

These insights allow us to design a new class of information processing systems with higher efficiency, resiliency and scalability in dealing with fluctuations going far beyond the capabilities possible for information processing systems based on the Turing machine model. To achieve these goals, we describe the utilization of this theory for building a self-managing federated edge cloud network deploying autopoietic federated AI applications to connect people, things, and businesses for enabling global communication, collaboration and commerce with high reliability, performance, security, and regulatory compliance. The next section discusses the infusion of autopoietic and cognitive behaviors in the application considered above.

## 3. Application of the Tools to Design a New Class of Digital Autopoietic Machines with Cognitive Behaviors

Autopoietic machines are built using a knowledge network with knowledge nodes and information sharing links between them. The knowledge nodes are wired together and fire together to manage the behavioral changes in the system [5–7]. Each knowledge node contains hardware, software and "infware" that manage the information processing and communication structures within the node. The infware of a system consists of diverse information carriers specifying how to discover, configure, monitor and manage the hardware, software and other infware to maintain their state evolution based on externally infused knowledge such as business requirements dealing with system availability, performance, security, privacy and regulatory compliance. There are three types of knowledge nodes depending on the nature of the infware:

1. Autopoietic Functional Node (AFN) provides autopoietic component information processing services. Each node executes a set of specific functions based on the inputs and provides outputs that other knowledge nodes utilize.
2. Autopoietic Network Node (ANN) makes available operations on a set of knowledge nodes to configure, monitor, and manage their behaviors based on the group level objectives.

3. Digital Genome Node (DGN) is a system-level node that configures a set of autopoietic sub-networks, monitors them and manages them based on system-level objectives.

Each knowledge node is specialized according to its infware defining the knowledge structures that model downstream entities/objects, their relationships and behaviors which are executed using appropriate software, and hardware. The infware contains the knowledge to obtain resources, configure, execute, monitor and manage the downstream components based on the node level objectives.

Figure 4 shows the functional, network and digital nodes with knowledge structures and infware managing various functions that hardware and software provide. There are two network nodes which provide the knowledge structures that manage autopoietic and cognitive behaviors.

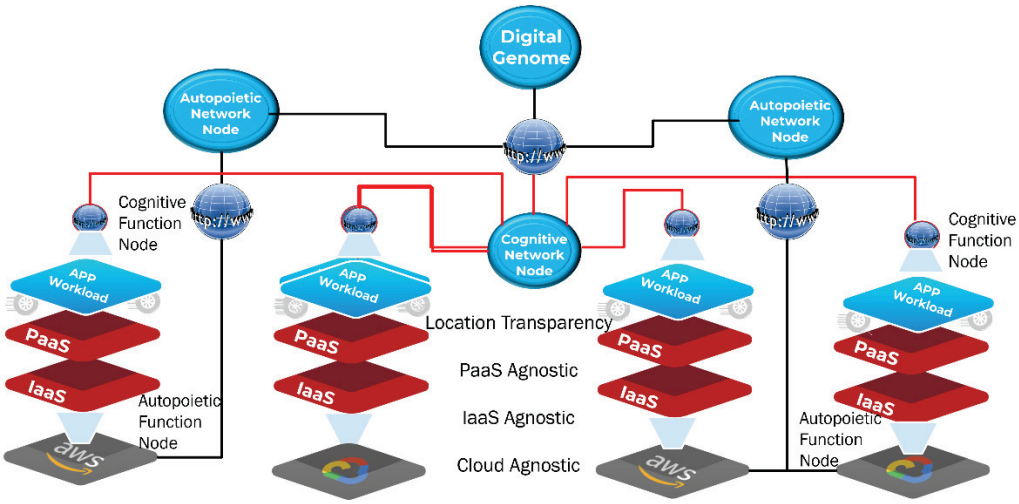



**Figure 4.** Structural machine and knowledge structures infuse autopoietic and cognitive behaviors into application deployment and operation to manage fluctuations. Cloud services IaaS and PaaS are used by the autopoietic and cognitive nodes.

The digital genome (similar to a genome in a biological system, which contains the "life" processes and executes the autopoietic and cognitive behaviors using genes and neural networks), contains the "life" processes of the application, and using the knowledge of the available resources in the form of infware, uses the knowledge network to execute various functions. All knowledge nodes are wired together to grant autopoietic and cognitive behaviors.

## 4. Conclusions

We have proposed an approach using GTI tools to infuse autopoietic and cognitive behaviors for the design and implementation of a self-managing application with the ability to integrate various components and their behaviors to act as a single unit with self-monitoring and self-management capabilities, to maintain stability in the face of fluctuations in a heterogeneous multi-cloud environment. Benefits of this approach include going beyond current techniques, providing an alternative with higher efficiency and addressing both single vendor lock-in and the complexity and tool fatigue of third party-based application orchestration across multiple cloud infrastructures. While application of GTI is in its infancy, there are currently several efforts underway in various laboratories and startups to apply the GTI in various domains.

**Author Contributions:** Both authors contributed to this paper using previous work by M. Burgin. All authors have read and agreed to the published version of the manuscript.

**Funding:** There is no external support for this research.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
2. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: Singapore, 2010.
3. Burgin, M. Information Processing by Structural Machines. In *Theoretical Information Studies: Information in the World*; World Scientific: New York, NY, USA; London, UK; Singapore, 2020; pp. 323–371.
4. Burgin, M. Triadic Automata and Machines as Information Transformers. *Information* **2020**, *11*, 102. [CrossRef]
5. Burgin, M.; Mikkilineni, R.; Phalke, V. Autopoietic Computing Systems and Triadic Automata: The Theory and Practice. *Adv. Comput. Commun.* **2020**, *1*, 16–35. [CrossRef]
6. Burgin, M.; Mikkilineni, R. From data Processing to Knowledge Processing: Working with Operational Schemas by Autopoietic Machines. *Big Data Cogn. Comput.* **2021**, *5*, 13. [CrossRef]
7. Mikkilineni, R. Information Processing, Information Networking, Cognitive Apparatuses and Sentient Software Systems. *Proceedings* **2020**, *47*, 27. [CrossRef]

proceedings

MDPI

Proceeding Paper

# Information Processing by Selective Machines †

**Mark Burgin * and Karthik Rajagopalan**

Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA; karthikrajagopalan.223@gmail.com

* Correspondence: markburg@cs.ucla.edu

† Presented at the 13th International Workshop on Natural Computing (IWNC), IS4SI Summit 2021, online, 12–19 September 2021.

**Abstract:** The goal of this paper is to develop the novel automaton model of learning processes called a selective machine and to study the properties of these machines. The model is based on the analysis of the process of language acquisition by people, although it correctly reflects how learning occurs in nature when animals, birds and even fish learn. A selective machine is an abstract automaton that has processors that can belong to different classes of conventional abstract automata. This creates various classes of selective machines. It is proved that in a general case, a selective machine can have higher learning abilities than any of its processors. This shows how synergy emerges in the technological sphere and explains why computer networks are able to outperform separate computers.

**Keywords:** natural computing; information processing; correction; selection; processor; language; synergy

**Citation:** Burgin, M.; Rajagopalan, K. Information Processing by Selective Machines. *Proceedings* **2022**, *81*, 122. https://doi.org/10.3390/proceedings2022081122

Academic Editors: Marcin J. Schroeder, Masami Hagiya, Yasuhiro Suzuki and Gordana Dodig-Crnkovic

Published: 19 April 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Learning is an important category of information acquisition. Machine learning utilizes automata for learning in general and language learning, in particular. In addition, abstract automata are used for modeling learning by people. In this work, analyzing how people learn natural languages, we develop a new approach to modeling and performing language learning by abstract automata. This allows treating natural language learning as natural computing.

The conventional models of natural language acquisition assume that in the process of learning, children, as well as adult learners, find out and memorize the correct words, rules of generating sentences, and rules of their utilization. However, this picture misses an important peculiarity of the learning process. Namely, people also gain knowledge of incorrect words and sentences, and this knowledge helps them avoid incorrect linguistic constructions in communication.

To model this process, we introduce a new type of computational automata called selective machines. A selective machine can not only generate (compute) words and texts but also eliminate (uncompute) words and texts. This property allows achieving higher power and lower complexity in computations.

It is necessary to remark that this approach to learning has been previously studied in the context of formal grammars [1–6]. Here, we explore learning as a natural information acquisition process, modeling it with computing automata.

## 2. Constructing Selective Machines

A *selective machine* M has positive and negative processors which accept/recognize words.

The difference between positive and negative processors is their purpose of computation. Positive processors accept or recognize tentative (or possible) elements of a language. However, it is not assumed that all of them are correct (belong to the language under construction). The goal of negative processors is to recognize those elements that do not

belong to the language under construction, that is, are incorrect. This allows building a language by the procedure where at first tentative (or possible) elements of the language are extracted, and then the incorrect words are eliminated.

We note that a language $L$ is accepted or recognized by a conventional automaton (machine) $M$, such as a finite automaton or a Turing machine, if this automaton accepts all words from $L$ and only these words. It is denoted by $L_M$ or $L(M)$ and is also called that language of the machine $M$.

In the case of selective machines, we have two types of languages.

The *positive language* $L(M_P)$ of the selective machines $M$ is the language accepted/recognized by all positive processors of $M$.

The negative language is defined in a similar way.

The *negative language* $L(M_N)$ of the selective machines $M$ is the language rejected/eliminated/prohibited by any of the negative processors of $M$.

Positive and negative languages together recognize the language of selective machines in the following way.

The language $L(M) = L(M_P) \backslash L(M_N)$ is the language of the selective machine $M$.

## 3. Classes of Selective Machines

Taking two classes **K** and **H** of automata (algorithms), we denote by **K/H** the class of all selective machines, in which the positive processors are automata from **K** and the negative processors are automata from **H**.

In what follows, we consider selective machines that have one positive processor and one negative processor, each of which belongs to one of the following classes:

- **FA** is the class all finite automata working with words in a given alphabet
- **PA** is the class all pushdown automata working with words in a given alphabet
- **TM** is the class all Turing machines working with words in a given alphabet

This gives us the following classes of selective machines:

1. **FA/FA** is the class of all *finite selective machines*, or FF-selective machines, in which both positive and negative processors are finite automata.
2. **FA/PA** is the class of all FP-selective machines, in which the positive processor is a finite automaton, and the negative processor is a pushdown automaton.
3. **FA/TM** is the class of all FT-selective machines, in which the positive processor is a finite automaton, and the negative processor is a Turing machine.
4. **PA/PA** is the class of all pushdown selective machines, in which both the positive and negative processors are pushdown automata.
5. **PA/FA** is the class of all PF-selective machines, in which the positive processor is a pushdown automaton, and the negative processor is a finite automaton.
6. **PA/TM** is the class of all PT-selective machines, in which the positive processor is a pushdown automaton, and the negative processor is a Turing machine.
7. **TM/TM** is the class of all *Turing selective machines*, or TM-selective machines, in which both positive and negative processors are Turing machines.
8. **TM/FA** is the class of all PF-selective machines, in which the positive processor is a Turing machine, and the negative processor is a finite automaton.
9. **TM/PA** is the class of all TP-selective machines, in which the positive processor is a Turing machine, and the negative processor is a pushdown automaton.

## 4. Properties of Selective Machines

We note that the recognizing linguistic power RL($A$) (RL(**Q**)) of an automaton $A$ (a class **Q** of automata) is the class of all formal languages recognized by the automaton $A$ (by the automata from the class **Q**).

Taking two classes **K** and **H** of automata (algorithms), we can compare their recognizing power. The recognizing power of **K** is larger than or equal to the recognizing power of **H** if $L(\mathbf{H}) \subseteq L(\mathbf{K})$. It is denoted by $\mathbf{H} \leq_L \mathbf{K}$. The recognizing power of **K** is larger than the recognizing power of **H** if $L(\mathbf{H}) \subset L(\mathbf{K})$. It is denoted by $\mathbf{H} <_L \mathbf{K}$.

**Lemma 1.** *(a) If* $\boldsymbol{D} \leq_L \boldsymbol{K}$*, then* $\boldsymbol{D/H} \leq_L \boldsymbol{K/H}$*.*
*(b) If* $\boldsymbol{D} \leq_L \boldsymbol{H}$*, then* $\boldsymbol{K/D} \leq_L \boldsymbol{K/H}$*.*

**Theorem 1.** *The selective (recognizing) power of the class* ***FA/FA*** *of finite selective machines is the same as the selective (recognizing) power of the class* ***FA*** *of all finite automata.*

**Proof.** To achieve better understanding of selective machines, we give two proofs of this theorem—one direct and another by reduction to formal grammars. □

(1) *Direct proof.* Let us consider the empty class $\wedge_{\mathrm{FA}}$ of finite automata, i.e., By Lemma 1, **FA** $\leq_L$ **FA/FA** because **FA** = **FA**$/\wedge_{\mathrm{FA}}$ and we need only to prove **FA/FA** $\leq_L$ **FA**. Finite automata recognize regular languages. The difference of two regular languages is a regular language, which is accepted by a finite automaton. Consequently, any language from the class $L_A$(**FA/FA**) belongs to the class $L_A$(**FA**), and thus, **FA/FA** $\leq_L$ **FA,** which implies the equality **FA/FA** $=_L$ **FA**.
(2) *Indirect proof.* Finite automata are linguistically equivalent to regular grammars, i.e., **FA** $=_L$ $\boldsymbol{G}_3$ where $\boldsymbol{G}_3$ is the class of all regular grammars. Thus, the class **FA/FA** of finite selective machines is linguistically equivalent to the class $\boldsymbol{G}_{33}$ of grammars with prohibition [1]. Then by Theorem 2 from [2], we have **FA/FA** $=_L$ **FA**.

Theorem is proved.

**Theorem 2.** *The selective (recognizing) power of the class* ***PA/FA*** *of selective machines is the same as the selective (recognizing) power of the class* ***PA*** *of all pushdown automata.*

*Proof* is similar to the proof of Theorem 1.

**Theorem 3.** *The selective (recognizing) power of the class* ***TM/FA*** *of selective machines is the same as the selective (recognizing) power of the class* ***TM*** *of all Turing machines.*

*Proof* is similar to the proof of Theorem 1.

**Theorem 4.** *The selective (recognizing) power of the class* ***TM/TM*** *of Turing selective machines is higher than the selective (recognizing) power of the class* ***TM*** *of all Turing machines.*

**Proof.** Turing machines are linguistically equivalent to unrestricted grammars from the Chomsky hierarchy, i.e., **TM** $=_L$ $\boldsymbol{G}_0$ where $\boldsymbol{G}_0$ is the class of all unrestricted grammars. Thus, the class **TM/TM** of finite selective machines is linguistically equivalent to the class $\boldsymbol{G}_{00}$ of grammars with prohibition [1]. Then by Theorem 7 from [2], we have **TM/TM** $>_L$ **TM**. □

Theorem is proved.

The statement of Theorem 4 means that selective machines with Turing machines as their processors can do more than Turing machines. Consequently, **TM/TM** is the class of super-recursive algorithms [7]. This, in turn, refutes the Church-Turing Thesis.

Theorems 3 and 4 display relations of selective machines to the arithmetical hierarchy [8,9].

The language (set) $L$ is *Turing selective recognizable* if it is recognized by some machine from the class **TM/TM**.

**Theorem 5.** *The class* ***STR*** *of all Turing selective recognizable languages (sets) contains the union* $\Sigma_1 \cup \Pi_1$.

This result brings us to the problem whether **STR** = $\Sigma_1 \cup \Pi_1$. One more interesting problem is to study selective machines with oracles.

**Theorem 6.** *The accepting (recognizing) complexity in the class **TM/TM** of selective machines can be much smaller than the accepting (recognizing) complexity in the class **TM** of all Turing machines.*

This shows that selective machines can be not only more powerful than Turing machines but also more efficient.

## 5. Conclusions

We described a novel model of abstract automata called a selective machine, demonstrating that in some cases selective machines have the same recognizing (learning) power as their constituents (processors), while in other important cases, such as selective machines with Turing machines as their constituents (processors), they have higher recognizing (learning) power than their constituents (processors).

In addition, selective machines formalize techniques used in proving some important results for Turing machines. An example of such a result is the Friedberg–Muchnik theorem [8,9].

It is possible to ask whether the same automaton can generate words and exclude those that do not belong to the language under construction. The answer is yes, it is possible, but the results proved by the authors demonstrate that in many important cases, two automata—one positive and another negative, which belong to the same class K (for example, both are Turing machines)—can generate, describe, and recognize many more languages than one automaton from this class K (one Turing machine) can. This clearly shows how synergy emerges from the interaction of constituents in a system.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. Grammars with Prohibition and Human-Computer Interaction. In Proceedings of the Business and Industry Simulation Symposium; Society for Modeling and Simulation International: San Diego, CA, USA, 2005; pp. 143–147.
2. Burgin, M. Basic Classes of Grammars with Prohibition, Preprint in Computer Science. *arXiv* **2013**, arXiv:1302.5181.
3. Burgin, M. Grammars with Exclusion. *J. Comput. Technol. Appl.* **2015**, *6*, 56–66.
4. Carlucci, L.; Case, J.; Jain, S. Learning correction grammars. *J. Symb. Logic* **2009**, *74*, 489–516. [CrossRef]
5. Case, J.; Jain, S. Rice and Rice-Shapiro theorems for transfinite correction grammars. *Math. Logic Q.* **2011**, *57*, 504–516. [CrossRef]
6. Case, J.; Royer, J. Program Size Complexity of Correction Grammars in the Ershov Hierarchy. In *12th Conference of Computability in Europe (CiE 2016), Proceedings, Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2016; Volume 7921, pp. 240–250.
7. Burgin, M. *Super-Recursive Algorithms*; Springer: New York, NY, USA; Berlin/Heidelberg, Germany, 2005.
8. Shen, A.; Vereshchagin, N.K. *Computable Functions*; AMS: Providence, RI, USA, 2003.
9. Rogers, H. *Theory of Recursive Functions and Effective Computability*; MIT Press: Cambridge, MA, USA, 1987.

proceedings

MDPI

*Proceedings*

# Problem-Oriented Foundations of Intelligence in the Context of Superintelligence †

**Mark Burgin**

Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

† Conference Theoretical Information Studies (TIS), Berkeley, CA, USA, 2–6 June 2019.

Published: 13 May 2020

**Abstract:** Despite the substantial interest in intelligence, both natural and artificial, and active research in this area, there is still considerable disagreement about what exactly constitutes intelligence. The problem-oriented approach to intelligence is based on evaluation of intelligence of a system by the level of problems this system is able to solve. The goal of this work is to develop a problem-oriented theory of intelligence in such a way that ecological characteristics become parameters of the problem-oriented model of intelligence. This model is constructed using mathematical theories and algorithmic reasoning. The suggested approach allows the reorganizing of the existing typologies of intelligence developed in works of different psychologists, providing better theoretical tools for intelligence measurement and evaluation.

**Keywords:** intelligence; superintelligence; information; problem; artificial intelligence; task; solution; measure

## 1. Introduction

Recently, the concept of superintelligence has been created, and become very popular [1,2]. However, to understand superintelligence, it is necessary at first to understand intelligence. Understanding of some systems or phenomena in science and often in mundane life comes through creation of relevant and adequate models of these systems and phenomena. For a long time, researchers ascribed intelligence only to people, and studied it in psychology, where it has become one of the most talked about subjects. In their studies of people in general, and their mental traits and behavior in particular, researchers elaborated various models of intelligence, discerning different types and kinds of intelligence of human beings and constructing measures of intelligence: analytical intelligence and analytical IQ (cf., for example, [3,4]), creative intelligence and creative IQ (cf., [5,6]), social intelligence [7], practical intelligence and practical IQ [5], and emotional intelligence and emotional IQ [8,9].

With the development of information technology, it became a focal goal to create artificial intelligence in the form of thinking machines or automata. This endeavor went into three directions: experimental studies of intelligence of people, theoretical studies of intelligence as a natural phenomenon, and the creation of more and more powerful information processing systems, such as computers, networks and other technical devices.

However, despite the substantial interest in the subject and active research, there is still considerable disagreement about what exactly constitutes intelligence. Futher, there is no standard definition of what exactly constitutes intelligence. For instance, Sternberg defines intelligence as "mental activity directed toward purposive adaptation to, selection, and shaping of real-world environments relevant to one's life." The goal of this work is to develop a problem-oriented theory of intelligence in such a way that ecological characteristics become parameters of the solvability

model of intelligence. This model is rooted in mathematics and algorithmic reasoning. Our approach allows the reorganization of the existing typologies of intelligence developed in works of different psychologists, providing better theoretical tools for intelligence measurement and evaluation.

## 2. Classification of Intelligence Models

The general theory of information (GTI) is an innovative theoretical system with three components: the axiomatic foundation, mathematical core and functional hull [1].

The diversity of models of intelligence can be demarcated into three classes: attributive or trait-oriented, behavioral or task-oriented, and enigmatic or problem-oriented models of intelligence.

According to the attributive or trait-oriented approach, a person is intelligent if this person has definite mental or personality traits. For instance, intelligence must involve such mental abilities as logical reasoning, problem-solving and planning.

According to the problem-oriented (enigmatic) approach, a person is intelligent if this person can solve definite problems. For instance, to be intelligent, people must then be able to take what they have learned to come up with a useful solution to a problem they have noticed in the world around them.

According to the behavioral or task-oriented approach, a person is intelligent if this person can perform definite tasks.

There are natural relations between these approaches.

**Proposition 1.** *It is possible to transform any behavioral (task-oriented) model into an enigmatic (problem-oriented) model and vice versa.*

**Proposition 2.** *It is possible to regard any behavioral (task-oriented) model as an attributive (trait-oriented) model.*

Propositions 1 and 2 imply the following result:

**Corollary 1.** *It is possible to regard any enigmatic (problem-oriented) model as an attributive (trait-oriented) model.*

## 3. Problem-Oriented Model of Intelligence

According to the second ontological principle of the general theory of information, O2 (the General Transformation Principle), information plays the same role in the World of Structures as energy plays in the Physical (material) World [2]. Physicists studied energy much longer than researchers explored information. It makes possible learning what physicists discovered about energy and applying this knowledge to the study of information.

The history of humankind and studies of psychologists demonstrate that the concept of intelligence is a relative characteristic of different systems. How we interpret this concept depends on social and natural environment, and thus, it is ecologically reliant and contingent. This was the main obstacle in finding an encompassing, and at the same time exact, definition of intelligence.

An important property of intelligence is contextuality, which means that intelligence depends on the context in which it is considered. In particular, context includes environment, and this makes intelligence ecologically contingent. For instance, a person intelligent in jungles can be dumb in a big contemporary city and vice versa. One more important characteristic of intelligence is its graduality. It means that different people can have different levels of intelligence.

To build the solvability model of intelligence, we consider a set $\boldsymbol{P}$ of problems and two classes of systems **H** and **K**.

**Condition (A):** For any system $T$ in **K**, there is a system $A$ in **H**, which can produce all results (solve all problems) that $T$ can produce (solve).

**Condition (B):** Any system $A$ from **H** can solve all problems from the set $\boldsymbol{P}$.

**Condition (C):** For any problem *P* from the set ***P***, there is a system *A* in **H**, which can solve it.

**Lemma 1.** *Condition (C) follows from Condition (B).*

**Definition 1.** *(a) A class **H** is called complete for a class **K** if it satisfies Condition (A).*
*(b) A system A is called total for a class **K** if the class {A} satisfies Condition (A).*
*(c) A class **H** is called complete in a class **K** if it satisfies Condition (A) and **H** ⊆ **K**.*
*(d) A system A from a class **K** is called universal in **K** if the class {A} satisfies Condition* (A).

**Example 1.** *A class **H** of Turing machines is complete in the class **T** of all Turing machines if it contains at least one universal Turing machine.*

**Example 2.** *A universal Turing machine is universal in the class **T** of all Turing machines.*

**Example 3.** *A class **H** of inductive Turing machines is complete in the class **T** of all inductive Turing machines if it contains at least one universal inductive Turing machine.*

**Lemma 2.** *Any class of information processing systems **K** is complete in itself.*

**Lemma 3.** *Any universal in **K** system is total for **K**.*

**Proposition 3.** *If a class **H** is complete for (in) a class **K** and **H** ⊆ **G**, then the class **G** is complete for the class **K**.*

**Corollary 2.** *If a class **H** contains a total for **K** system, then the class **H** is complete for the class **K**.*

**Proposition 4.** *If a class **H** is complete for (in) a class **K** and **F** ⊆ **K**, then the class **H** is complete for the class **F**.*

**Corollary 3.** *If a system A is total for **K** and **F** ⊆ **K**, then the system A is total for **F**.*

**Corollary 4.** *If a class **H** contains a total for a class **K** system and **F** ⊆ **K**, then the class **H** is complete for the class **F**.*

Using the problem-oriented approach and taking a set of problems ***P***, which is called an intelligence parameter, we define intelligent subclasses and systems in a class **K** of systems.

**Definition 2.** *A class **H** of systems is called weakly intelligent with respect to **P** if it satisfies Condition (B).*

We denote this by **H** = $\mathrm{Int}_w^{P}$, and take the class **K*P***, that consists of all problems each of which can be solved by some system from a class **K** of systems, and systems from a class **H** can do necessary reductions of problems.

**Proposition 5.** *A class **H** is complete for a class **K** if and only if **H** = $Int_w^{KP}$.*

**Corollary 5.** *A class **H** is complete in a class **K** if and only if **H** ⊆ **K** and **H** = $Int_w^{KP}$.*

**Proposition 6.** *If **H** = $Int_w^{P}$ and **Q** ⊆ **P**, then **H** = $Int_w^{Q}$.*

The mental effort of an individual who learns some portion of knowledge can be measured by time of this learning. It is also possible to define mental effort by measuring intensity of the brain activity during learning.

If the system ***P*** is determined by the class **K**, we denote this by **H** = $\mathrm{Int}_w^{P}$ **K**.

**Definition 3.** *(a) A class **H** of systems is called robustly intelligent for a class **K** with respect to **P** if it satisfies Conditions (A) and (B). We denote this by **H** = $Int_r^{P}$ **K**.*

*(b) A class* $\boldsymbol{H}$ *of systems is called robustly intelligent in a class* $\boldsymbol{K}$ *with respect to* $\boldsymbol{P}$ *if* $\boldsymbol{H} \subseteq \boldsymbol{K}$ *and it satisfies Conditions (A) and (B). We denote this by* $\boldsymbol{H} = Int_{rin}^{P}\ \boldsymbol{K}$.

In other words, a robustly intelligent with respect to $\boldsymbol{P}$ class of problems is complete and weakly intelligent.

**Proposition 7.** *If a class* $\boldsymbol{H}$ *is robustly intelligent for (in) a class* $\boldsymbol{K}$*, i.e.,* $\boldsymbol{H} = Int_{r}^{P}\ \boldsymbol{K}$ *(*$\boldsymbol{H} = Int_{rin}^{P}\ \boldsymbol{K}$*), and* $\boldsymbol{F} \subseteq \boldsymbol{K}$*, then the class* $\boldsymbol{H}$ *is robustly intelligent for the class* $\boldsymbol{F}$*., i.e.,* $\boldsymbol{H} = Int_{r}^{P}\ \boldsymbol{F}$.

**Proposition 8.** *(a) If* $\boldsymbol{H} = Int_{r}^{P}\ \boldsymbol{K}$ *and* $\boldsymbol{Q} \subseteq \boldsymbol{P}$*, then* $\boldsymbol{H} = Int_{r}^{Q}\boldsymbol{K}$.

*(b) If* $\boldsymbol{H} = Int_{rin}^{P}\ \boldsymbol{K}$ *and* $\boldsymbol{Q} \subseteq \boldsymbol{P}$*, then* $\boldsymbol{H} = Int_{rin}^{Q}\boldsymbol{K}$.

## 4. Conclusions

The basic elements of a problem-oriented theory of intelligence are elaborated in the context of superintelligence in such a way that ecological characteristics become parameters of the problem-oriented model of intelligence. Mathematical theories and algorithmic reasoning form foundations of this theory. The suggested approach allows the reorganization of the existing typologies of intelligence developed in works of different psychologists. The next step in the development of the problem-oriented theory of intelligence presented here is to build better theoretical tools for intelligence measurement and evaluation.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Bostrom, N. *Superintelligence*: *Paths, Dangers, Strategies*; Oxford University Press: Oxford, UK, 2014.
2. Burgin, M. Swarm Superintelligence and Actor Systems. *Int. J. Swarm Intell. Evol. Comput.* **2017**, *6*, 3.
3. Spearman, C. *The Abilities of Man*: *Their Nature and Measurement*; Macmillan: New York, NY, USA, 1927.
4. Duncan, J.; Rüdiger, J.S.; Kolodny, J.; Bor, D.; Herzog, H.; Ahmed, A.; Newell, F.N.; Emslie, H. A neural basis for general intelligence. *Science* **2000**, *289*, 457–460.
5. Sternberg, R.J. *Beyond IQ*: *A Triarchic Theory of Intelligence*; Cambridge University Press: Cambridge, UK, 1985.
6. Burgin, M. *Intellectual Components of Creativity*; International Academy "Man in Aerospace Systems": Kiev, Ukraine, 1998. (In Ukrainian)
7. Thorndike, R.K. Intelligence and Its Uses. *Harpers Mag.* **1920**, *140*, 227–335.
8. Mayer, J.D.; Salovey, P. The intelligence of emotional intelligence. *Intelligence* **1993**, *17*, 433–442.
9. Koonce, R. Emotional IQ, a new secret of success? *Train. Dev.* **1996**, *50*, 19–21.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

Proceeding Paper

# Three Approaches to Artificial Intelligence †

**Mark Burgin**

Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA; markburg@cs.ucla.edu

† Presented at the 3rd Global Forum on Artificial Intelligence (GFAI), IS4SI Summit 2021, online, 12–19 September 2021.

**Abstract:** The aim of this paper is the expansion of understanding what intelligence is, what is being performed in the area of AI and in which direction to move further in this area. To achieve these goals, the stratified componential model of intelligence in general and AI, in particular, is introduced and studied. Its application entails three approaches to the development of artificial intelligence.

**Keywords:** intelligence; actor; agent; oracle; environment; problem solver; assimilation; adaptation; accommodation

**Citation:** Burgin, M. Three Approaches to Artificial Intelligence. *Proceedings* **2022**, *81*, 1147. https://doi.org/10.3390/proceedings2022081147

Academic Editors: Yixin Zhong and Kun Wu

Published: 25 May 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Currently there is no generally accepted definition of intelligence. Furthermore, there is no consensus on what intelligence is. Researchers have elaborated a variety of models of intelligence and constructed many measures of intelligence such as analytical intelligence and analytical IQ, creative intelligence and creative IQ, social intelligence, practical intelligence and practical IQ, and emotional intelligence and emotional IQ. In this paper, we further develop our understanding of what intelligence is, what is being pursued in the area of AI and the directions in which to develop in this area. To achieve these goals, the stratified componential model of intelligence in general and AI, in particular, is introduced and studied. The application of this model provides three approaches to the development of artificial intelligence.

## 2. Levels of Intelligence

It is possible to distinguish three ways of defining intelligent systems.

The first approach implies that a system is intelligent when it can solve some complex problem or carry out some complex activity in a simple environment [1]. It is called ***local intelligence***.

Examples of intelligent problems are games such as chess or GO as well as image recognition. Examples of an intelligent activity are conversation, as in the Turing test [2], or a robot that performs an intelligent task.

A system of local intelligence can be simply called a problem solver.

It is the *unit-component approach* to intelligence. One problem is a unit component of intelligence.

The second approach considers an intelligent system as a system that can well function in a definite domain, which demands solving a group of (complex) problems. It is called (***advanced***) ***cluster intelligence***.

It is the bona fide *componential approach* to intelligence.

Cluster intelligence is used in the Triarchic Theory of Intelligence developed by Robert Sternberg [3,4].

A self-organizing system that can read a textbook and answer questions displays cluster intelligence [5,6].

It is necessary to understand that cluster intelligence is not always higher than local intelligence. For instance, by contemporary measures, cluster intelligence that can solve a

group of problems by performing arithmetical operations is lower than local intelligence that can play chess on the level of a world champion.

The third approach assumes that a system is intelligent when it can efficiently function in complex conditions or successfully survive in a hostile environment. It is called *global intelligence*.

Although it is important to construct artificial systems that solve complex problems or/and display intelligent activity, the imperative goal of AI research is to create systems that can achieve global intelligence.

In each of the considered groups, we can separate different levels of intelligence. For instance, to be able to perform arithmetical operations without technical devices is also a kind of human intelligence although it is considered to be a very low intelligence by the criteria of contemporary society.

## 3. Intelligence in Science and Mathematics

The suggested stratification of intelligence has interesting connections to the types of scientists and mathematicians, which are presented in the following classification.

Mission-oriented types of scientists and mathematicians include:

1. World explorer;
2. Theory builder;
3. Problem solver.

The lowest level in this classification, a problem solver, corresponds to the lowest level of intelligence—local intelligence. Indeed, as we mentioned before, a system of local intelligence can simply be called a problem solver.

The middle level in this classification, a theory builder, corresponds to the middle level of intelligence—advanced cluster intelligence. Indeed, a theory, as a rule, cansolve-many problems.

The highest level in this classification, a world explorer, corresponds to the highest level of intelligence—global intelligence. Indeed, in science and mathematics, a world explorer studies nature or the ideal world of mathematical structures [7].

## 4. Intelligence Stratification

According to the theory of intellectual activity [8,9] and the Existential Triad of the world, intelligence is stratified into three principal forms:

- *Mental intelligence* mirrors the level of thinking;
- *Structural* (in particular, *linguistic*) *intelligence* reveals the quality of expression;
- *Action intelligence* exposes the reflected in behavior.

Note that mental intelligence can be essentially different from action intelligence. First, there are many examples of people who have very advanced intelligent thinking abilities but their behavior is far from optimal. Second, mentality with its mental intelligence is essentially different from the physical world with its action intelligence. People often create a specific world in their mentality and even their mental models of social and natural environment can poorly reflect the physical world.

It is important to understand that these forms of intelligence often interact and one system can possess two or all of the three forms of intelligence. For instance, action intelligence is often based on mental intelligence.

In addition, by its organization, intelligence can be of three pure types:

- Cognitive intelligence;
- Emotional intelligence;
- Instructional intelligence.

This classification is based on the Triune Model of the brain.

These pure types can interact and form the combined types of intelligence, for example, cognitive–emotional intelligence.

## 5. Terminological Aspect of AI

It is also important to establish a consistent terminology in the area of AI. A terminology is a system of terms (names) used in a definite area, e.g., in science, art, business or industry. Importance of the usage of correct names was stressed by the famous Chinese philosopher Confucius, who wrote:

> "*If names be not correct, language is not in accordance with the truth of things. If language be not in accordance with the truth of things, affairs cannot be carried on to success. When affairs cannot be carried on to success, proprieties and music do not flourish. When proprieties and music do not flourish, punishments will not be properly awarded. When punishments are not properly awarded, the people do not know how to move hand or foot*" [10].

Observing the terminology of AI, we can see that the most popular term in contemporary theoretical and philosophical studies of AI is that of an agent. However, oftentimesthat whichis called an agent, e.g., an intelligent agent, is actually an actor. The difference between actors and agents is described by the following definitions [11].

Taking a system C that consists of interacting systems $\{R_k; k \in K\}$, which have the lower rank than $C$, the systems $R_k$ are called **actors** and treated as actors.

An **agent** is an actor that (who) acts on behalf of another system performing some tasks of that system.

This shows that an agent performs the role of an actor, who (which) can also have other roles.

An **operator** also performs the role of an actor, that is, an operator is an actor that (who) transforms the environment to achieve the necessary or desired goals.

## 6. Features of Global Intelligence

For global action intelligence, the environment consists of living systems, communities and societies of living and artificial systems, artificial devices, and unanimated natural and artificial objects.

For global mental intelligence, the environment leads to the creation of the mentality and can include models of physical reality as well as mental systems deliberately generated by the mind. Note that it is possible to consider mental intelligence on different levels, which comprise individual mentality, group mentality, and social mentality [6,7].

For global structural intelligence, the environment consists of a variety ofknowledge and data structures such as mathematical or linguistic systems.

We see that on any level and especially on the global level, intelligence is expressed in terms of functioning, e.g., problem solving, in a particular environment. As the result, the exploration and development of intelligence in general and artificial intelligence in particular must take into account their environment und thus, must explore the context of ecology. The foundations of this approach are developed in the works of Yixin Zhong and Mark Burgin (see, for example, [12–15]). Projecting this situation on the realm of rituals, we see that by the investigating rituals, it is useful to make a distinction between a ritual, performance of a ritual, and description of a ritual.

## 7. Forms of Global Intelligence

The functioning of a system on the level of global intelligence can have three forms:

- *Assimilation*;
- *Adaptation*;
- *Accommodation*.

By defining these forms, we reflect the functioning of a system in some environment by the diagram, in which the arrow moves from the essence that is changing to the essence that determines changes and the size of the arrow shows the extent of changes.

*Adaption* of a system $A$ to the environment means changing $A$ to better fit the environment.

$$A \longrightarrow E$$

*Assimilation* into the environment means becoming a part of the environment.

$$A \Longrightarrow E$$

*Accommodation* of the environment by a system *A* means changing the environment to better fit the system *A*.

$$A \longleftarrow E$$

The invention of technical devices is an example of adaptation. Many games, such as chess or GO, provide examples of accommodation.

Assimilation, adaptation and accommodation can be inner, outer and combined, that is, both inner and outer.

*Inner adaptation* is a changing of the inner world, e.g., values, goals, habits, attitudes, the worldview, etc., of a person to fit the environment.

*Outer adaptation* is a changing of the behavior, relations and if possible, an individual on the physical level to fit the environment.

*Inner assimilation* is making your inner world (mentality) similar to the mentality of (the members of) the social environment.

*Outer assimilation* is adapting behavior, relations and if possible, themselves, to the physical level similar to the members of the social environment.

*Inner accommodation* is changing the inner model of the environment to better fit to the environment.

*Outer accommodation* is changing the environment to better fit to them.

With respect to its environment, a system can be:

- Non-interfering actor;
- Experimenting actor;
- Operator.

In this context, it is possible to set apart three approaches in the creation of globally intelligent systems.

In the *non-interfering actor approach*, an intelligent system is treated as an actor that only observes and adapts to the environment but does not impact it.

In the *experimenting actor approach*, an intelligent system is treated as an actor that only observes and adapts to the environment by performing experiments.

In the *operator approach*, an intelligent system is treated as an operator inits environment.

The difference is that an actor simply functions in the environment while an operator transforms the environment to achieve the necessary or desired goals.

## 8. Conclusions

Different approaches to understanding intelligence in general and to building artificial intelligence in particular are explicated and analyzed with the goal of better decision making and a selection of ways to develop artificial intelligence. Now, this process is based on the invention of better tools for solving more and more complex problems. However, a real breakthrough in this area is only possible by building a general theory of intelligence.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References


1. Burgin, M. Problem-oriented foundations of intelligence in the context of superintelligence. *Proceedings* **2020**, *47*, 21. [CrossRef]
2. Turing, A. Computing machinery and intelligence. *Mind* **1950**, *236*, 433–460. [CrossRef]
3. Sternberg, R.J. *Beyond IQ: A Triarchic Theory of Human Intelligence*; Cambridge University Press: Cambridge, UK, 1985.
4. Sternberg, R.J. *Successful Intelligence*; Plume: New York, NY, USA, 1997.
5. Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; Steinhardt, J. Measuring Massive Multitask Language Understanding. *arXiv* **2020**, arXiv:2009.03300.
6. Littman, M.L.; Ajunwa, I.; Berger, G.; Boutilier, C.; Currie, M.; Doshi-Velez, F.; Hadfield, G.; Horowitz, M.C.; Isbell, C.; Kitano, H. *Gathering Strength, Gathering Storms: The One Hundred Year Study on Artificial Intelligence (AI100) 2021 Study Panel Report*; Stanford University: Stanford, CA, USA, September 2021. Available online: http://ai100.stanford.edu/2021-report (accessed on 16 September 2021).
7. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
8. Burgin, M.S. *Intellectual Activity in Creative Work, in Forms of Knowledge Representation and Creative Thinking*; NSU: Novosibirsk, Russia, 1989; pp. 53–56. (In Russian)
9. Burgin, M. *Intellectual Components of Creativity*; AACS: Kiyiv, Ukraine, 1998. (In Ukrainian)
10. Confucius. *The Analects*; Harmondsworth: New York, NY, USA, 1979.
11. Burgin, M. Actors, Agents and Oracles in the Context of Artificial Intelligence. *J. Artif. Intell. Res. Adv.* **2017**, *4*, 17–25.
12. Zhong, Y.X. *Principles of Information Science*; BUPT Press: Beijing, China, 1988. (In Chinese)
13. Zhong, Y.X. Information Ecology. *Proceedings* **2017**, *1*, 200.
14. Burgin, M. Principles of General Ecology. *Proceedings* **2017**, *1*, 148. [CrossRef]
15. Burgin, M.; Zhong, Y. Information Ecology in the Context of General Ecology. *Information* **2018**, *9*, 57. [CrossRef]

proceedings

MDPI

Proceeding Paper

# Relational Structure of Conceptual Spaces †

**Mark Burgin [1,*] and José María Díaz-Nafría [2]**

[1] Department of Computer Science, University of California, Los Angeles, CA 90095, USA
[2] Department of Telecommunication Engineering, Madrid Open University, 28001 Madrid, Spain; josemaria.diaz.n@udima.es
* Correspondence: markburg@cs.ucla.edu
† Presented at the Conference on Theoretical and Foundational Problems in Information Studies, IS4SI Summit 2021, Online, 12–19 September 2021.

**Abstract:** Concepts play an important role in human culture, forming explicit and implicit conceptual spaces. The goal of this paper is to explore the structural organization of information in conceptual spaces.

**Keywords:** concept; information; conceptual space; relation; operation

**Citation:** Burgin, M.; Díaz-Nafría, J.M. Relational Structure of Conceptual Spaces. *Proceedings* **2022**, *81*, 39. https://doi.org/10.3390/proceedings2022081039

Academic Editor: Marcin Schroeder

Published: 19 September 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Concepts play an important role in human culture as the basic level of our knowledge systems [1]. In essence, all meaningful words are names of concepts. Consequently, dictionaries are collections of concept definitions, while encyclopedias are collections of extended concept descriptions. Thus, it is important to study concepts and their arrangements in the context of knowledge systems.

Concepts are intrinsically related to signs. According to the general understanding, a sign is an object that points to something different than itself, being transcendent to it. At the same time, a concept, or notion, is usually defined as a general idea derived from specific instances, i.e., a concept is a symbolic (usually, linguistic) representation of these instances, and this representation naturally points to these instances. Consequently, a concept is a specific sign.

In terms of their organization, concepts form conceptual spaces, which include relations between and operations with concepts as elements of these spaces [2–4]. The goal of this work is to study relations in conceptual spaces.

It is important to understand that the conceptual spaces introduced and explored in [4] are aimed at the study and further development of the structural organization of encyclopedias and encyclopedic dictionaries, while conceptual spaces introduced and studied by other authors are oriented at modeling mental knowledge structures in the mind. The conceptual spaces of the first type are called structural, while the conceptual spaces of the secondtype are called attributive. Here, we continue our studies of structural conceptual spaces with emphasis on the relations of these spaces.

## 2. Concepts and Their Models

In conceptual spaces, concepts are represented by their models. In our study, we use the most advanced model, which is called the representational model of concept [5]. On its second level, it can be treated as a synthesis of the Russell's model of concept and Peirce's model of sign. In the representational model of concept, the name of the concept is connected to the concept representative, which consists of three components: denotat as the collection of all particular exemplifications or instantiations of this concept, meaning or connotation, and sense or intentionality. (cf. Figure 1). We will call this special case of the representational model by the name of the extended Peircean model of a concept.

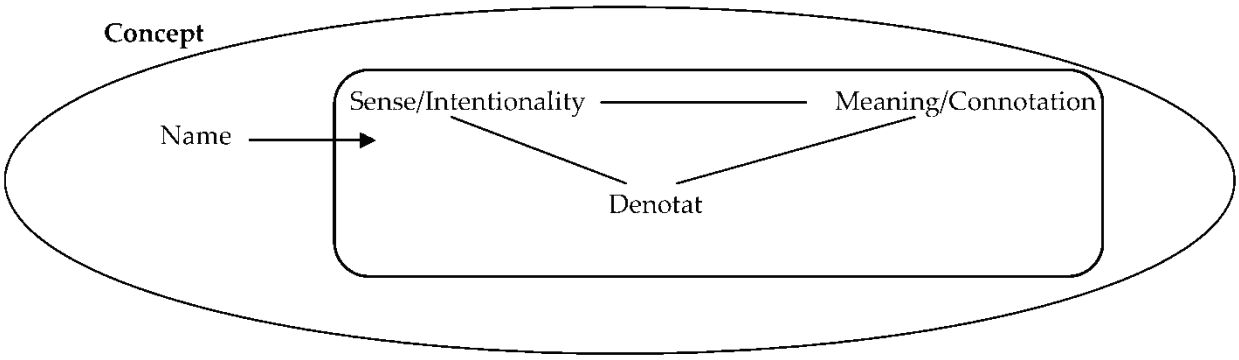



**Figure 1.** The extended Peircean model of a concept.

Meaning represents the semantics of the concept. *Sense* is associated with the pragmatics of the concept. Denotat is the generalized syntax of the concept. For a symbolic concept, its denotata consists of syntactic relations of this concept.

It is possible to treat other known models of concepts as specifications of the representational model of the concept [6,7].

Thus, the theory of concepts in general and the extended Peircean model of a concept, in particular, bring us to the notion of the generalized syntax of the concept, which consists not only of names (words) of objects that constitute the denotat, but also include the physical and mental objects from the denotat and relations between all kinds of these objects. While the conventional syntax is defined as the arrangement of words (names) in a sentence, the generalized syntax is defined as the arrangement of symbolic (e.g., words as names), mental and physical objects in the denotat of a concept.

Using properties of concepts, which form intermediate relations in conceptual spaces, it is possible to differentiate all concepts in a conceptual space into three groups—general, individual and impossible concepts—which are defined in the following ways:

- A general concept has many instantiations;
- An individual concept has only one instantiation;
- An impossible concept does not have instantiations.

Note that the membership of a concept in one of these groups is contextual, i.e., it depends on the context. For instance, the name of a person can be an individual concept in one group where there are no other people with this name, and a general concept in another group where there are several people with this name.

The concept cat is general.The concept of the Earth is individual. The concept "a ball larger than the Sun and smaller than the Earth" is impossible.

According to the type of the denotat, general concepts can be:

- Set concepts, in which the denotat is a set;
- Class concepts, in which the denotat is a class;
- Ensemble concepts, in which the denotat is an ensemble.

In contrast to general sets and classes, elements in an ensemble are related to one another.

Note that each classification or typology of a concept determines intermediate relations in the conceptual space.

According to the exactness of the denotat, general concepts can be:

- Strict concepts, in which the denotat is a set;
- Fuzzy concepts, in which the denotat is a fuzzy set;
- Blurry concepts, in which the membership in the denotat is not clearly defined.

## 3. Categorization of Conceptual Spaces

Existential stratification induces three basic types of conceptual spaces:

- Mental conceptual spaces consist of mental concepts and relations between them;

- Symbolic conceptual spaces consist of symbolic concepts and relations between them being physical representations of ideal conceptual structures;
- Substantial conceptual spaces include physical components of concepts in addition to symbolic conceptual representations.

Mental conceptual spaces are in the heads of people and are studied by psychologists and educators.

Encyclopedia is an example of a symbolic conceptual space.

If the concept tree belongs to asubstantial conceptual space, then this space also includes some number of trees.

Here, we study symbolic conceptual spaces with the goal to explicate the structural organization of information in conceptual spaces.

## 4. Relations in Conceptual Spaces

Studying conceptual spaces, we consider relations of two types: inner and intermediate relations [6]. On the first level of the space structure, inner relations of a conceptual space are relations between elements of this space (cf. Figure 2), while intermediate relations of a conceptual space are relations between elements of this space and some other objects (see Figure 3). Examples of intermediate relations are abstract properties of concepts. Examples of inner relations are relations"to be more (less) general" or "to be more (less) abstract."

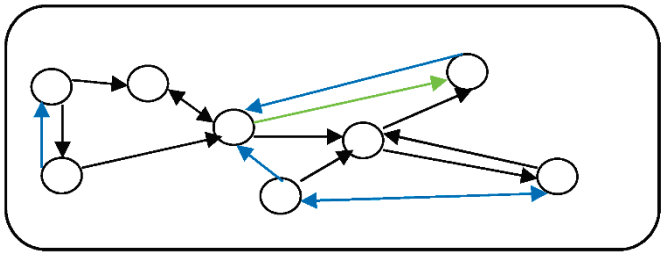


**Figure 2.** Conceptual space with inner relations between concepts.

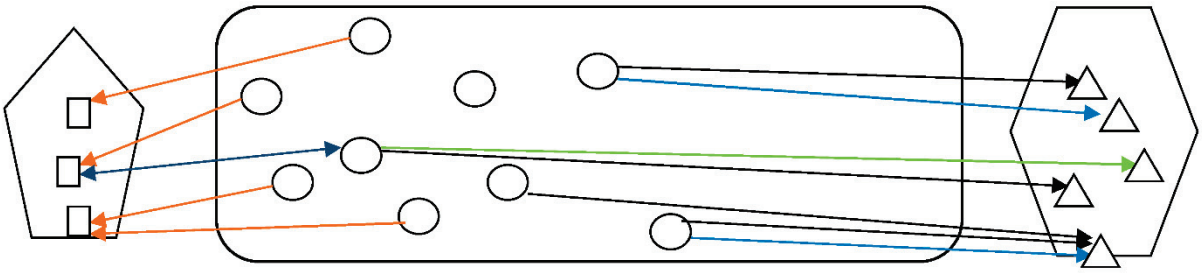


**Figure 3.** Conceptual space with intermediate relations of concepts.

Conceptual relations can be divided into three groups:

- Semantic relations explicate the meaning of the concept;
- Pragmatic relations display the intentionality of the concept;
- Syntactic relations form the (symbolic) denotat of the concept.

Let us consider some basic inner relations in conceptual spaces.

A concept $C$ is more abstract than a concept $D$ if the properties that define the concept $C$ form a part of the properties that define the concept $D$.

For instance, the concept tree is more abstract than the concept pine. The concept number is more abstract than the concept natural number. The concept feeling is more abstract than the concept love.

**Proposition 1.** *The relation "to be more abstract" is transitive.*

A concept $C$ is more general than a concept $D$ if the denotat of the concept $D$ is a subset of the denotat of the concept $C$.

For instance, the concept real number is more general than the concept whole number.

**Proposition 2.** *The relation "to be more general" is transitive.*

Often, but not always, more general concepts are also more abstract. For instance, the concept tree is more abstract and more general than the concept pine. At the same time, the concept real number is more general than the concept whole number, but it is not more abstract than the concept whole number. Both concepts are situated on the same level of abstraction. In a similar way, the concept feeling is more abstract than the concept love, but it is not more general than the concept love.

## 5. Conclusions

Explicating and exploring the relational structure of conceptual spaces, here we formally described and studied only the most popular relations between concepts. Derivation of a more complete picture of conceptual spaces is the goal of further studies.

Another important direction of the research in this area is the study of existing operators and the creation of new operators in structural conceptual spaces forming, in such a way, conceptual algebras.

**Author Contributions:** These authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. *Theory of Knowledge: Structures and Processes*; World Scientific: New York, NY, USA; London, UK; Singapore, 2016.
2. Doignon, J.-P.; Falmagne, J.-C.L. *Knowledge Spaces*; Springer: Berlin/Heidelberg, Germany, 1999.
3. Gärdenfors, P. Conceptual Spaces as a Framework for Knowledge Representation. *Mind Matter* **2004**, *2*, 9–27.
4. Burgin, M.; Díaz-Nafría, J.M. Introduction to the Mathematical Theory of Knowledge Conceptualization: Conceptual Systems and Structures. In Proceedings of the Second International Conference on Applied Informatics (ICAI 2019), Madrid, Spain, 7–9 November 2019; Springer: Cham, Switzerland, 2019; pp. 469–482.
5. Burgin, M.S.; Gorsky, D.P. *Towards the Construction of a General Theory of Concept*; The Opened Curtain: Oulder, UK; San Francisco, CA, USA; Oxford, UK, 1991; pp. 167–195.
6. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
7. Hampton, J.A. Concepts and Natural Language. In *Concepts and Fuzzy Logic*; MIT Press: Cambridge, MA, USA, 2011; pp. 233–258.

*big data and cognitive computing*

MDPI

*Article*

# Operations with Nested Named Sets as a Tool for Artificial Intelligence

**Mark Burgin**

Department of Computer Science, University of California, Los Angeles, CA 90095, USA; markburg@cs.ucla.edu

**Abstract:** Knowledge and data representations are important for artificial intelligence (AI), as well as for intelligence in general. Intelligent functioning presupposes efficient operation with knowledge and data representations in particular. At the same time, it has been demonstrated that named sets, which are also called fundamental triads, instantiate the most fundamental structure in general and for knowledge and data representations in particular. In this context, named sets allow for effective mathematical portrayal of the key phenomenon, called nesting. Nesting plays a weighty role in a variety of fields, such as mathematics and computer science. Computing tools of AI include nested levels of parentheses in arithmetical expressions; different types of recursion; nesting of several levels of subroutines; nesting in recursive calls; multilevel nesting in information hiding; a variety of nested data structures, such as records, objects, and classes; and nested blocks of imperative source code, such as nested repeat-until clauses, while clauses, if clauses, etc. In this paper, different operations with nested named sets are constructed and their properties obtained, reflecting different attributes of nesting. An AI system receives information in the form of data and knowledge and processing information, performs operations with these data and knowledge. Thus, such a system needs various operations for these processes. Operations constructed in this paper perform processing of data and knowledge in the form of nested named sets. Knowing properties of these operations can help to optimize the processing of data and knowledge in AI systems.

**Keywords:** artificial intelligence; knowledge; data; nesting; named set; operation; function; name; information; structure

**Citation:** Burgin, M. Operations with Nested Named Sets as a Tool for Artificial Intelligence. *Big Data Cogn. Comput.* **2022**, *6*, 37. https://doi.org/10.3390/bdcc6020037

Academic Editor: Domenico Talia

Received: 30 December 2021
Accepted: 9 March 2022
Published: 1 April 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Knowledge and data representations are important for artificial intelligence (AI), as well as for intelligence in general. With respect to knowledge and data representation, it has been demonstrated that named sets, which are also called fundamental triads, instantiate the most fundamental structure [1].

On the one hand, it has been proved that basic mathematical structures, such as functions, relations, morphisms in categories, functors, operators, graphs, multigraphs, and sets, are named sets. As a result, the theory of named sets forms the unified foundation for mathematics, comprising set theory, logic, category theory, algorithms, univalent foundations, and intuitionism [2].

On the other hand, named sets have demonstrated their efficacy in a diversity of computer and network applications. For instance, as demonstrated in [3], natural language processing applications and downstream tasks often employ named entity recognition (NER). At the same time, contemporary NER systems work with flat annotation from widespread datasets, avoiding the semantic information contained in nested entities. To exclude these deficiencies, a more powerful technique—nested named entity recognition (NNER)—was elaborated [4].

Named sets also form the structure of dynamics at different levels. In particular, it is found that events, actions, and processes have the structure of a named set, or what is the

same, a fundamental triad [5,6]. An efficient model of concurrent computations to express branching time is based on the special case of named sets called a Chu space [7,8].

Named sets play an important role in programming languages because utilization of identifiers must be completely namespace-qualified, whereas namespaces are special named sets. Programming languages with direct support for namespaces regularly provide tools for programmers to declare up front what identifiers from a specific namespace they are going to use, and then they can use them without references to the namespace for the remainder of the block [9]. Named sets were useful for constructing a mathematical model of professional organizations and processing XML structures [10].

Named sets form the most fundamental and useful data structure in databases and knowledge bases. Because an important special case of named sets is set-theoretical relations, all relational databases stockpile named sets as data structures and use them to provide information to their users [11,12]. It has also been demonstrated that in temporal databases, named set chains are basic structures [13,14]. Utilization of named sets for data visualization and information retrieval in databases was developed in [15–18], while their usage for database management was elaborated in [19,20].

Additionally, named sets are used as a unified data metamodel, which allows for adaptation of data models on all levels of data representations: from conceptual or high-level to implementation or representational to physical models on the low level. In turn, named sets and their chains serve as resourceful high-level metadata for a variety of applications (cf., [21]).

One of the highest achievements of modern technology is the Internet. At the same time, any network—especially the World Wide Web—extensively uses names and, consequently, name sets because names are related to what they name, which produces a named set [22]. For instance, the essential component of the Internet is the Domain Name System (DNS). It converts user-level domain names into IP addresses. This means that DNS builds in named sets for web operation. A more advanced approach to the Internet employs the Intentional Naming System (INS) instead of DNS [23]. This is a new naming system intended for naming and discovering a variety of resources in future networks of devices and services by building named sets. In INS, names describe applications and other resources rather than network locations, as is the case in DNS. This means that INS also builds in named sets for web operation. The Internet uses naming schemata and, in particular, intentional naming schemata, which provide dynamic construction of named sets used in the functioning of the Internet.

Although DNS has one level of name resolution in building a dynamic named set, researchers argue that there should be three levels of name resolution: from user-level descriptors to service identifiers; from service identifiers to endpoint identifiers; and from endpoint identifiers to IP addresses [24]. This implies the construction of a more complex mathematical system called a named set chain [1].

Moreover, utilization of names on the Internet has become even more important with anew network technology, Named Data Networking (NDN), a future internet architecture inspired by years of empirical research into network usage and a growing awareness of unsolved problems in contemporary internet architectures [25].

Studying and utilizing data and knowledge, researchers found that complex structures often include nested substructures. Informally, nesting means that one system is a subsystem of another system. Nesting exists in many areas as an essential technique in computer science and technology. Examples of nesting include: nested levels of parentheses in arithmetical expressions; different types of recursion; nesting of several levels of subroutines; nesting in recursive calls; multilevel nesting in information hiding; a variety of nested data structures, such as records, objects, and classes; and nested blocks of imperative source code, such as nested repeat-until clauses, while clauses, if clauses, etc. [3,4,26–29].

Taking into account the fundamental nature of named sets, it is natural to utilize named sets for building a mathematical model of nesting. The goal of this model is to develop a system of efficient operations with nested entities for various applications in

general intelligence and artificial intelligence, forming a base for algorithmic structures of artificial intelligence. With this in mind, the remainder of this paper is structured as follows.

In Section 2, basic definitions and constructions from named set theory [1] are presented. In Section 3, basic definitions and constructions from nested named set theory [30] are presented. In Section 4, we develop a theory of operations with nested named sets aimed at creating an operational foundation for the development of artificial intelligence. Section 5 contains conclusions and direction for further research.

## 2. General Named Sets: Basic Definitions and Constructions

We start with the definition of the basic structure for this work: a *named set* or a *fundamental triad*. Calling the same object by different names reflects dissimilar features of the object. The name *named set* directs our attention to the inner structure of the object. The name *fundamental triad* reveals the unity of the object. To achieve higher generality and flexibility, herein, we utilize relative named sets (fundamental triads).

Let us consider two arbitrary classes of objects $\mathfrak{V}$ and $\mathfrak{B}$ as well as a third class $\mathfrak{R}$ of relations, correspondences, ties or connections between objects from $\mathfrak{V}$ and objects from $\mathfrak{B}$. They form the triad ($\mathfrak{V}$, $\mathfrak{R}$, $\mathfrak{B}$), which is also a fundamental triad or named set.

**Definition 1.** *A named set (fundamental triad) in* ($\mathfrak{V}$, $\mathfrak{R}$, $\mathfrak{B}$) *is a triad* $\mathbf{X} = (X, f, N)$ *where the component* $X$ *of* $\mathbf{X}$ *is an object from* $\mathfrak{V}$*, which is called the support of* $\mathbf{X}$ *and denoted as* $S(\mathbf{X})$*; the component* $N$ *of* $\mathbf{X}$ *is an object from* $\mathfrak{B}$*, which is called the component of names (reflector) or set of names of* $\mathbf{X}$ *and denoted as* $N(\mathbf{X})$*; and* $f$ *is a relation (correspondence, tie, or connection) from* $\mathfrak{R}$ *between* $X$ *and* $N$*, which is called the naming correspondence (reflection) of* $\mathbf{X}$ *and denoted as* $r(\mathbf{X})$*.*

This means that $\mathbf{X} = (S(\mathbf{X}), r(\mathbf{X}), N(\mathbf{X}))$. Note that in $\mathbf{X}$, components $X$ and $N$ are not automatically sets, whereas $f$ is not necessarily a mapping or a function, even if $X$ and $N$ are sets. For instance, $X$ and $N$ are sets of words, and $f$ is an algorithm.

When a triad ($\mathfrak{V}$, $\mathfrak{R}$, $\mathfrak{B}$) of classes is not specified, we simply call $\mathbf{X} = (X, f, N)$ a named set or fundamental triad.

If we take a set of people $P$, the set of their names $L$, and the connection $c$ between people and their names, we obtain a basic named set (fundamental triad) of the form $(P, c, L)$.

Taking a set $X$ of things, the set $N$ of their names and the connection $f$ between things and their names, we have a basic named set (fundamental triad) of the form $(X, f, N)$ [31].

Sometimes people do not understand the difference between triples and triads. Whereas a *triple* is any group with three elements, a *triad* forms a system that has three connected elements or parts. This informal notion of a triple was formalized in mathematics in the form of a structure in an abstract category [32]. This structure is a triad formed from three fundamental triads. Consequently, it is a triad of the second order [1]. When mathematicians understood the composite organization of the *categorical triple*, they changed its name and called it a *monad* [33]. This explicated the relationship between Leibniz's monads and fundamental triads. Besides, a categorical monad is formed from triads and not the other way around. This is similar to the situation when named sets comprise sets as their special case of and not the other way around [1].

**Definition 2.**

(a) *A named set (fundamental triad)* $\mathbf{A} = (A, r, B)$ *in* ($\mathfrak{V}$, $\mathfrak{R}$, $\mathfrak{B}$) *is called basic if the relation (correspondence, tie, or connection) goes from* $A$ *to* $B$ *(cf. Diagram (1)).*

(b) *A named set (fundamental triad)* $\mathbf{A} = (A, r, B)$ *in* ($\mathfrak{V}$, $\mathfrak{R}$, $\mathfrak{B}$) *is called bidirectional if the relation (correspondence, tie, or connection) goes from* $A$ *to* $B$ *and from* $B$ *to* $A$ *(cf. Diagram (2)).*

(c) *A named set (fundamental triad)* $\mathbf{A} = (A, r, B)$ *in* ($\mathfrak{V}$, $\mathfrak{R}$, $\mathfrak{B}$) *is called cyclic if* $A = B$ *(cf. Diagram (3)).*

$$A \xrightarrow{\quad r \quad} B \tag{1}$$

$$A \xleftrightarrow{\quad r \quad} B \tag{2}$$

$$A \xrightarrow{\quad R \quad} A \tag{3}$$

Basic and bidirectional named sets are essentially different, even in the case of set-theoretical named sets where the naming correspondence is some binary relation [1]. In this case, a *directed binary relation* between sets $A$ and $B$ consists of pairs with the first element from $A$ and the second element from $B$. At the same time, an *undirected binary relation* between sets $A$ and $B$ consists of pairs where either the first element is from set $A$ and the second element is from set $B$ or the first element is from set $B$ and the second element is from set $A$. As a result, the naming correspondence of a basic named set is a directed binary relation. Contrary to this, the naming correspondence of a bidirectional named set is an undirected binary relation.

In many books, binary relations are defined as arbitrary subsets of the Cartesian product of two sets. In contrast to this, Bourbaki constructed a binary relation in the form of a set-theoretical named set in their highly formalized monograph on mathematics. Specifically, they describe a binary relation, $R$, in the form of an ordered triad $(A, G, B)$, taking $A$ and $B$ as arbitrary sets (or classes) and the naming correspondence, $G$, as a subset of the Cartesian product, $A \times B$ [34]. This shows that a conventional binary relation has the form of a basic set-theoretical named set.

In many cases, it is possible to represent bidirectional set-theoretical named set with a naming correspondence, which is formed of two directed binary relations. In the case when one of these binary relations is empty, an arbitrary basic set-theoretical named set becomes a special form of a bidirectional set-theoretical named set. In addition, it is possible to represent any bidirectional set-theoretical named set as a composition of some basic set-theoretical named sets.

If we consider two people who are exchanging messages, such as e-mails, or are talking, we obtain a set-theoretical named set $(X, f, Y)$, where the sets $X$ and $Y$ consist of people, and the naming correspondence, $f$, comprises messages that go between these individuals. A computer network is an example of a cyclic named set $(X, f, X)$ in which the set $X$ consists of computers, and the naming correspondence $f$ consists of all connections between these computers.

Named sets have diverse types and categories [1]. In particular, by varying the classes $\mathfrak{V}$, $\mathfrak{R}$, and $\mathfrak{B}$, we obtain different classes of named sets in the triad $(\mathfrak{V}, \mathfrak{R}, \mathfrak{B})$.

**Definition 3.** *Named sets in* $(\mathfrak{V}, \mathfrak{R}, \mathfrak{B})$ *are called set-based if* $\mathfrak{V}$ *and* $\mathfrak{B}$ *are classes of sets.*

As a host of mathematical constructions are set-based named sets, they are imperative for mathematics. In addition, set-based named sets play a considerable role in networking and computation due to the fact that data have the form of set-theoretical constructions, such as stacks, records, lists, or arrays.

Let us consider some principal classes of set-based named sets in $(\mathfrak{V}, \mathfrak{R}, \mathfrak{B})$.

**Definition 4.** *Named sets in* $(\mathfrak{V}, \mathfrak{R}, \mathfrak{B})$ *are called:*

- *Set-theoretical when* $\mathfrak{R}$ *consists of binary relations between sets;*
- *Algorithmic when* $\mathfrak{R}$ *consists of algorithms;*
- *Automaton named sets when* $\mathfrak{R}$ *consists of automata;*
- *Elementary when* $\mathfrak{V}$ *and* $\mathfrak{B}$ *consist of sets with a single element and* $\mathfrak{R}$ *consists of one tie (relation, connection, or correspondence) between two elements;*
- *Labeled when* $\mathfrak{R}$ *consists of labeled ties (relations, connections, or correspondences);*
- *Labeled set-theoretical when* $\mathfrak{R}$ *consists of labeled binary relations between sets;*
- *Normalized when* $\mathfrak{R}$ *consists of projective ties (relations, connections, or correspondences);*
- *Conormalized when* $\mathfrak{R}$ *consists of totally defined ties (relations, connections, or correspondences);*

- *Binormalized when they are both normalized and conormalized;*
- *Partially functional when $\Re$ consists of partial functions;*
- *Functional when $\Re$ consists of total functions;*
- *Partially cofunctional when $\Re$ is a set of binary relations, the inverses of which are partial functions;*
- *Cofunctional when $\Re$ is a set of binary relations, the inverses of which are total functions;*
- *Categorical when each element in $\mathfrak{A}$ and $\mathfrak{B}$ is a single object that belongs to some category C and each correspondence from $\Re$ consists of morphisms between an object, A, from $\mathfrak{A}$ and between an object B from $\mathfrak{B}$;*
- *Individually named when $\Re$ consists of bijections;*
- *Dynamic when $\Re$ consists of processes;*
- *Topological when $\mathfrak{A}$ and $\mathfrak{B}$ consist of topological spaces and $\Re$ consists of continuous mappings;*
- *Algebraic when $\mathfrak{A}$ and $\mathfrak{B}$ consist of algebraic systems and $\Re$ consists of homomorphisms.*

**Remark 1.** *Named sets from some of these classes of named sets are individually defined in* [1,18,30]. *Examples of named sets from the defined classes:*

1. *Cyclic set-theoretical named sets are graphs.*
2. *Cyclic labeled set-theoretical named sets are labeled graphs and hypergraphs.*
3. *A set-theoretical named set **A** = (A, f, B) where f is a function, is a functional set-theoretical named set.*
4. *An arrow (morphism) in a category is a categorical named set, as well as an elementary named set.*
5. *A triad (A, H(A,B), B), where H(A,B) is the set of all morphisms between objects A and B in a category K is a categorical named set.*
6. *All namespaces are set-theoretical or algorithmic named sets.*
7. *Fiber spaces are topological named sets.*
8. *Sheaves are topological named sets.*

**Remark 2.** *By introducing set-based named sets, we implicitly assume the existence of two kinds of object sets and named sets. The theory of named sets* [1] *allows for different formalizations of this situation. The simplest approach is to take set as the basic concept and construct named sets using sets, as was done early in the development of named set theory when only set-theoretical named sets were introduced and studied. A more advanced methodology takes named set as the basic concept of the theory, introducing sets as singlenamed sets* [1]. *The synthesized technique is based on two basic types of objects—sets and named sets. The most detailed system starts with three basic categories of objects: named sets; sets; and urelements, which are neither sets nor named sets. However, sets are simply singlenamed sets, whereas any object, including urelements, is a named set because it has a name. In particular, any urelement has the name "urelement".*

In what follows, we consider only set-based named sets.

## 3. Nested Named Sets: Basic Definitions and Constructions

Set-based named sets are nested when named sets appear as the elements of their supports or/and sets of names. Here is a formal definition.

**Definition 5** ([18]). *A set-based named set,* **A** *= (A, r, B) = (S(***A***), r(***A***), N(***A***)), is called:*

- *Nested from above when some elements from the support, N(**A**),are also named sets;*
- *Nested from below when some elements from the set of names, S(**A**),are also named sets;*
- *Amply nested when some elements from both sets, S(**A**)and N(**A**),are also named sets;*
- *Completely nested from above when all elements from the set, N(**A**), are also named sets;*
- *Completely nested from below when all elements from the set, S(**A**), are also named sets;*
- *Completely amply nested when all elements from both sets, S(**A**) and N(**A**), are also named sets.*

**Example 1.** *An important dynamic structure of computations is recursion* [35]. *A computational process that works with some input and gives some output is a fundamental triad (named set), with its input as the support, its output as the set of names, and the process itself as the correspondence between input and output. In this context, recursion is a nested named set (or more exactly, dynamic self-nested named set) because performance of recursion demands using the same recursion only with another input. This named set is nested from below because inputs on some levels are formed by recursion outputs on the previous level. This means that such inputs, in essence, are processes.*

**Example 2.** *When a basis B in the two-dimensional vector space* $\boldsymbol{V}$ *over the field of complex numbers* $\boldsymbol{C}$ *is defined, any vector v from* $\boldsymbol{V}$ *is represented by a pair of complex numbers* $(a_1, a_2)$ *from the space* $\boldsymbol{C}^2$. *As a result, we obtain the named set.*

$$\mathbf{Repr} = (\boldsymbol{V}, r_B, \boldsymbol{C}^2)$$

In it, the naming correspondence $r_B$ connects each vector from the space $\boldsymbol{V}$ with its numerical representation in the space $\boldsymbol{C}^2$ with respect to the basis $B$.

This is an amply nested named set because the complex vector space $\boldsymbol{V}$ has a complex construction with operations, relations, and their properties. For instance, identities in the vector space $\boldsymbol{V}$ define its properties. Consequently, the space $\boldsymbol{V}$ has the structure of a named set, namely:

$$\boldsymbol{V} = (V, p, \{O, R, P\})$$

In this named set, the support $V$ consists of vectors from the space $\boldsymbol{V}$; the naming correspondence, $p$, connects vectors with their properties, operations, and relations; the set $O$ consists of operations in $\boldsymbol{V}$; the set $R$ consists of relations in $\boldsymbol{V}$; and the set $P$ consists of properties of $\boldsymbol{V}$.

As a result, $\boldsymbol{V}$ is an amply nested named set because any binary relation, operation, or property is a named set [1].

At the same time, $\boldsymbol{C}^2$ is also a named set because each pair (two-dimensional vector) of complex numbers $(a_1, a_2)$ is a named set of the following form:

$$(X, f, \{1, 2, 3\})$$

In this named set, the set $X$ consists of two complex numbers, and the naming correspondence $f$ connects each of these numbers to the natural number indicating its position in the vector $(a_1, a_2)$. Specifically, $a_1$ is connected to 1, and $a_2$ is connected to 2. If we take the named set of the vector (2, 3), then the number 2 is connected to 1, whereas the number 3 is connected to 2.

The named set $(X, f, \{1, 2, 3\})$ is also amply nested because each number is (represented by) a named set, and each digit is (represented by) a named set as a symbol [1].

**Example 3.** *Let us consider* $3 \times 3$ *matrices of real numbers. The same reasoning as before shows that each such a matrix is a nested named set because any real number is a named set.*

**Example 4.** *Any algorithm or program is a named set in its complete representation. The full description of an algorithm/program must include a description of possible inputs I; a description of possible outputs O; and a constructive compressed description of the process DP. As a result, we obtain a named set (fundamental triad) (I, DP, O). Note that usually, only the constructive compressed description of the process is called an algorithm or a program.*
*A subroutine is a nested algorithm/program because it is an algorithm or a program in its own right.*

**Example 5.** *Let us consider chess. Chess is an abstract strategy game with no hidden information, in which each position emerges by conditionally emergent extraction when a player chooses the next move.*

Any position in a chess party is a named set, **CH**. Its support P consists of two groups: white and black pieces. At the beginning of the game, each group has 16 pieces. In the process of the game, some pieces are eliminated from the support. The set of names is the chess board B, which is formally represented by a two-dimensional 8 × 8 array or matrix containing 64 squares. The name of each piece from the support is the coordinate of a square on the chess board. The naming correspondence, c, connects each piece with its position (place) on the board B. As a result, for each position, we have the named set (fundamental triad).

$$\mathbf{CH} = (\mathrm{P}, c, \mathrm{B})$$

At the same time, each piece has a name, e.g., pawn or king, and this name is connected to the rules of the piece movements, as well as to the physical image representing this piece. Traditionally, the image was a material thing. When people started printing chess positions and displaying them on the screen, images of pieces became pictures on paper or on the screen.

This means that each piece is also a named set. Consequently, a position is a nested named set.

Because squares of the board have their coordinates and there are relations between them, each square on the board, B, is also a named set.

In the process of the game, the set of names (the board) remains the same, whereas the naming correspondence changes with each move, and the support (the set of pieces) changes from time to time. Each move is an operation with this named set, which is performed according to the definite chess rules. This operation is called a mapping of named sets in the theory of named sets [1].

Many other abstract strategy games, such as GO or checkers, as well as many real-life games, can be represented as processes of sequential application of operations with named sets.

As nested named sets form a special class of named sets, we need to distinguish nested named sets from other named sets.

**Definition 6.** *Named sets that are not nested named sets are called plain named sets.*

**Remark 3.** *It is possible to treat any nested named set as a plain named set, ignoring its nested structure.*

Named sets that belong to a nested named set comprise different levels [30].

**Definition 7.**

*(a)* *If* **X** *is a nested named set, then it is called the top named set, and it has level 0 with respect to its nesting.*
*(b)* *The first-level named sets of* **X** *are named sets that are either elements of the set of names, N(***X***), or elements of the support S(***X***).*
*(c)* *The nth-level named sets of* **X** *are named sets that are either elements of the set of names, N(***Z***) or elements of the support S(***Z***) for some (n -1)th-level named set,* **Z** *of* **X**.

**Remark 4.** *It is necessary to make a distinction between named sets nested in some named set* **X** *and named subsets of* **X** [1]. *The difference is similar to the difference between subsets and elements of a set.*

**Example 6**. *Let us consider the named set* **CH** *= (P, c, B), representing a position in a chess party. Each piece is a first-level named set of* **CH**, *whereas the named set* **CP** *= (Pn, c, B), which represents the positions of all pawns in* **CH,** *is a named subset of* **CH**.

**Remark 5.** *In general, a named set can have different levels in another named set as the following example demonstrates.*

**Example 7**. *Let us consider a list* **L** *that represents people and their characteristics. For instance, each element of* **L** *consists of the name of a person, her/his height, weight, age, and the number of this element in the list. In essence,* **L** *is a named set with characteristics forming the support and names of people constituting the set of names. Moreover, it is a nested named set because each characteristic is also a named set (fundamental triad), for example, height equal to 6 ft. These characteristics have level 1 in* **L**. *The considered characteristics are named numbers, for example, 6 ft or 150 lb. Consequently, these characteristics are named sets of level 2 in* **L**. *The (abstract) numbers, which form the support of the later named sets, have level 3 in* **L**. *At the same time, the same number, e.g., 6, can be also the number of the top named set, i.e., the list* **L**, *and thus have the level 1 in* **L**.

**Proposition 1.** *If a named set* **Y** *has level n in a nested named set* **X** *and a named set* **Z** *has level m in a nested named set* **Y**, *then the named set* **Z** *has level n + m in the nested named set* **X**.

*Proof* is performed by induction on the levels of named sets.

**Proposition 2.** *If a named set* **Y** *is a named subset of a nested named set* **X** *and a named set* **Z** *has level m in a nested named set* **Y**, *then the named set* **Z** *has level m in the nested named set* **X**.

*Proof* is performed by induction on the levels of named sets.

## 4. Operations with Nested Named Sets

Here, we study operations with set-theoretical completely amply nested named sets. In this case, reflections in named sets are binary relations, and all elements from the support and reflectors of these named sets are also set-theoretical named sets [1].

The intricate structure of nested named sets implies that there are three types of operations with nested named sets:

*Outer operations* are operations with named sets without taking into account their nested structure, i.e., without changing named sets that are nested.

*Inner operations* are operations with named sets that are applied to the named sets nested within them, i.e., operations that cause changes (transformations) of named sets are induced by changes (transformations) of named sets nested within them.

*Combined operations* are operations with named sets based on their nested structure, i.e., operations that cause direct changes (transformations) of named sets and changes (transformations)induced by changes (transformations) of named sets nested within them.

**Example 8**. *Let us consider such an operation as the unification* $\upsilon$ *of a set-theoretical named set. In it, all elements from the set of names are changed for one element, i.e., the set of names becomes a one-element set. The result of unification is a singlenamed set, which is, in essence, an ordinary set* [1]. *By its definition, unification* $\upsilon$ *is an outer operation with named sets.*

**Example 9**. *As an example of an inner operation, it is possible to take the inner unification* $\upsilon$ *of a named set. In it, if* **X** *is a nested named set, then the first-level named sets nested in* **X** *are unified.*

**Example 10**. *Let us consider the following operation. Taking a nested named set* **X**, *we delete all elements from the support of* **X** *that are not singlenamed sets and connections of these elements. This is a combined operation with named sets.*

There are two classes of outer operations with nested named sets.

*Free outer operations* with nested named sets do not depend on the structure of nesting.

*Conditional outer operations* with nested named sets are dependent on the structure of nesting.

*First-level conditional outer operations* with nested named sets are dependent on the first-level nested named sets of the operated named sets.

*n-level conditional outer operations* with nested named sets are dependent on the nested named sets of level *n* of the operated named sets.

*Flat first-level conditional outer operations* with nested named sets are dependent only on the first-level nested named sets of the operated named sets.

*Flat n-level conditional outer operations* with nested named sets are dependent only on the nested named sets of level $n$ of the operated named sets.

*Free outer operations with nested named sets* are operations with plain named sets, which are studied in [1].

Here, we consider some inner operations with amply nested set-theoretical named sets, i.e., we assume that all elements of the supports and sets of names in given named sets are also set-theoretical named sets. For simplicity here, we define and study only first-level inner and combined operations, i.e., operations that involve only the top named sets and named sets that are elements of the support and the set of names of the top named set, i.e., first-level named sets. First, we introduce and study set-theoretical operations. The basic operations with sets are union and intersection. Analyzing the counterparts of these operations for named sets, we find that there are six such operations [1]. For nested named sets, there are even more unions and intersections, which are introduced and studied below.

**Definition 8.** *The first-level disjunctive union* $\mathbf{X} \cup_{\mathrm{d}} \mathbf{Y}$ *of two nested named sets* $\mathbf{X} = (X, f, N)$ *and* $\mathbf{Y} = (Y, g, M)$ *is defined as the named sets* $\mathbf{Z} = (Z, h, Q)$*, in which:*

$$Z = \{Z_{ij} = X_i \cup Y_j;\ X_i \in X\ \&\ Y_j \in Y\}$$

$$Q = \{Q_{kr} = N_k \cup M_r;\ N_k \in N\ \&\ M_r \in M\}$$

The constructed relation $h$ has the following property.

**Condition D**. *The named set* $X_i \cup Y_j$ *is connected to the named set* $N_k \cup M_r$ *by h if and only if* $X_i$ *is connected to* $N_k$ *by f or* $Y_j$ *is connected to* $M_r$ *by g.*

Note that all elements, $X_i$ from $X$; $Y_j$ from $Y$; $N_k$ from $N$; and $M_r$ from $M$, are named sets. Here and in what follows, $\cup$ is the union of (flat) named sets studied in [1].

Let us study properties of the first-level disjunctive union of nested named sets. It has many properties similar to but not identical to those of union of sets.

By definition, the empty named set $\mathbf{\Lambda} = (\varnothing, \varnothing, \varnothing)$ is a nested named set, and we have the following result.

**Theorem 1** (Identity Law)**.** $\mathbf{X} \cup_d \mathbf{\Lambda} = \mathbf{\Lambda} \cup_d \mathbf{X} = \mathbf{X}$ *for any nested named set* $\mathbf{X} = (X, f, N)$.

Indeed, $\mathbf{\Lambda} = (\varnothing, \varnothing, \varnothing)$, whereas $X_i \cup \varnothing = X_i$, $r \cup \varnothing = r$, and $N_k \cup \varnothing = N_k$.

This result shows that the empty named set $\mathbf{\Lambda}$ is the identity element with respect to first-level disjunctive union.

**Theorem 2** (Commutative Law). *The first-level disjunctive union of named sets is commutative, i.e.,* $\mathbf{X} \cup_d \mathbf{Y} = \mathbf{Y} \cup_d \mathbf{X}$ *for any nested named sets* $\mathbf{X}$ *and* $\mathbf{Y}$.

**Proof**. Let us take two named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, building their first-level disjunctive unions $\mathbf{X} \cup_{\mathrm{d}} \mathbf{Y} = (Z, h, Q)$ and $\mathbf{Y} \cup_{\mathrm{d}} \mathbf{X} = (V, k, R)$. Then, according to Definition 8, we have:

$$Z = \{Z_{ij} = X_i \cup Y_j\ ;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{Q_{kr} = N_k \cup M_r;\ N_k \in N \text{ and } M_r \in M\}$$

$$\mathrm{V} = \{\mathrm{V}_{\mathrm{ij}} = \mathrm{Y}_{\mathrm{i}} \cup \mathrm{X}_{\mathrm{j}};\ \mathrm{X}_{\mathrm{i}} \in \mathrm{X} \text{ and } \mathrm{Y}_{\mathrm{j}} \in \mathrm{Y}\}$$

$$R = \{R_{kr} = M_k \cup N_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$ satisfies Condition D.

The named set $X_i \cup Y_j$ is connected to the named set $N_k \cup M_r$ by $h$ if and only if $X_i$ is connected to $N_k$ by $f$ or $Y_j$ is connected to $M_r$ by $g$.

The relation $k$ also satisfies Condition D.

The named set $X_i \cup Y_j$ is connected to the named set $N_k \cup M_r$ by $h$ if and only if $Y_j$ is connected to $M_r$ by $g$ or $X_i$ is connected to $N_k$ by $f$.

The union of (plain) named sets is a commutative operation [1]. This means that for all named sets $X_i \in X$ and $Y_j \in Y$, we have $Z_{ij} = V_{ij}$, and for all named sets $M_i \in M$ and $N_j \in N$, we have $Q_{ij} = R_{ij}$. Consequently, we have $Z = V$ and $Q = R$

In addition, relations $h$ and $k$ are constructed in the same way and therefore coincide. Then, by definition, we have:

$$\mathbf{X} \cup_{\mathrm{d}} \mathbf{Y} = \mathbf{Y} \cup_{\mathrm{d}} \mathbf{X}$$

Theorem is proved. □

**Theorem 3** (Associative Law)**.** *The first-level disjunctive union is associative, i.e.,* $\mathbf{X} \cup_d (\mathbf{Y} \cup_d \mathbf{Z}) = (\mathbf{X} \cup_d \mathbf{Y}) \cup_d \mathbf{Z}$ *for any nested named sets* **X**, **Y** *and* **Z**.

*Proof* is similar to the proof of Theorem 3. However, instead of the commutativity of the union of (plain) named sets, we use the associativity of the union of (plain) named sets [1].

**Theorem 4** (Normalization Law). *For any non-empty nested named sets* **X** *and* **Y**, *their first-level disjunctive union* $\mathbf{Y} \cup_d \mathbf{X}$ *is a normalized named set if and only if at least one of the named sets* **X** *and* **Y** *is a normalized named set*.

**Proof**. *Sufficiency*. Let us take two non-empty named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, and suppose that the named set **X** is normalized. Building their first-level disjunctive union $\mathbf{X} \cup_{\mathrm{d}} \mathbf{Y} = (Z, h, Q)$, we have:

$$Z = \{Z_{ij} = X_i \cup Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{Q_{kr} = N_k \cup M_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$ satisfies Condition D.

The named set $X_i \cup Y_j$ is connected to the named set $N_k \cup M_r$ by $h$ if and only if $X_i$ is connected to $N_k$ by $r$ or $Y_j$ is connected to $M_r$ by $q$.

As the named set **X** is normalized, for any element $b = N_k$ from $N$, there is an element $a = X_i$ from $X$ connected to $b$ by $r$. Consequently, for any element $N_k \cup M_r$ from $Q$, the element $X_i \cup Y_j$ from $Z$ is connected to $N_k \cup M_r$ by $h$. As $b$ is an arbitrary element from $N$, the named set $N_k \cup M_r$ is an arbitrary element from $Q$, and it has an element from $Z$ connected to it by $h$. This means that $\mathbf{Y} \cup_{\mathrm{d}} \mathbf{X}$ is a normalized named set.

When **Y** is a normalized named set, it is treated in the same way.

Sufficiency is proved.

*Necessity* is proved by contradiction. To do this, we take two non-empty named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, and suppose that neither **X** nor **Y** is normalized. By definition, this means that there is an element $b = N_k$ from $N$ for which there is no element $a = X_i$ from $X$ connected to $b$. Besides, there is an element $d = M_r$ from $N$ for which there is no element $c = Y_j$ from $Y$ connected to $d$. Consequently, there is no element $X_i \cup Y_j$ from $Z$ connected to $N_k \cup M_r$. This means that $\mathbf{Y} \cup_{\mathrm{d}} \mathbf{X}$ is not a normalized named set.

Thus, if the first-level disjunctive union $\mathbf{Y} \cup_{\mathrm{d}} \mathbf{X}$ is a normalized named set, then at least one of the named sets **X** and **Y** must be normalized.

Theorem is proved. □

This theorem is complemented by the following result, which is implied by Theorem 1.

**Proposition 3.** *The named set* $\mathbf{X} \cup_d \mathbf{\Lambda} = \mathbf{\Lambda} \cup_d \mathbf{X}$ *is normalized if and only if* $\mathbf{X}$ *is a normalized named set.*

It is possible to characterize conormalization in a similar way to normalization.

**Theorem 5** (Conormalization Law)**.** *For any non-empty nested named sets* $\mathbf{X}$ *and* $\mathbf{Y}$*, their first-level disjunctive union* $\mathbf{X} \cup_d \mathbf{Y}$ *is a conormalized named set if and only if at least one of the named sets* $\mathbf{X}$ *and* $\mathbf{Y}$ *is a conormalized named set.*

*Proof* is similar to the proof of Theorem 4.

**Remark 6.** *It is also possible to prove Theorem 5 using the duality relation between a named set and its inverse* [1].

Theorem 5 is complemented by the following result, which is directly implied by Theorem 1.

**Proposition 4.** *The named set* $\mathbf{X} \cup_d \mathbf{\Lambda} = \mathbf{\Lambda} \cup_d \mathbf{X}$ *is conormalized if and only if* $\mathbf{X}$ *is a conormalized named set.*

Theorems 4 and 5 imply the following result.

**Corollary 1** (Binormalization Law)**.** *For any non-empty nested named sets* $\mathbf{X}$ *and* $\mathbf{Y}$*, their first-level disjunctive union* $\mathbf{Y} \cup_d \mathbf{X}$ *is a binormalized named set if and only if at least one of the named sets* $\mathbf{X}$ *and* $\mathbf{Y}$ *is a binormalized named set.*

Propositions 3 and 4 imply the following result.

**Corollary 2.** *The named set* $\mathbf{X} \cup_d \mathbf{\Lambda} = \mathbf{\Lambda} \cup_d \mathbf{X}$ *is binormalized if and only if* $\mathbf{X}$ *is a binormalized named set.*

**Remark 7.** *The analogues of Theorems 4 and 5 are not true for functional named sets because in a general case, the first-level disjunctive union of two nested named sets can be not functional (cofunctional) even when both named sets are functional (cofunctional), as the following example demonstrates.*

**Example 11.** *Let us consider nested named sets* $\mathbf{X} = (X, r, N)$ *and* $\mathbf{Y} = (Y, q, M)$*, where* $X = \{X_0\}$*,* $N = \{N_0\}$*,* $Y = \{Y_0\}$*,* $M = \{M_0\}$*,* $r$ *connects* $X_0$ *with* $N_0$*,* $q$ *connects* $Y_0$ *with* $M_0$*,* $X_0 = (T, p, P)$*,* $Y_0 = (V, t, U)$*,* $T = \{a\}$*,* $V = \{b\}$*, and* $N_0 = M_0 = (\{1\}, e, \{1\})$*, where* $e$ *is the identity mapping.*

Both named sets $\mathbf{X}$ and $\mathbf{Y}$ are functional. However, taking the first-level disjunctive union $\mathbf{Y} \cup_{\mathrm{d}} \mathbf{X} = (Z, h, Q)$, we see that it is not a functional named set because by construction, the element $X_0 \cup Y_0$ from its support $Z$ is connected by the relation $h$ to three elements:$N_0 \cup N_0 = N_0$, $M_0 \cup M_0 = M_0$, and $N_0 \cup M_0$ from $Q$.

**Theorem 6** (Idempotent Law)**.** $\mathbf{X} \cup_d \mathbf{X} = \mathbf{X}$ *for any nested named set* $\mathbf{X}$ *if and only if the support* $S(\mathbf{X})$ *and the set of names* $N(\mathbf{X})$ *are closed with respect to the unions of their elements while the naming relation* $r(\mathbf{X})$ *satisfies Condition D.*

**Proof.** *Sufficiency*. Let us take a nested named set $\mathbf{X} = (X, r, N)$ and suppose that it satisfies the conditions of the theorem. Building the first-level disjunctive union $\mathbf{X} \cup_{\mathrm{d}} \mathbf{X} = (Z, h, Q)$, we have $Z = \{X_0 \cup X_0 = X_0\}$ and $Q = \{N_0 \cup N_0 = N_0\}$. Then the support $\mathrm{S}(\mathbf{X})$ is closed with respect to the unions of its elements. Consequently, by Definition 8, we have:

$$\mathrm{S}(\mathbf{X} \cup_{\mathrm{d}} \mathbf{X}) = \mathrm{S}(\mathbf{X})$$

Besides, the set of names N(**X**) is also closed with respect to the unions of its elements. Consequently, by Definition 8, we have:

$$\mathrm{N}(\mathbf{X} \cup_{\mathrm{d}} \mathbf{X}) = \mathrm{N}(\mathbf{X})$$

As the relation $h$ satisfies Condition D, by construction, the relation $r$ satisfies this condition and thus, the relation $h$ coincides with the relation $r$. As a result, we obtain the necessary equality $\mathbf{X} \cup_{\mathrm{d}} \mathbf{X} = \mathbf{X}$.

Sufficiency is proved.

*Necessity*. If $\mathbf{X} = (X, r, N)$, $\mathbf{X} \cup_{\mathrm{d}} \mathbf{X} = (Z, h, Q)$, and $\mathbf{X} \cup_{\mathrm{d}} \mathbf{X} = \mathbf{X}$, then $X = Z$, i.e., the support, $\mathrm{S}(\mathbf{X}) = X$ is closed with respect to the unions of its elements, $N = Q$, i.e., the set of names $\mathrm{N}(\mathbf{X}) = N$ is closed with respect to the unions of its elements, and relations $h$ and $r$ coincide, and thus, $r$ satisfies Condition C

Theorem is proved. □

**Corollary 3.** $\mathbf{X} \cup_{\mathrm{d}} \mathbf{X} = \mathbf{X}$ *if the support* S(**X**) *consists of one non-empty element and the set of names* N(**X**) *also consists of only one non-empty element*.

**Remark 8.** *In a general case, the first-level disjunctive union of two nested named sets is not an idempotent operation, i.e., Theorem 6 is not true for all nested named sets, as the following example demonstrates.*

**Example 12**. *Let us take a nested named set* $\mathbf{X} = (X, r, N)$*, in which the set X contains exactly two non-empty named sets,* $X_1$ *and* $X_2$*, and the set of names N contains only one named set. Building the first-level disjunctive union* $\mathbf{X} \cup_{\mathrm{d}} \mathbf{X} = (Z, h, Q)$*, we see that the support Z contains three named sets,* $X_1$*,* $X_2$ *and* $X_1 \cup X_2$*, which are equal neither to* $X_1$ *nor to* $X_2$*. Thus, Z is not equal to X, and* $\mathbf{X} \cup_{\mathrm{d}} \mathbf{X}$ *is not equal to* **X**.
*One more binary operation with nested named sets is first-level strict disjunctive union.*

**Definition 9.** *The first-level strict disjunctive union of two nested named sets* $\mathbf{X} = (X, f, N)$ *and* $\mathbf{Y} = (Y, g, M)$, is defined as the named sets $\mathbf{Z} = (Z, h, Q) = \mathbf{X} \cup_{\mathrm{sd}} \mathbf{Y}$, *in which*:

$$Z = \{Z_{ij}=X_i \cup Y_j;\ X_i \in X,\ Y_j \in \& X_i \cap Y_j \neq \varnothing\} \cup \{X_i \in X;\ \forall Y_j \in Y(X_i \cap Y_j = \varnothing)\} \cup \{Y_i \in Y;\ \forall X_j \in X(Y_i \cap X_j = \varnothing)\}$$

$$Q = \{Q_{kr} = N_k \cup M_r;\ N_k \in N,\ M_r \in M\ \&\ N_k \cap M_r \neq \varnothing\} \cup \{N_i \in N;\ \forall M_j \in M\ (N_i \cap M_j = \varnothing)\} \cup \{M_i \in M;\ \forall N_j \in N\ (M_i \cap N_j = \varnothing)\}$$

The relation $h$ is constructed in the following way.

If $X_i \cap Y_j \neq \varnothing$, $X_i$ is connected to $N_k$ or $Y_j$ is connected to $M_r$ and $N_k \cap M_r \neq \varnothing$, then $X_i \cup Y_j$ is connected to $N_k \cup M_r$ by $h$.

If $X_i \cap Y_j \neq \varnothing$, $X_i$ is connected to $N_k$, $Y_j$ is connected to $M_r$, and $N_k \cap M_r = \varnothing$, then $X_i \cup Y_j$ is connected to $N_k$ and to $M_r$ by $h$.

If for any $Y_j \in Y$ $(X_i \cap Y_j = \varnothing)$, $X_i$ is connected to $N_k$ by $f$ and $\forall M_j \in M$ $(N_i \cap M_j = \varnothing)$, then $X_i$ is connected to $N_k$ by $h$.

If for any $X_j \in X$ $(Y_i \cap X_j = \varnothing)$, $Y_i$ is connected to $M_k$ by $f$ and $\forall N_j \in N$ $(M_i \cap N_j = \varnothing)$, then $Y_i$ is connected to $M_k$ by $h$.

If for any $Y_j \in Y$ $(X_i \cap Y_j = \varnothing)$, $X_i$ is connected to $N_k$ by $f$ and $N_k \cap M_r \neq \varnothing$, then $X_i$ is connected to $N_k \cup M_r$ by $h$.

If for any $X_j \in X$ $(Y_i \cap X_j = \varnothing)$, $Y_i$ is connected to $M_k$ by $f$ and $N_k \cap M_r \neq \varnothing$, then $Y_i$ is connected to $N_k \cup M_r$ by $h$.

Some properties of first-level strict disjunctive union are similar to properties of first-level disjunctive union, whereas others are dissimilar. For instance, Definition 9 implies the following results.

**Theorem 7** (Identity Law). $\mathbf{X} \cup_{\mathrm{sd}} \mathbf{\Lambda} = \mathbf{\Lambda} \cup_{\mathrm{sd}} \mathbf{X} = \mathbf{X}$ *for any nested named set* $\mathbf{X} = (X, f, N)$.

This means that the empty named set $\mathbf{\Lambda}$ is the identity element with respect to strict first-level disjunctive union.

**Theorem 8** (Commutative Law)**.** $\mathbf{X} \cup_{sd} \mathbf{Y} = \mathbf{Y} \cup_{sd} \mathbf{X}$ *for any nested named sets* **X** *and* **Y**.

First-level strict disjunctive union is intrinsically connected to the union of plain named sets. To show this, we use the following concept.

**Definition 10.** *Two named sets* **X** *and* **Y** *are disjunctive on the first level if any two named sets* $X_i$ *from* **X** *and* $Y_j$ *from* **Y** *do not intersect.*

Let us consider two nested named sets $\mathbf{X} = (X, f, N)$ and $\mathbf{Y} = (Y, g, M)$.

**Theorem 9.** *The first-level strict disjunctive union* $\mathbf{X} \cup_{sd} \mathbf{Y} = \mathbf{Y} \cup_{sd} \mathbf{X}$ *of two nested named sets* **X** *and* **Y** *coincides with their union as plain named sets if X and Y, as well as N and M, are disjunctive on the first level.*

One more binary operation with nested named sets is first-level conjunctive union.

**Definition 11.** *The first-level conjunctive union of two nested named sets* $\mathbf{X} = (X, f, N)$ *and* $\mathbf{Y} = (Y, g, M)$, is defined as the named sets $\mathbf{Z} = (Z, h, Q) = \mathbf{X} \cup_c \mathbf{Y}$, *in which:*

$$Z = \{X_i \cup Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{N_k \cup M_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$ is constructed in such a way that it satisfies the following condition.

**Condition C**. *The named set* $X_i \cup Y_j$ *is connected to the named set* $N_k \cup M_r$ *by h if and only if* $X_i$ *is connected to* $N_k$ *by f and* $Y_j$ *is connected to* $M_r$ *by g.*

Let us study the properties of the first-level conjunctive union of nested named sets.

**Definition 12.** *A disconnected named set* $\mathbf{\Lambda}_{X,N}$ *has the following form:*

$$\mathbf{\Lambda}_{X,N} = (X, \varnothing, N)$$

**Theorem 10.** $\mathbf{X} \cup_c \mathbf{\Lambda} = \mathbf{\Lambda} \cup_c \mathbf{X} = \mathbf{\Lambda}_{X,N}$ *for any nested named set* $\mathbf{X} = (X, f, N)$.

*Proof* is similar to the proof of Theorem 1.

**Theorem 11** (Commutative Law). $\mathbf{X} \cup_c \mathbf{Y} = \mathbf{Y} \cup_c \mathbf{X}$ *for any nested named sets* **X** *and* **Y**.

*Proof* is similar to the proof of Theorem 2.

**Theorem 12** (Associative Law). *The first-level conjunctive union is associative, i.e.,* $\mathbf{X} \cup_c (\mathbf{Y} \cup_c \mathbf{Z}) = (\mathbf{X} \cup_c \mathbf{Y}) \cup_c \mathbf{Z}$ *for any nested named sets* **X**, **Y**, *and* **Z**.

*Proof* is similar to the proof of Theorem 3.

**Theorem 13** (Normalization Law)**.** *For any non-empty nested named sets* **X** *and* **Y**, *their first-level conjunctive union* $\mathbf{Y} \cup_c \mathbf{X}$ *is a normalized named set if and only if both named sets* **X** *and* **Y** *are normalized named sets.*

**Proof**. *Sufficiency*. Let us take two named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, and suppose that both of them are normalized. Building their first-level conjunctive union $\mathbf{X} \cup_c \mathbf{Y} = (Z, h, Q)$, we have:

$$Z = \{Z_{ij} = X_i \cup Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{Q_{kr} = N_k \cup M_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$ satisfies Condition D.

The named set $X_i \cup Y_j$ is connected to the named set $N_k \cup M_r$ by $h$ if and only if $X_i$ is connected to $N_k$ by $r$ or $Y_j$ is connected to $M_r$ by $q$.

As the named set **X** is normalized, for any element $b = N_k$ from $N$, there is an element $a = X_i$ from $X$ connected to $b$ by $r$. As the named set **Y** is normalized, for any element $d = M_r$ from $M$, there is an element $c = Y_j$ from $Y$ connected to $d$ by $q$. Consequently, for any element $N_k \cup M_r$ from $Q$, the element $X_i \cup Y_j$ from $Z$ is connected to $N_k \cup M_r$. As $b$ is an arbitrary element from $N$, the named set $N_k \cup M_r$ is an arbitrary element from $Q$, and it has an element from $Z$ connected to it by $h$. This means that $\mathbf{Y} \cup_c \mathbf{X}$ is a normalized named set.

Sufficiency is proved.

We prove *necessity* by contradiction. Let us take two named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, and suppose that either **X** or **Y** is not normalized. For convenience, we assume that **X** is not normalized. By definition, this means that there is an element $b = N_k$ from $N$, for which there is no element $a = X_i$ from $X$ connected to $b$. Then, by the construction of the relation $h$ for any element $d = M_r$ from $M$, there is no element $X_i \cup Y_j$ from $Z$ connected to $N_k \cup M_r$. This means that $\mathbf{Y} \cup_c \mathbf{X}$ is not a normalized named set.

Thus, if the first-level conjunctive union $\mathbf{Y} \cup_c \mathbf{X}$ is a normalized named set, then both named sets **X** and **Y** must be normalized.

Theorem is proved. □

**Theorem 14** (Conormalization Law)**.** *For any nested named sets* **X** *and* **Y**, *their first-level conjunctive union* $\mathbf{Y} \cup_d \mathbf{X}$ *is a conormalized named set if and only if both named sets* **X** *and* **Y** *are conormalized*.

*Proof* is similar to the proof of Theorem 14.

Theorem 14 and 15 imply the following result.

**Corollary 4** (Binormalization Law)**.** *For any nested named sets* **X** *and* **Y**, *their first-level conjunctive union* $\mathbf{Y} \cup_d \mathbf{X}$ *is a binormalized named set if and only if both named sets* **X** *and* **Y**, *are binormalized.*

To continue with properties of first-level conjunctive union, we remind that if $\mathbf{X} = (X, f, N)$ is a named set, $a \in X$, and $f$ connects $a$ with $b$, then $b$ is called a name of $a$ [1].

**Theorem 15** (Functionality Law)**.** *For any non-empty nested named sets* **X** *and* **Y**, *their first-level conjunctive union* $\mathbf{Y} \cup_c \mathbf{X}$ *is a functional named set if and only if both* **X** *and* **Y** *are functional named sets*.

**Proof.** *Necessity*. Let us take two named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, and suppose that **X** is not functional and both named sets are not empty. By definition, this means that there is an element $a = X_i$ from $X$ that has two names $b$ and $c$, i.e., $a$ is connected to elements $b$ and $c$ by the relation $f$. Then, by the construction of the conjunctive union $\mathbf{Y} \cup_c \mathbf{X}$, any element $X_i \cup Y_j$ from $Z$ is connected to two elements from the set of names, $\mathrm{N}(\mathbf{Y} \cup_d \mathbf{X})$. This means that $\mathbf{Y} \cup_d \mathbf{X}$ is not a functional named set.

The case when **Y** is not a functional named set is treated in the same way.

Necessity is proved.

*Sufficiency*. Let us take two named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, and suppose that both of them are functional. First, we also suppose that these named sets are not empty. Building their first-level conjunctive union $\mathbf{X} \cup_d \mathbf{Y} = (Z, h, Q)$, we have:

$Z = \{Z_{ij} = X_i \cup Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$

$Q = \{Q_{kr} = N_k \cup M_r;\ N_k \in N \text{ and } M_r \in M\}$

The relation $h$ satisfies Condition D.

The named set $X_i \cup Y_j$ is connected to the named set $N_k \cup M_r$ by $h$ if and only if $X_i$ is connected to $N_k$ by $f$ or $Y_j$ is connected to $M_r$ by $g$.

As the named set **X** is functional, then any element $a = X_i$ from $X$ is connected to not more than one element $b = N_k$ from $N$ by the relation $r$. As the named set **Y** is functional, then any element $d = Y_j$ from $Y$ is connected to not more than one element $c = M_r$ from $M$ by the relation $q$.

Consequently, the element $X_i \cup Y_j$ is connected to not more than one element, $N_k \cup M_r$ from $Q$ by the relation $h$. This means that the first-level conjunctive union $\mathbf{X} \cup_c \mathbf{Y}$ is a functional named set.

Theorem is proved. □

Note that the empty named set $\mathbf{\Lambda}$ is functional. Therefore, if we have the first-level conjunctive union $\mathbf{X} \cup_c \mathbf{\Lambda}$, and **X** is functional, then $\mathbf{X} \cup_c \mathbf{\Lambda}$ is functional because according to Theorem 1, it is equal to $\mathbf{\Lambda}$.

**Theorem 16** (Cofunctionality Law)**.** *For any nested named sets* **X** *and* **Y**, *their first-level conjunctive union* $\mathbf{Y} \cup_c \mathbf{X}$ *is a cofunctional named set if and only if both* **X** *and* **Y** *are cofunctional named sets*.

*Proof* is similar to the proof of Theorem 15.

**Remark 9.** *It is also possible to prove Theorem 16 using the duality relation between a named set and its inverse* [1].

Theorems 15 and 16 imply the following result.

**Corollary 5** (Individualization Law)**.** *For any nested named sets* **X** *and* **Y**, *their first-level conjunctive union* $\mathbf{Y} \cup_c \mathbf{X}$ *is an individually named set if and only if both* **X** *and* **Y** *are individually named sets*.

We see that some properties of first-level conjunctive union are similar to properties of first-level disjunctive union, whereas others are dissimilar.

One more binary operation with nested named sets is strict first-level conjunctive union.

**Definition 13.** *The first-level strict conjunctive union of two nested named sets* $\mathbf{X} = (X, f, N)$ *and* $\mathbf{Y} = (Y, g, M)$ *is defined as the named sets* $\mathbf{Z} = (Z, h, Q) = \mathbf{X} \cup_{sc} \mathbf{Y}$, *in which*:

$$Z = \{Z_{ij} = X_i \cup Y_j;\ X_i \in X,\ Y_j \in Y \text{ and } X_i \cap Y_j \neq \varnothing\}$$

$$Q = \{Q_{kr} = N_k \cup M_r;\ N_k \in N,\ M_r \in M \text{ and } N_k \cap M_r \neq \varnothing\}$$

The relation,$h$, is constructed in the following way.

If $X_i \cap Y_j \neq \varnothing$, $X_i$ is connected to $N_k$, $Y_j$ is connected to $M_r$, and $N_k \cap M_r \neq \varnothing$, then $X_i \cup Y_j$ is connected to $N_k \cup M_r$ by $h$.

If $X_i \cap Y_j \neq \varnothing$, $X_i$ is connected to $N_k$,$Y_j$ is connected to $M_r$, and $N_k \cap M_r = \varnothing$, then $X_i \cup Y_j$ is connected to $N_k$ and to $M_r$ by $h$.

If for any $Y_j \in Y$ ($X_i \cap Y_j = \varnothing$), $X_i$ is connected to $N_k$ by $f$ and $\forall M_j \in M$ ($N_i \cap M_j = \varnothing$), then $X_i$ is connected to $N_k$ by $h$.

If for any $X_j \in X$ ($Y_i \cap X_j = \varnothing$), $Y_i$ is connected to $M_k$ by $f$ and $\forall N_j \in N$ ($M_i \cap N_j = \varnothing$), then $Y_i$ is connected to $M_k$ by $h$.

If for any $Y_j \in Y$ ($X_i \cap Y_j = \varnothing$), $X_i$ is connected to $N_k$ by $f$ and $N_k \cap M_r \neq \varnothing$, then $X_i$ is connected to $N_k \cup M_r$ by $h$.

If for any $X_j \in X$ ($Y_i \cap X_j = \varnothing$), $Y_i$ is connected to $M_k$ by $f$ and $N_k \cap M_r \neq \varnothing$, then $Y_i$ is connected to $N_k \cup M_r$ by $h$.

Some properties of first-level strict conjunctive union are similar to properties of first-level conjunctive union, whereas others are dissimilar. In particular, Definition 13 directly implies the following result.

**Theorem 17** (Identity Law)**.** $\mathbf{X} \cup_{sc} \mathbf{\Lambda} = \mathbf{\Lambda} \cup_{sc} \mathbf{X} = \mathbf{X}$ *for any nested named set* $\mathbf{X} = (X, f, N)$.

This means that the empty named set $\mathbf{\Lambda}$ is the identity element with respect to strict first-level conjunctive union.

**Theorem 18** (Commutative Law)**.** $\mathbf{X} \cup_{sc} \mathbf{Y} = \mathbf{Y} \cup_{sc} \mathbf{X}$ *for any nested named sets* $\mathbf{X}$ *and* $\mathbf{Y}$.

*Proof* is similar to the proof of Theorem 2.

One more binary operation with nested named sets is first-level disjunctive intersection.

**Definition 14.** *The first-level disjunctive intersection of two nested named sets* $\mathbf{X} = (X, f, N)$ *and* $\mathbf{Y} = (Y, g, M)$ is defined as the named sets $\mathbf{Z} = (Z, h, Q)$, *in which:*

$$Z = \{Z_{ij} = X_i \cap Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{Q_{kr} = N_k \cap M_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$ is constructed in the following way.

If $X_i$ is connected to $N_k$ by $f$ or $Y_j$ is connected to $M_r$ by $g$, then when $X_i \cap Y_j$ is not empty, it is connected to $N_k \cap M_r$ by $h$.

Here and in what follows, $\cap$ is the intersection of (flat) named sets studied in [1].

**Theorem 19** (Identity Law)**.** $\mathbf{X} \cap_d \mathbf{\Lambda} = \mathbf{\Lambda} \cap_d \mathbf{X} = \mathbf{\Lambda}$ *for any nested named set* $\mathbf{X} = (X, f, N)$.

Indeed, $\mathbf{\Lambda} = (\varnothing, \varnothing, \varnothing)$, while as $X_i \cap \varnothing = \varnothing$, $r \cap \varnothing = \varnothing$, and $N_k \cap \varnothing = \varnothing$.

This result shows that the empty named set $\mathbf{\Lambda}$ is the null element with respect to first-level disjunctive intersection.

**Theorem 20** (Commutative Law)**.** $\mathbf{X} \cap_d \mathbf{Y} = \mathbf{Y} \cap_d \mathbf{X}$ *for any nested named sets* $\mathbf{X}$ *and* $\mathbf{Y}$.

**Proof.** Let us take two named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, building their first-level disjunctive intersections $\mathbf{X} \cap_d \mathbf{Y} = (Z, h, Q)$ and $\mathbf{Y} \cap_d \mathbf{X} = (V, k, R)$. Then, according to Definition 8, we have:

$$Z = \{Z_{ij} = X_i \cap Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{Q_{kr} = N_k \cap M_r;\ N_k \in N \text{ and } M_r \in M\}$$

$$V = \{V_{ij} = Y_i \cap X_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$R = \{R_{kr} = M_k \cap N_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$ is constructed in the following way.

If $X_i$ is connected to $N_k$ by $r$ or $Y_j$ is connected to $M_r$ by $q$, then when $X_i \cap Y_j$ is not empty, it is connected to $N_k \cap M_r$ by $h$.

The relation $k$ is constructed in the following way.

If $X_i$ is connected to $N_k$ by $r$ or $Y_j$ is connected to $M_r$ by $q$, then $X_i \cap Y_j$ is connected to $N_k \cap M_r$ by $k$.

The intersection of (plain) named sets is a commutative operation [1]. This means that for all named sets $X_i \in X$ and $Y_j \in Y$, we have $Z_{ij} = V_{ij}$, and for all named sets $M_i \in M$ and $N_j \in N$, we have $Q_{ij} = R_{ij}$. Consequently, we obtain the equalities $Z = V$ and $Q = R$

In addition, relations $h$ and $k$ are constructed in the same way and therefore coincide. Then by definition, we have:

$$\mathbf{X} \cap_{\mathrm{d}} \mathbf{Y} = \mathbf{Y} \cap_{\mathrm{d}} \mathbf{X}$$

Theorem is proved. □

**Theorem 21** (Associative Law)**.** *The first-level disjunctive intersection is associative, i.e.,* $\mathbf{X} \cap_d (\mathbf{Y} \cap_d \mathbf{Z}) = (\mathbf{X} \cap_d \mathbf{Y}) \cap_d \mathbf{Z}$ *for any nested named sets* **X**, **Y**, *and* **Z**.

*Proof* is similar to the proof of Theorem 3. However, instead of the associativity of the union of (plain) named sets, we use the associativity of the intersection of (plain) named sets [1].

**Theorem 22** (Normalization Law)**.** *For any non-empty nested named sets* **X** *and* **Y**, *their first-level disjunctive intersection* $\mathbf{Y} \cap_d \mathbf{X}$ *is a normalized named set if at least one of the named sets* **X** *and* **Y** *is a normalized named set and each of the elements from its support has a non-empty intersection with some elements from the support of the other named set*.

**Proof.** Let us take two non-empty named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, supposing that the named set **X** is normalized. Building their first-level disjunctive intersection $\mathbf{X} \cap_{\mathrm{d}} \mathbf{Y} = (Z, h, Q)$, we have:

$$Z = \{Z_{ij} = X_i \cap Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{Q_{kr} = N_k \cap M_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$ is constructed in the following way.

If $X_i$ is connected to $N_k$ by $r$ or $Y_j$ is connected to $M_r$ by $q$, then when $X_i \cap Y_j$ is not empty, it is connected to $N_k \cap M_r$ by $h$.

As the named set **X** is normalized, for any element $b = N_k$ from $N$, there is an element $a = X_i$, from $X$ connected to $b$. By the initial conditions, there is an element $Y_j$ from $Y$ such that the element $X_i \cap Y_j$ from $Z$ is not empty. Consequently, for any element $N_k \cap M_r$ from $Q$, the element $X_i \cap Y_j$ from $Z$ is connected to $N_k \cap M_r$. As $b$ is an arbitrary element from $N$, $N_k \cap M_r$ is an arbitrary element from $Q$. This means that $\mathbf{Y} \cap_{\mathrm{d}} \mathbf{X}$ is a normalized named set.

The case when **Y** is a normalized named set is treated in the same way.

Theorem is proved. □

**Remark 10.** *In a general case, the first-level disjunctive intersection* $\mathbf{Y} \cap_d \mathbf{X}$ *is not always a normalized named set, even if both named sets* **X** *and* **Y** *are normalized, as the following example demonstrates*.

**Example 13.** *Let us consider nested named sets* $\mathbf{X} = (X, r, N)$ *and* $\mathbf{Y} = (Y, q, M)$, *where* $X = \{X_0\}$; $N = \{N_0\}$; $Y = \{Y_0\}$; $M = \{M_0\}$; $r$ *connects* $X_0$ *with* $N_0$; $q$ *connects* $Y_0$ *with* $M_0$; $X_0 = (T, p, P)$; $Y_0 = (V, t, U)$; $T = \{a\}$, $V = \{b\}$; *and* $N_0 = M_0 = (\{1\}, e, \{1\})$, *where* $e$ *is the identity mapping. Bothnamed sets* **X** *and* **Y** *are normalized. However, taking the first-level disjunctive intersection* $\mathbf{Y} \cap_{\mathrm{d}} \mathbf{X} = (Z, h, Q)$, *we see that it is not a normalized named set because its support is empty, as* $X_0 \cap Y_0 = \mathbf{\Lambda}$, *whereas its set of names,* $Q$, *contains the named set* $N_0$ *and is therefore not empty.*

It is possible to characterize conormalization in a similar way to normalization.

**Theorem 23** (Conormalization Law)**.** *For any non-empty nested named sets* **X** *and* **Y**, *their first-level disjunctive intersection* $\mathbf{Y} \cap_d \mathbf{X}$ *is a conormalized named set if at least one of the named sets* **X** *and* **Y** *is a conormalized named set and each of the elements from its set of names has a non-empty intersection with some elements from the set of names of the other named set*.

*Proof* is similar to the proof of Theorem 22.

**Remark 11.** *It is also possible to prove Theorem 23 using the duality relation between a named set and its inverse* [1].

**Remark 12.** *In a general case, the first-level disjunctive intersection* $\mathbf{Y} \cap_d \mathbf{X}$ *is not always a conormalized named set, even if both named sets* **X** *and* **Y** *are conormalized.*

Theorems 22 and 23 imply the following result.

**Corollary 6** (Binormalization Law)**.** *For any non-empty nested named sets* **X** *and* **Y**, *their first-level disjunctive intersection* $\mathbf{Y} \cap_d \mathbf{X}$ *is a binormalized named set if and only if at least one of the named sets* **X** *and* **Y** *is a binormalized named set, each of the elements from its support has a non-empty intersection with some elements from the support of the other named set, and each of the elements from its set of names has a non-empty intersection with some elements from the set of names of the other named set.*

**Remark 13.** *The first-level disjunctive intersection of two nested named sets can be non-cofunctional, even if both nested named sets are cofunctional.*

**Remark 14.** *In a general case, the first-level disjunctive intersection of two nested named sets is not an idempotent operation, i.e., an analogue of Theorem 6 is not true for first-level disjunctive intersection and all nested named sets, as the following example demonstrates.*

**Example 14**. *Let us take a nested named set* $\mathbf{X} = (X, r, N)$, *in which the set X contains exactly two different non-empty named sets,* $X_1$ *and* $X_2$ *with anon-empty intersection, whereas the set of names N contains only one named set. Building the first-level disjunctive intersection* $\mathbf{X} \cap_d \mathbf{X} = (Z, h, Q)$, *we see that the support, Z, contains three named sets,* $X_1$, $X_2$, *and* $X_1 \cap X_2$, *which is equal to neither to* $X_1$ *nor* $X_2$. *Therefore, Z is not equal to X, and* $\mathbf{X} \cap_d \mathbf{X}$ *is not equal to* **X**.

One more binary operation with nested named sets is first-level conjunctive union.

**Definition 15.** *The first-level conjunctive intersection of two nested named sets* $\mathbf{X} = (X, f, N)$ *and* $\mathbf{Y} = (Y, g, M)$ *is defined as the named sets* $\mathbf{Z} = (Z, h, Q)$, *in which:*

$$Z = \{X_i \cap Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{N_k \cap M_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$, is constructed in the following way.

If $X \cap Y_j$ if it is not empty, $X_i$ is connected to $N_k$ by $f$ and $Y_j$ is connected to $M_r$ by $g$, then $X_i \cap Y_j$ is connected to $N_k \cap M_r$ by $h$.

Let us suppose that in non-empty named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, each of the elements from $X$ has a non-empty intersection with some elements from $Y$, and each of the elements from $Y$ has a non-empty intersection with some elements from $X$.

**Theorem 24** (Functionality Law)**.** *The first-level conjunctive intersection* $\mathbf{Y} \cap_c \mathbf{X}$ *is a functional named set if and only if both* **X** *and* **Y** *are functional named sets.*

**Proof.** *Sufficiency.* Let us take two named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, and suppose that both of them are functional. First, we also suppose that these named sets are not empty. Building their first-level conjunctive intersection $\mathbf{X} \cap_c \mathbf{Y} = (Z, h, Q)$, we have:

$$Z = \{Z_{ij} = X_i \cap Y_j;\ X_i \in X \text{ and } Y_j \in Y\}$$

$$Q = \{Q_{kr} = N_k \cap M_r;\ N_k \in N \text{ and } M_r \in M\}$$

The relation $h$ is constructed in the following way.

If $X_i$ is connected to $N_k$ by $r$ and $Y_j$ is connected to $M_r$ by $q$, then when $X_i \cap Y_j$ is not empty, it is connected to $N_k \cap M_r$ by $h$.

As the named set **X** is functional, any element $a = X_i$, from $X$ is connected to not more than one element $b = N_k$ from $N$ by the relation $r$. As the named set **Y** is functional, any element $d = Y_j$ from $Y$ is connected to not more than one element $c = M_r$ from $M$ by the relation $q$.

Consequently, the element $X_i \cap Y_j$ is connected to not more than one element $N_k \cap M_r$, from $Q$ by the relation $h$. This means that the first-level conjunctive intersection $\mathbf{X} \cap_c \mathbf{Y}$ is a functional named set.

Sufficiency is proved.

*Necessity*. Let us take two named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$ and suppose that **X** is not functional, whereas both named sets are not empty. By definition, this means that there is an element $a = X_i$ from $X$ that has two names $b$ and $c$, i.e., $a$ is connected to $b$ and $c$ by the relation $r$. By the initial assumption, there is an element $d = Y_j$ from $Y$ such that $a \cap d \neq \emptyset$. Then, by the construction of their first-level conjunctive intersection $\mathbf{X} \cap_c \mathbf{Y} = (Z, h, Q)$, the element $a \cap d = X_i \cap Y_j$ from $Z$ is connected to $b \cap c$ from $Q$. As $a$ is an arbitrary element from $X$, this means that $\mathbf{Y} \cap_c \mathbf{X}$ is not a functional named set.

The case when **Y** is not a functional named set is treated in the same way.

Thus, if the first-level conjunctive intersection $\mathbf{Y} \cap_c \mathbf{X}$ is a functional named set, then both of the named sets **X** and **Y** must be functional.

Theorem is proved. □

Let us suppose that in non-empty named sets $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, each of the elements from $N$ has a non-empty intersection with some elements from $M$, and each of the elements from $M$ has a non-empty intersection with some elements from $N$.

**Theorem 25** (Cofunctionality Law)**.** *For any nested named sets* **X** *and* **Y**, *their first-level conjunctive intersection* $\mathbf{Y} \cap_c \mathbf{X}$ *is a cofunctional named set if and only if both* **X** *and* **Y** *are cofunctional named sets.*

*Proof* is similar to the proof of Theorem 6.

**Remark 15.** *It is also possible to prove Theorem 25 using the duality relation between a named set and its inverse* [1].

Let us assume that in non-empty named sets, $\mathbf{X} = (X, r, N)$ and $\mathbf{Y} = (Y, q, M)$, each of the elements from $X$ has a non-empty intersection with some elements from $Y$, each of the elements from $Y$ has a non-empty intersection with some elements from $X$, each of the elements from $N$ has a non-empty intersection with some elements from $M$, and each of the elements from $M$ has a non-empty intersection with some elements from $N$. Then, Theorems 24 and 25 imply the following result.

**Corollary 7** (Individualization Law)**.** *For any nested named sets* **X** *and* **Y**, *their first-level conjunctive intersection* $\mathbf{Y} \cup_c \mathbf{X}$ *is an individually named set if and only if both* **X** *and* **Y** *are individually named sets*.

## 5. Conclusions

Information processing, as the base for natural and artificial intelligence, consists of transformations of data and knowledge. As data and knowledge are represented by named sets, these transformations are represented by operations transforming named sets. Here, various properties of binary operations on systems of nested named sets are obtained because nesting structures are important for data and knowledge representations, as well as for programming languages and algorithms. The goal is to form a base of operations for algorithmic systems of artificial intelligence.

When we built here binary operations with nested named sets, unions, and intersections, we used only one operation of the union of (plain) named sets and only one operation of the intersection of (plain) named sets introduced and studied in [1]. These operations are used when people combine data and knowledge or find common information from different sources. Naturally, AI systems must also perform such operations, and knowing their properties allows us to build more efficient software and hardware for AI systems.

At the same time, it is necessary to understand that there are other operations with (plain) named sets that are performed by people and could be useful for AI systems. Thus, an interesting and practical problem for future research is to use these operations to build operations for nested named sets and study their properties.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The author would like to express gratitude to the second reviewer for useful remarks.

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Burgin, M. *Theory of Named Sets, Mathematics Research Developments*; Nova Science: New York, NY, USA, 2011.
2. Burgin, M. Unified Foundations of Mathematics. Preprint Mathematics LO/0403186. 2004, p. 39. Available online: https://arxiv.org/abs/math/0403186 (accessed on 29 December 2021).
3. Ringland, N.; Dai, X.; Hachey, B.; Karimi, S.; Paris, C.; Curran, J.R. NNE: A Dataset for Nested Named Entity Recognition in English Newswire, Preprint in Computation and Language (cs.CL). *arXiv* **2019**, arXiv:1906.01359.
4. Katiyar, A.; Cardie, C. Nested named entity recognition revisited. In *Proceedings of the NAACL-HLT'18*; New Orleans, LO, USA, 1–8 June 2018, Association for Computational Linguistics (ACL): Stroudsburg, PA, USA, 2018; pp. 861–871.
5. Burgin, M.; Smith, M.L. A Theoretical Model for Grid, Cluster and Internet Computing. In *Selected Topics in Communication Networks and Distributed Systems*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010; pp. 485–535.
6. Burgin, M. Triadic structures in interpersonal communication. *Information* **2018**, *9*, 283. [CrossRef]
7. Barr, M. The Chu construction. *Theory Appl. Categ.* **1996**, *2*, 17–35.
8. Chu, P.-H. *Constructing *-Autonomous Categories, Appendix to Barr (1979), *-Autonomous Categories*; Lecture Notes in Mathematics 752; Springer: New York, NY, USA, 1979.
9. Bray, T.; Hollander, D.; Layman, A. Namespaces in XML. March 1998. Available online: http://www.w3.org/TR/1998/WD-xml-names-19980327 (accessed on 29 December 2021).
10. Nocedal, A.S.; Gerrikagoitia Arrien, J.K.; Burgin, M. A mathematical model for managing XML data. *Int. J. Metadata Semant. Ontol. IJMSO* **2011**, *6*, 56–73. [CrossRef]
11. Date, C.J. *An Introduction to Database Systems*; Addison Wesley: Boston, MA, USA; San Francisco, CA, USA; New York, NY, USA, 2004.
12. Elmasri, R.; Navathe, S.B. *Fundamentals Database Systems*; Addison-Wesley Publishing Company: Reading, MA, USA, 2000.
13. Snodgrass, R.T.; Jensen, C.S. *Developing Time-Oriented Database Applications in SQL*; Morgan Kaufmann: Burlington, MA, USA, 1999.
14. Burgin, M. Structural Organization of Temporal Databases. In Proceedings of the 17th International Conference on Software Engineering and Data Engineering (SEDE-2008), ISCA, Los Angeles, CA, USA, 30 June–2 July 2008; pp. 68–73.
15. Zellweger, H.P. Unifying Data Relations to Enable End-Users to Navigate Over Relation Data. In Proceedings of the 21st International Conference on Software Engineering and Data Engineering (SEDE-2012), Los Angeles, CA, USA, 27–29 June 2012; pp. 87–92.
16. Zellweger, H.P. Tree Visualizations in Structured Data Recursively Defined by the Aleph Data Relation. In Proceedings of the 20th International Conference Information Visualisation, Lisbon, Portugal, 19–22 July 2016; pp. 21–26.
17. Zellweger, H.P. A Decision Tree Interface Based on Predicate Calculus. In Proceedings of the Information Visualisation (IV) 21st International Conference, London, UK, 11–14 July 2017; pp. 188–193.
18. Burgin, M.; Zellweger, P. Nested Named Sets in Information Retrieval. In *Advances in Data Science and Information Engineering*; Springer: New York, NY, USA, 2021; pp. 451–467.
19. Ivanova, K. A survey of mathematical and informational foundations of the BigArM access method. *Int. J. Inf. Technol. Knowl.* **2015**, *9*, 272–299.

20. Ivanova, K. Storing and processing accounting information based on collect/report paradigm, International Journal. *Inf. Content Process.* **2015**, *2*, 103–114.
21. Tannenbaum, A. *Metadata Solutions*; Addison-Wesley: Reading, MA, USA, 2002.
22. Burgin, M.; Tandon, A. Naming and its Regularities in Distributed Environments. In Proceedings of the 2006 International Conference on Foundations of Computer Science, Las Vegas, NV, USA, 12–15 July 2010; pp. 10–16.
23. Adjie-Winoto, W.; Schwartz, E.; Balakrishnan, H.; Lilley, J. The design and implementation of an intentional naming system. *Oper. Syst. Rev.* **1999**, *34*, 186–201. [CrossRef]
24. Balakrishnan, H.; Lakshminarayanan, K.; Ratnasamy, S.; Shenker, S.; Stoica, I.; Walfish, M. A Layered Naming Architecture for the Internet. *ACM SIGCOMM Comput. Commun. Rev.* **2004**, *34*, 343–352. [CrossRef]
25. O'Toole, J.; Gifford, D.K. Names should mean what, not where, in EW 5. In Proceedings of the 5th Workshop on ACM SIGOPS European Workshop: Models and Paradigms for Distributed Systems Structuring, Mont Saint-Michel, France, 21–23 September 1992; pp. 1–5.
26. Alex, B.; Haddow, B.; Grover, C. Recognising nested named entities in biomedical text. In Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing, Prague, Czech Republic, 29 June 2007; pp. 65–72.
27. Chu, G.; Stuckey, P.J. Nested Constraint Programs. In *Principles and Practice of Constraint Programming (CP 2014)*; Lecture Notes in Computer Science, v. 8656; Springer: New York, NY, USA, 2014.
28. Finkel, J.R.; Manning, C.D. Nested named entity recognition. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, 6–7 August 2009; pp. 141–150.
29. Kawaguchi, A.; Lieuwen, D.; Mumick, I.; Ross, K. Implementing incremental view maintenance in nested data models. In *Database Programming Languages (DBPL 1997)*; Lecture Notes in Computer Science, v. 1369; Springer: Berlin/Heidelberg, Germany, 1998; pp. 202–221.
30. Burgin, M. Elements of the Theory of Nested Named Sets. *Theory Appl. Math. Comput. Sci.* **2020**, *10*, 46–70.
31. Dalla Chiara, M.L.; Toraldo di Francia, G. Individuals, Kinds and Names in Physics. In *Bridging the Gap: Philosophy, Mathematics, Physics*; Corsi, G., Ed.; Kluwer Academic Publishers: The Alfven am Rhein, The Netherlands, 1993; pp. 261–283.
32. Barr, M.; Wells, C. *Toposes, Triples, and Theories, Grundlehren Der Math*; Wissenschaften, v. 278; Springer: Berlin/Heidelberg, Germany, 1985.
33. Wadler, P. Comprehending monads. *Math. Struct. Comput. Sci.* **1992**, *2*, 461–493. [CrossRef]
34. Bourbaki, N. *Theorie Des Ensembles*; Hermann: Paris, France, 1960.
35. Burgin, M. *Super-Recursive Algorithms*; Springer: New York, NY, USA, 2005.

proceedings

MDPI

*Proceedings*

# Knowledge Processing as Structure Transformation †

**Mark Burgin [1], Rao Mikkilineni [2,*] and Samir Mittal [3]**

[1] University of California, Los Angeles (UCLA), 700 Tiverton Ave Los Angeles, CA 90095, USA; mburgin@math.ucla.edu
[2] C3DNA, 7533 Kingsbury Ct., Cupertino, CA 95014, USA
[3] SCUTI AI, 271 Ely Place Palo Alto, CA 94306, USA; samir@scutiai.com
* Correspondence: rao@c3dna.com; Tel.: +1-408-406-7639
† Presented at the IS4SI 2017 Summit DIGITALISATION FOR A SUSTAINABLE SOCIETY, Gothenburg, Sweden, 12–16 June 2017.

Published: 9 June 2017

**Abstract:** As big enterprises and consumers communicate, collaborate and conduct commerce almost at the speed of light using voice, data and video, information explosion (a term first used in 1941, according to the Oxford English Dictionary) has created a need for its accumulation, processing and integration to create "knowledge." Knowledge processing, in turn, allows us to use the information to make strategic decisions and improve the efficiency of the processes involved. Therefore, knowledge processing systems, their theory and practice are receiving renewed focus. These systems include processes and activities such as cognition, knowledge production, learning, knowledge acquisition, reasoning, management and application. In this paper we discuss how knowledge processing can be viewed as manipulation of various knowledge structures and their transformation. We argue that efficient organization of knowledge processing has to be based on structure transformations of data represented in a symbolic form.

**Keywords:** knowledge structures; cognition; knowledge transformation; named data networks

## 1. Introduction

Knowledge processing is not only a central problem of Artificial Intelligence (AI) but also an urgent dilemma of contemporary society and especially, in business and industry.

As Bray [1] writes, organizational knowledge processes deal with the creation, distribution, use and exchange of knowledge for purposes of value creation. They involve managing the intellectual capital of organizations. These processes are best understood with the ecology and ecosystem metaphors. Performative organizational knowledge is a knowledge ecology—a system consisting of many sources, venues, forms and species of knowledge agents in a symbiotic relationship of productive exchange and value creation.

In order to process knowledge and derive value from it, we need to understand the relationship between data, information and knowledge and create knowledge structures. According to Mark Burgin [2] knowledge is derived from information obtained from data resulting from study, experience or instruction. Information gives knowledge of a specific event or situation; or information provides intelligence to make decisions and take action. In this paper we present various structure transformation techniques for knowledge processing.

## 2. Knowledge Processing

It is possible to understand knowledge processing in three different ways:

1. Transformation of data into knowledge
2. Changing the form of knowledge representation

## 3. Deriving new knowledge from a given knowledge

However, whatever understanding we take, it is evident that knowledge processing involves manipulation with knowledge structures. Thus, to be able to efficiently process knowledge using computers and networks, it is essential to know and properly use knowledge structures. The most basic knowledge structures are described in the *synthetic theory of knowledge* presented in [2]. This theory shows that there are several levels or types of data, which by an enrichment (enhancement) process become knowledge.

The first type is *raw* or *un-interpreted data*. Their first-order structure is represented by Picture (1).

$$U \xrightarrow{q} A \qquad (1)$$

In this Picture, U consists of some objects, q is a relation between U and A, while A denotes: (a) attributes of these objects in the case of descriptive data, (b) representations of these objects in the case of representational data, and (c) operations, action and processes related to these objects in the case of operational data. For instance, all three types of data are used in the object-oriented programming for object description. Namely, object-oriented programming (OOP) is a programming paradigm based on the concept of abstract objects represented by data structures that include attributes in the form of descriptive data, characteristics in the form of representational data and methods in the form of operational data.

Raw data correspond to the substantial component of knowledge [2]. The difference is that in contrast to the substantial component of knowledge, raw data are not related to any definite property.

Having raw data, a person or a computer system can transform them into knowledge by means of additional knowledge that this person or computer system already has.

The second type of data is *formally interpreted data*. They are related to the abstract property P, forming a named subset of the information component of a knowledge unit represented by Picture (1), and the first-order structure of this component is represented by Picture (2).

$$N \xrightarrow{p} L \qquad (2)$$

Here N consists of the names of objects and L is the set of values (the scale) of the property P = (N, p, L) on the names of objects that tentatively have these properties, while p is the relation that connects names of considered objects with values of the ascribed properties of these objects, i.e., *p* is the functional component (evaluation function) of the property P.

Formally interpreted data correspond to the symbolic component of knowledge [2]. The difference is that formally interpreted data are not necessarily included in a knowledge system.

The third type is *attributed data*. They are related not to the abstract property P but to the values of an intrinsic property, i.e., to the attribute A. The first-order structure of attributed data is represented by Picture (3).

$$A \xrightarrow{g} L \qquad (3)$$

The fourth type of data is *naming data*, the first-order structure of which is represented by Picture (4).

$$U \xrightarrow{f} N \qquad (4)$$

There are two more types of data, which are more enhanced and are closer to knowledge.

The fourth type is *object interpreted data*. Their first-order structure is represented by Picture (5).

$$\begin{tikzcd} & L \\ U \arrow[r, "f"] & N \arrow[u, "p"'] \end{tikzcd} \qquad (5)$$

The fifth type of data is *object attributed data*. Their first-order structure is represented by Picture (6).

$$\begin{tikzcd} A \arrow[r, "g"] & L \\ N \arrow[u, "q"] & \end{tikzcd} \qquad (6)$$

It is interesting to remark that the statement about the correspondence between linguistic constructions representing knowledge and things in the external world as a necessary component of knowledge, which makes it different from data, was discovered in [3] and then reiterated by Davis [4] and Burgin [5,6].

It is possible to treat all types of data as incomplete knowledge.

Formally, naming data are names of considered objects and correspond to the naming component of knowledge [2]. The difference is that naming data are not necessarily included in a knowledge system.

It is important to understand that data and knowledge can themselves be objects, which have names, as well as intrinsic and ascribed properties. In particular, when the domain *U* consists of (some kind of) data, then in this case, we come to *named data*, which play an important role in the recent ideas for the development of the Internet [7,8]. To understand what named data are and why they are so popular, we consider the schema of the data transfer on the Internet.

The contemporary Internet is based on the TCP/IP communication protocol. In it, the TCP (Transmission Control Protocol) part is performs separation of the file/message into packets on the source computer and reassembling the received packets at the destination, e.g., at the recipient computer. The IP (Internet Protocol) part handles the address of the destination computer so that each packet is routed (sent) to its proper destination.

In Named Data Networking (NDN) architecture for the future Internet, the transmitted packets of data carry data names rather than source or destination addresses. The developers of this architecture believe that this conceptually simple shift will have far-reaching implications for how people design, develop, deploy, and use networks and applications. The Named Data principle implies that a communication network should allow a user to focus on the data identified by their names he or she needs, rather than having to reference a specific, physical location where that data would be retrieved.

Actually, Internet packets of data are already named by destination addresses and the new approach suggests changing these names to the original data names (identifiers). It is assumed that

such a renaming brings potential for a wide range of benefits such as simpler configuration of network devices, building security into the network at the data level and content caching to reduce congestion and improve delivery speed. In addition, sustained growth in e-commerce, digital media, social networking, and smartphone applications has led to prevailing use of the Internet in the role of a distribution network. Utilization of a point-to-point communication protocol in distribution networks is complex and error-prone, while Named Data Networking better suits distribution environment.

In this context, named sets give a natural mathematical model for named data, which form the naming component of the knowledge quanta with data as their object or domain [9]. Consequently, named set theory provides powerful means for network algorithms and procedures in the form of various operations and correspondences [10].

Another example of naming data is named graphs, which are a key structure of the Semantic Web architecture [11]. In it, a set of Resource Description Framework (RDF) statements (a graph) are identified using a URI (Universal Resource Identifier), allowing derivation of descriptions of context, provenance information or other metadata. This shows that named graphs form an extension of the RDF data model giving additional evidence for importance of naming data in contemporary information technology.

Combining all previous Pictures together, we get the quantum knowledge structure or structure of a knowledge quantum.

$$\begin{tikzcd} A \arrow[r, "g"] & L \\ U \arrow[u, "q"] \arrow[r, "f"] & N \arrow[u, "p"'] \end{tikzcd} \qquad (7)$$

In Picture (7), U is the knowledge domain (knowledge object), A is an aspect of the domain (object) U, the symbol N, which denotes the name of U or a class of names of the objects from U (a name of the object U), and *L* is the set of values (the scale) of the property P = (N, p, L) on the names of objects.

There is a tendency to treat knowledge as data with ontology. This approach is represented by Picture (8).

$$\begin{tikzcd} \mathrm{Ont} \arrow[r, "g"] & \mathrm{DescOnt} \\ D \arrow[u, "q"] \arrow[r, "f"] & N \arrow[u, "p"'] \end{tikzcd} \qquad (8)$$

In Picture (8), D denotes raw data, Ont is an ontology or interpretation of the data D, the symbol N denotes the names of the elements from the raw data D, and DescOnt is a description of the ontology *Ont*, which provides understanding of the ontology Ont.

Thus, we see that efficient organization of knowledge processing has to be based on structure transformations of data represented in a symbolic form.

## 3. Conclusions

Explication of structural features of knowledge processing in this work is aimed at achieving highly efficient semantically based computing and networking in general and at the development of an advanced semantic web, in particular.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Bray, D. Organizational Dynamics of Technology-Based Innovation: Diversifying the Research Agenda. In *IFIP International Federation for Information Processing*; McMaster, T., Wastell, D., Femeley, E., DeGross, J., Eds.; Springer: Boston, MA, USA, 2007; Volume 235, pp. 457–462.
2. Burgin, M. *Theory of Knowledge: Structures and Processes*; World Scientific: New York, NY, USA; London, UK; Singapore, 2016.
3. Burgin, M. Knowledge in Intelligent Systems. In Proceedings of the Conference on Intelligent Management Systems, Varna, Bulgaria, 27 June–1 July 1989; pp. 281–286.
4. Davis, R.; Shrobe, H.; Szolovits, P. What is knowledge representation? *AI Mag*. **1993**, *14*, 17–33.
5. Burgin, M. Data, Information, and Knowledge. *Information***2004**, *7*, 47–57.
6. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
7. Jacobson, V.; Smetters, D.K.; Thornton, J.D.; Plass, M.F.; Briggs, N.H.; Braynard, R.L. Networking Named Content. *CACM* **2012**, *55*, 117–124.
8. Ntuli, N.; Han, S. Detecting router cache snooping in Named Data Networking.In Proceedings of the International Conference on ICT Convergence (ICTC), Jeju, Korea, 15–17 October 2012; pp. 714–718.
9. Burgin, M.; Tandon, A. Naming and its Regularities in Distributed Environments. In Proceedings of the 2006 International Conference on Foundations of Computer Science, Las Vegas, NV, USA, 26–29 June 2006; pp. 10–16.
10. Burgin, M. *Theory of Named Sets*; Mathematics Research Developments, Nova Science: New York, NY, USA, 2011
11. Carroll, J.J.; Bizer, C.; Hayes, P.; Stickler, P. Named graphs, provenance and trust. In Proceedings of the 14th International Conference on World Wide Web (WWW '05), Amsterdam, The Netherlands, 12–15 December 2005; pp. 613–622.

© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/)

philosophies

MDPI

Concept Paper

# Operators in Nature, Science, Technology, and Society: Mathematical, Logical, and Philosophical Issues

**Mark Burgin and Joseph Brenner**

1 Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA
2 International Center for Transdisciplinary Research, Paris, France, c/o Chemin du Collège 1, CH-1865 Les Diablerets, Switzerland; joe.brenner@bluewin.ch
* Correspondence: mburgin@math.ucla.edu

Received: 10 May 2017; Accepted: 16 August 2017; Published: 7 September 2017

**Abstract:** The concept of an *operator* is used in a variety of practical and theoretical areas. Operators, as both conceptual and physical entities, are found throughout the world as subsystems in nature, the human mind, and the manmade world. Operators, and what they operate, i.e., their substrates, targets, or operands, have a wide variety of forms, functions, and properties. Operators have explicit philosophical significance. On the one hand, they represent important ontological issues of reality. On the other hand, epistemological operators form the basic mechanism of cognition. At the same time, there is no unified theory of the nature and functions of operators. In this work, we elaborate a detailed analysis of operators, which range from the most abstract formal structures and symbols in mathematics and logic to real entities, human and machine, and are responsible for effecting changes at both the individual and collective human levels. Our goal is to find what is common in physical objects called operators and abstract mathematical structures, with the name operator providing foundations for building a unified but flexible theory of operators. The paper concludes with some reflections on functionalism and other philosophical aspects of the 'operation' of operators.

**Keywords:** function; information; logic; mathematics; ontology; epistemology; nature; operator; science; society; system; theory

## 1. Introduction: Operators and Change

Human cognition is based on creation of concepts that reflect studied phenomena. However, in this process, many important concepts in philosophy and science have proven extremely resistant to systematic explanation and exact definition. Examples are *change*, *consciousness*, *information* and *intelligence*. At the same time, an improved understanding of these concepts is of increasing importance for building a more satisfactory theory of individual and collective behavior, ultimately aimed at an efficient support of responsible attitudes toward economic growth and sustainable environment. Some of the problems with the above concepts can be addressed using process methodology or the interactivist approach [1]. A theory of information that emphasizes both qualitative and quantitative properties of information and provides efficient means for solving problems of the contemporary Information Society has been recently created [2]. Conceptual understanding is additionally provided by the research focused on the inter- or trans-disciplinary aspects of these concepts [3], as well as on the combination of concepts from neuro-science and philosophy [4].

An additional general notion that has received little rigorous attention, and yet has implications throughout the whole reality is that of an *operator*. To our knowledge, no comparative study of operators and their substrates or operands in different areas, "from minds to machines", has been made. Compartmentalized formalized concept definitions of operators are used in mathematics, logic,

programming languages, and linguistics, while in everyday language, informal notions refer to familiar activities in the domains of machines, medicine, organizations and social activity.

However, the broad intermediate domain of non-mathematical real phenomena in which a causal impact is exerted by a person or entity that performs an operation and is, accordingly, an operator deserves a comprehensive unified theory. In this paper, we develop in some detail the notion of such "*natural*" operators. We position them as the proximal causes of the real change in a framework that includes the well-accepted symbolic operators of mathematics and physics and show the interrelationships abound in nature, mind and human organization and artifacts.

The major objective of this paper is, accordingly, to provide a comprehensive classification and discussion of natural operators and operations in several practical and theoretical areas of current interest, including a new kind of logic called Logic in Reality [5]. This logic is an extension of logic from its standard linguistic domain to real, complex processes and natural operators are themselves considered as active processes involved in effecting change at biological and cognitive levels of reality. Our grounding of their properties in physics, as well as our conceptualization of the self-operation of human beings and their organizations, further supports the utility and roles of natural operators in general and their usefulness for information technology in particular. We further propose that the natural-social operator split provides a key criterion for determining what entities may be considered autonomous, morally responsible agents and thus, contributing to a solution of the principal-agent problem (agency dilemma), which is urgent for political science, sociology and economics. We develop detailed consideration of the critical relations between the concepts of operators and those of self-organization in humans. This leads naturally to the point that if the hermeneutics of philosophy also depend on real, cognitive processes, operators *ipso facto* need to be considered in a philosophical role as well.

Operators definitely have explicit philosophical significance. On the one hand, they represent important ontological issues of reality reflecting its dynamical nature when some objects act on other objects. Our universe exists in interaction of its components and elements, while interaction can be decomposed of separate actions involving the structure of an operator. On the other hand, epistemological operators form the basic mechanism of cognition. Many cognitive processes, such as search, selection or recognition, are performed by physical operators, which can be natural or artificial and are represented by abstract operators decomposable into simpler operators. In this context, any scientist, measuring device or computer is a physical operator.

*Contents of the Paper*

The next Section 2 contains basic definitions of operators, specifying basic classes of operators and their properties and providing a conceptual framework for the other parts of this study.

Section 3 defines and analyzes natural operators, including information as a natural operator and the natural operators of logic in reality, explicating their relations to epistemology, causality, and the philosophy of mind. It is important to understand that information, as an operator, acts both in the mind, which is a repository and processor of information, and in machines, especially highly organized machines, such as computers.

In Section 4, we further study mind and knowing, defining the psyche as a natural operator with special attention to the complex set of operators that comprise different kinds of self-operation, such as self-organization, self-regulation and self-management. In the last sub-section of Section 4, we study knowledge as a natural operator.

Section 5 discusses the nature and role of operators in technology so that the differences, important for a complete theory, with natural operators are explicitly stated.

Section 6 discusses human operators and their intentionality paying critical attention to the on-going debate on the nature of agents vs. operators, suggesting possible new interpretations and approaches. Prospective work is suggested in both theoretical and practical directions in the areas of symbolic operators.

Our concluding Section 7 provides a brief discussion of operators from a perspective of the philosophy of science.

## 2. Basic Definitions and Classifications: Function and Type of Operators

The absence of a comprehensive discussion of operators has been due in part to the complexity of their differences in kind. The categorization we propose takes into account most if not all types of different operators, starting from a basic Definition 1, which identifies the most general concept of an operator. It allows us to categorize all existing operators by their essential characteristics.

**Definition 1.** *An **operator** is an object (system) that operates, i.e., performs operations on, some object, system or process.*

**Definition 2.** *An **operand** is an object, system or process operated by an operator.*

These definitions show that being an operator or an operand is a role and a characteristic of a system. One and the same system/object can be an operator in some situations and an operand in other situations, and an operator with respect to some systems and not an operator with respect to other systems. All operators are systems, but not all systems are operators since subsequent to their formation, some may exist in substantial isolation from their environment to all intents and purposes.

Definitions 1 and 2 express the fundamental dyadic relation between operators and their operands, which is actualized in the form of the *operator triad*:

$$\text{Operator} \xrightarrow{\text{Operation/function}} \text{Operand} \tag{1}$$

This diagram demonstrates that an operation is a component of an operator. In addition to operations, operators have instructions and/or material tools for performing operations. This peculiarity was discovered by programmers. They found that when they want computers to execute some operation, it is necessary to create a symbolic operator, which provided computer realization of the given operation. As a result, for example, all logical operations were transformed into logical operators. For instance, AND *operator* is a Boolean operator that returns a value of TRUE if both its operands are TRUE, and FALSE otherwise. The difference between an operator and an operation or a function is that the operator acts, while the function/operation is realized or performed.

Symbolic and natural operators function in a variety of areas: linguistic operators operate on languages; topological operators operate in and on topological spaces; standard logical operators operate in standard logic; network operators operate in networks; program operators operate on data processed by computers and other information processing systems; bus and plane operators operate buses and planes respectively, and so on.

To put some order into this diversity of operators, we have developed the following framework of operator classifications. On the first level of this framework, operators are primarily classified by three basic parameters: *form*, *operational medium* and *target*.

<u>The form-oriented classification</u>:

**Definition 3.** *A **symbolic operator** is an operator that has a symbolic form.*

**Definition 4.** *A **material operator** is an operator that has a material form.*

**Definition 5.** *A **mental operator** is an operator that is a part (element) of mentality.*

<u>The medium-oriented classification</u>:

**Definition 6.** *A **social operator** is an operator that works (functions) in society.*

**Definition 7.** *A **nature operator** is an operator that works (functions) in nature.*

**Definition 8.** *A **technology operator** is an operator that works (functions) in an artificial world created by people, which includes technology and has been created by technology.*

The target-oriented classification:

**Definition 9.** *A **socialized operator** is an operator that works with/on social structures.*

**Definition 10.** *A **symbolized operator** is an operator that works with/on symbols (symbolic structures).*

**Definition 11.** *A **naturalized operator** is an operator that works/on with natural objects (systems).*

Note that it is possible that an operator has different medium and target types. For instance, a social operator can and often does work with symbols, e.g., a writer, and thus, is a symbolized operator. Software systems are technology operators, which work with symbols and thus, are also symbolized operators. Besides, the same system, e.g., an individual, can work in society as well as in nature and machines. This means that this system acquires the roles of a social and nature operator.

At a second level of classification, the basic parameters are *dynamics, origin* (*emergence*), and *function.*

The dynamic classification:

**Definition 12.** *A **system operator** is a system that has both a static and dynamic structures or/and operates systems.*

**Definition 13.** *A **function operator** is a function and/or an operator that operates functions.*

**Definition 14.** *A **process operator** is a process and/or an operator that operates processes.*

The existential classification:

**Definition 15.** *A **natural operator** is a natural object, i.e., an object that emerged in nature and functions as an operator.*

**Definition 16.** *An **artificial operator** is an artificial object, i.e., an object that is created by people or other living beings and functions as an operator.*

**Definition 17.** *A **hybrid operator** is partially natural and partially artificial.*

The relation between a natural operator and a nature operator, Definition 7, is complex and requires further study.

The function-oriented classification:

**Definition 18.** *A **cognitive operator** is an operator the goal of which is knowledge acquisition.*

**Definition 19.** *A **search operator** is an operator the goal of which is finding some object or information.*

**Definition 20.** *A **construction operator** is an operator the goal of which is building some system.*

Note that these categories are not independent. In fact, their key categorial feature is their *Non-Separability*, for which LIR provides a logical basis.

For a study of mental and machine operators, it is practical to take into account a number of important quantitative and qualitative properties of operators in addition to their basic classifications. The set of properties considered here is not intended to provide a complete scientific description of operators, but to serve to highlight general principles applicable to the entire category of operators.

For convenience, we have separated considered properties into three essentially philosophical categories, as follows:

- Intrinsic Properties
  - *Complexity* reflects how much resources are required
  - *Power* reflects what the operator can do
  - *Capabilities* reflect what means of actions are accessible (used) by the operator
- Relational Properties
  - *Relation* to the operated system (operand)
  - *Attitude* to the operated system (for human operators)
  - *Conditions and Restrictions* on operation of the operator
- Pragmatic Properties
  - *Tentative* (*theoretical*) *costs* of operation (individual *human cost, moral cost* and *reputation cost*)
  - *Actual results (benefits, profit)* from operation
  - *Productivity* reflects how well the operator functions

This overall approach establishes that operators are characterized by their single or multiple classificatory definitions and their properties. For example, human beings are *material*, *natural*, *naturalized* and *socialized* operators. When reference is made to one of these definitions in the text, it will be in the bold italicized font, as here. The indicated intrinsic, relational and pragmatic properties and its relevant disciplinary methodology further characterize operators. In the remainder of this paper, however, we have limited the discussion of the sets of properties to a few key examples.

## 3. *Natural* Operators

### *3.1. The Ubiquity of Natural Operators*

The original concept of an *operator*, such as the differential operators $\mathrm{d}/\mathrm{d}x$, $\partial/\partial x$, $\partial/\partial y$, and $\partial/\partial z$, Laplacian $L$, and Hamiltonian $H$, has been one of the most efficient and developed tools of theoretical physics. Even the basic arithmetic operations, such as + and $-$, are also operators as it has become clear with the advent of computer programming. Such symbolic, mostly mathematical, operators are *natural* operators that are studied by physics and describe physical concepts, reflecting their properties. For instance, mathematical operators are used as models of the term *observable*, which has become the standard concept in quantum mechanics as the counterpart of the term *physical quantity* or *measurable quantity* in classical physics. This term originated from the term *observable quantity* (*beobachtbare Grösse*) defined by Heisenberg in his groundbreaking work on matrix mechanics [6]. His goal was to specify physical quantities by means of an operational definition. As a result, in contemporary quantum mechanics, the observables of a physical system are represented by self-adjoint operators acting in the Hilbert space $H$ associated with the system. In general, operators play a fundamental role in physics. For instance, the physics of quantum mechanics was reduced to the mathematics of operators on Hilbert spaces [7,8].

In general, an observable is a physical object or a property of such an object that some system (in a more restricted sense, people) can observe and/or measure. Thus, in the later development of quantum theory, several authors suggested the generalized representation of observables as positive

operator measures [9–11]). This concept advanced the mathematical coherence and conceptual clarity at first of quantum mechanics and then of quantum field theory. From physics, mathematical operators were extended to quantum chemistry where they also have been successfully used to model different processes.

However, taking the Definition 15 of a *natural* operator (cf. Section 2), we see that the entire real world is full of operators. In essence, any physical or biological object or system that interacts with and impacts other physical and/or biological objects or systems is a *natural* operator. Indeed, *all* systems in nature interact with their environment, but our operator approach enables the characteristics of the interaction to appear more formally. Thus, a system $A$, which interacts with a system $B$, as a rule, has some impact on the system $B$—either the state of $B$ is changed or functioning/behavior of $B$ is changed or the whole system $B$ is changed. Consequently, according to Definition 1, the system $A$ is a *natural* operator, while according to Definition 2, the system $B$ is a *natural* operand. The multitude of *natural* operators is studied by natural sciences. As a result, *natural* sciences could be called a study of operators!

### *3.2. Information as a Natural Operator*

In the General Theory of Information [2] information is characterized by a system of principles and the General Transformation Principle 2, describes the essence of *information* in a broad sense as the potential (capacity) of things, both *material* and abstract, to cause changes (transform) other things. When this capacity (potential) is actualized, it becomes a *nature* or *technology operator* (cf. Definitions 7 and 8), which acts on different systems. Thus, it is reasonable to distinguish *potentialized and actualized* components of information, whose evolution follows the pattern of Logic in Reality, as discussed in Section 3.3.

It is necessary to stress that information, as an operator, acts both in the mind, which is a repository and processor of information, and in machines, especially highly organized machines, such as computers. Some of the further aspects of information that justify its designation as a *natural* operator emerge from theories that give a fundamental role to information in existence. For instance, Scarrott [12] asserts that every living organism, its vital organs and its cells are organized systems bonded by information, which operates organisms, organs and cells. In a similar way, Reading [13] writes, "one of the main impediments to understanding the concept of information is that the term is used to describe a number of disparate things, including a property of organized matter . . . " He considers energy and information as the two fundamental causal agents, i.e., *natural* operators, acting in the natural world.

Information plays an important role in evolution, as in the elegant theory of evolution developed by Csanyi [14] and Kampis [15]. Burgin and Simon [16] also demonstrated that information has been and is the currently prevailing force for evolution both in nature and society. Smith and Szathmary [17] discuss evolutionary progress in terms of radical improvements in the representation of biological information. All these processes are initiated and controlled by information as a *natural* operator. More recently, Deacon [18] has developed a theory of dynamic levels of complex processes that ties together both evolution and information by the concept of *absence*.

The issue of the 'physicality' of information is the subject of intensive on-going debate (information as a "physical essence"). Crutchfield [19] treats information as "the primary physical entity from which probabilities can be derived". Landauer [20] stresses, information is inevitably physical. However, it is more reasonable not to claim that information itself is a physical essence but to suggest that people observe information only when it has a physical representation. Thus, all information in social organization and communities requires some physical form for its content to be transmitted.

Information exists in the form of *portions of information*. Informally, a portion of information is or can be considered (treated) as a separate entity. For instance, information in a word, in a sentence or in a book is a portion of information. Each such portion is an operator in its own right. Thus, we can conclude with Kaye [21]:

> "Information is not merely a necessary adjunct to personal, social and organizational functioning, a body of facts and knowledge to be applied to solutions of problems or to support actions. Rather it is a central and defining characteristic of all life forms, manifested in genetic transfer, in stimulus response mechanisms, in the communication of signals and messages and, in the case of humans, in the intelligent acquisition of understanding and wisdom".

In other words, *natural* information operators are pervasive in all complex systems.

We now come to another portrayal of *natural* operators, including informational ones, that emerges from the extension of formal logic to real systems made by Brenner [5]. According to this approach, the discussion of operators moves from the primarily theoretical domain toward the operational characteristics of the human mind.

One of the well-known examples of a *natural* operator is DNA, which stores genetic information in the linear sequence of nucleotides in DNA (deoxyribonucleic acid) and written as text in the alphabet of three base pair sequences (tri-nucleotides) called *codons*, while the *genome* is the entirety of an organism's hereditary information [22]. Usually, the term *genome* means genetic information stored on a complete set of nuclear DNA. Sometimes this term is also applied to genetic information stored within DNA of organelles. To discern these cases biologists use such terms as the *nuclear genome*, *mitochondrial genome* and *chloroplast genome*. DNA of the human genome is arranged into 24 distinct *chromosomes*—physically separate molecules. Each chromosome contains many *genes*, the basic physical and functional units of heredity. Genes are specific sequences of bases that encode instructions on how to make proteins. Thus, genomes are examples of information as *natural* operators.

In his book "The Touchstone of Life" [23], Loewenstein persuasively demonstrates that information is the foundation of life. To do this, he gives his own definition of information, the conventional definition of Hartley-Shannon information theory being inapplicable. According to Loewenstein, information, in its connotation in physics, is a *measure of order*—a universal measure applicable to any structure or system. It quantifies the instructions that are needed to produce a certain organization. "The pivotal role of DNA for all living beings made it clear that life as a phenomenon is based on biological structures and information they contain. Information encoded in DNA molecules controls the creation of complex informational carriers such as protein molecules, cells, organs, and complete organisms." As a result, genetic information plays the role of an operator for protein molecules, cells, organs, and complete organisms.

Another important example a *natural* operator is energy, which is information in a broad sense [2]. According to Smolin [24], the three-dimensional energetic world is the flow of information. In a similar way Stonier [25] asserts, structural and kinetic information is an intrinsic component of the universe, independently of whether any form of intelligence can perceive it or not. Reflecting this approach, Bekenstein [26] asserts that there is a growing trend in physics to define the physical world as being made of information itself. From this point of view, *natural* information operators are present in all *natural* systems. Even more radical point of view is expressed by Wheeler [27], who claims that every item of the physical world is information-theoretic in origin. In this view, all such information is composed of a multitude of information operators, e.g., information in an instruction is an information operator, a *system* or *function* operator (Definitions 12 or 13). Brenner [28], however, points out that views such as those of Wheeler and Bekenstein can lead to some misunderstandings about the correct ontological relation of priority between information and matter-energy. Usually it is supposed the latter being primitive in contrast to the former. However, the unified approach of the General Theory of Information explains that both phenomena are primitive but belong to the different spheres of reality [2]. Namely, matter-energy belongs to the ordinary physical world, while information dwells in the cognitive structural reality, whose instantiation in matter-energy, while real, is secondary to its functionality [29].

We know that the common usage of the word information does not imply such wide generalizations as the Ontological Principle 2 does. Thus, we need a more restricted theoretical

meaning because an adequate theory, whether of information or of anything else, should correspond to our commonsense notions of its content. This brings us to the idea of proper information, which is formalized by the Ontological Principle 2a in the General Theory of Information [2].

### *3.3. The Natural Process Operators of Logic in Reality (LIR)*

Logic is often defined as the theory of correct reasoning, where logic is understood as classical bivalent propositional and predicate logics or their modern multivalent, fuzzy, epistemic, temporal, modal, deontic or intuitionist versions [30]. Such logics are neither intended to describe nor have the capability of describing reality in nature or society. The applicability of logic and its symbolic operators and operations has been limited to language and mathematics. This is, however, based on a discretionary metalogical principle introduced by the scholastic followers of Aristotle and maintained by the linguistic turn of the 19th–20th centuries.

Writing (in French) in the middle of the last century, the Franco-Romanian philosopher Stéphane Lupasco (Bucharest, 1900–Paris, 1988) challenged the monolithic propositional, truth-functional character of logic and proposed an extension of logic to real process phenomena, based on the perceived dualities of matter-energy. In this new (old) way of "doing" logic, Lupasco essentially carried out a metalogical rejunction of logic with its original function as natural science [31]. Based on this work of Lupasco, Brenner [5] developed Logic in Reality (LIR), which is grounded in a particle/field view of the universe. At the same time, axioms and rules from LIR provide a framework for analyzing and explaining real world entities and processes, including information, at biological, cognitive and social levels of reality or complexity.

Details of LIR are provided in elsewhere (cf. [5,32]). Stated in a compressed form, the most important concepts of LIR are:

(1) every real complex process is accompanied, logically and functionally, by its opposite or contradiction, but only in the sense that when one element is (predominantly) present or actualized, the other is (predominantly) absent or potentialized, alternately and reciprocally, without either ever going to zero (the Axioms of Conditional Contradiction and Asymptoticity);
(2) the emergence of a new entity at a higher level of reality or complexity can take place at the point of equilibrium or maximum interaction or "counter-action" between the two (the Axiom of the *Included* Middle).

Together, these contradictional relations will be referred to as the Principle of Dynamic Opposition (PDO) of LIR . It can be roughly visualized in Figures 1 and 2.

| | | |
|---|---|---|
| Principal Directions: | *Potential* → *Actual* (Actualization) | *Actual* → *Potential* (Potentialization) |
| Entities: | *Identity* (Homogeneity) | *Diversity* (Heterogeneity) |

**Figure 1.** Process change: LIR non-contradiction.

| | |
|---|---|
| Principal Direction: | → *Semi-Actuality* and *Semi-Potentiality* |
| Entity: | *Emergent* T-*State* |

**Figure 2.** Process change: LIR contradiction (counteraction).

These figures show the relationship between LIR and the fundamental triads (named sets) that represent processes of change [33]: processes can move in three directions, two toward non-contradiction as their LIR identity or diversity increases and one toward (maximum) non-linguistic contradiction, a T-state from which a new entity can emerge. All of these are considered aspects of the *logic* in reality. (In standard logic, of course, the contradiction at the point of semi-actuality and semi-potentiality simply invalidates a proposition.) Unlike the Hegelian triad of thesis, antithesis and

synthesis, terms in which represent diachronic processes and form three fundamental triads, the LIR changes can be synchronic, with the initial elements and the emergent ones present at the same time, having different degrees of actuality and potentiality.

A major component of LIR is its categorial ontology in which the sole material category is Energy, and the most important formal category is Dynamic Opposition. From the LIR metaphysical standpoint, for real systems or phenomena or processes in which real dualities are instantiated, their terms are *not* separated or separable! Real complex phenomena display a contradictory relation to or interaction between themselves and their opposites or contradictions. On the other hand, there are many phenomena in which such interactions are not present, and they, and the simple changes in which they are involved can be described by classical, binary logic or its modern versions.

Therefore, LIR approaches in a new way the unavoidable cognitive problems that emerge from the classical philosophical dichotomies, such as *appearance* and *reality*, as well as the complementary concepts of *space*, *time* and *causality*, which are categories with separable categorial features, including, for example, final and effective causes. Non-Separability underlies a quantity of metaphysical and phenomenal dualities of reality, such as *determinism* and *indeterminism* (see below), *subject* and *object*, *continuity* and *discreteness*, *internal* and *external*, and *simultaneity* and *succession*. This is a 'vital' concept: to consider process elements that are contradictorially linked as separable is a form of a category error. The claim is that Non-Separability exists on the macroscopic and on the quantum levels, providing a principle of organization or structure in macroscopic phenomena that has been neglected in science and philosophy.

The *function* (Definition 13) and *process* (Definition 14) information operators in the General Theory of Information [2] provide the basis for a more formal characterization of the calculus developed by Lupasco and outlined in [5,32]. The connectives, that is, what is usually defined as the symbolic logical operators of implication, conjunction and disjunction, all correspond in LIR to real operators on real elements in the evolution of real dynamic processes. Accordingly, these operators are, also, subject to being actualized, potentialized or in a T-state. They operate not on theoretical states-of-affairs or propositions, considered as the abstract meaning of statements, but on events, processes and properties, where properties also have the character of processes.

The key concept is that the LIR operators themselves must be considered as processes, subject to the same logical rules, fundamental postulates and formalisms as other real and hence, natural processes. This answers a potential objection that the operations themselves would imply or lead to rigorous non-contradiction. Real processes are, accordingly, seen as constituted by series of series of series, etc., of alternating actualizations and potentializations. These series are not finite, however, in reality, processes do stop, and they are thus not infinite. Following Lupasco, we use the term transfinite for these series or chains, which are called ortho- or para-dialectics. The reader is referred to [5] for details of this applicable non-standard calculus.

One of the basic areas of application of these *natural* operators is, of course, language. However, the issues and relations addressed are much more complex than by standard linguistic operators. Ghils [34] has shown, for example, that the spatio-temporal dialectics in the linguistic theory of Roman Jakobson [35] is best described by the movement between actual and potential, using the corresponding operators as expressed by the Lupasco (LIR) calculus.

The *natural* operators of Logic in Reality are extremely complex, being both *symbolic*, *material* and *mental*, but also in part *symbolized, naturalized* and *social*, since implication, conjunction and disjunction obviously also function within social systems. As a final remark, the same picture applied to conjunction and disjunction as opposites provides the basis for a non-classical set theory, in which there is no absolute separation between sets and their members. According to de Morgan duality in classical logic, conjunction and disjunction are not *independent*, in the sense that a complementation operator takes any proposition to a similar one with the negative and operation inversed. This duality, however, still refers to a relation between abstract entities.

### *3.4. Operators and Causality*

The discussion of the nature of operators allows us to explicate their relations to the notion of causality or cause and effect. It is easy to accept that all operators are efficient causes. However, are all causes operators? Further, what is the causal nature of the operator-operand interaction?

The key conceptions in standard theories are those of Prigogine [36] of dissipative systems far from equilibrium, continued by Salthe among many others. The thermodynamic view focuses upon a final causality that operates universally. A functional separation is maintained between cause and effect, reifying them as entities separate from the property-processes they are supposed to operate on. For example, the intervention I must completely change any causal relationships between X and *its* prior causes. Nevertheless, one comment of Woodward [37] suggests some underlying common intuition. He suggests that philosophers do tend to think of causes as properties or events, but that it is possible to move back and forth between such talk and a representation in terms of variables. When there is no well-defined notion of change or variation in value, almost any standard theory, e.g., of manipulability, will not see genuine cause, but some form of epiphenomenality.

By localizing the origin of action in the *potential aspects* of phenomenal processes, which in the LIR view are intrinsic properties of all operators, our approach cuts through the debate on whether causation by omission, absence and prevention are cases of causation or not. They are. This line of argument also applies to the artificial distinction between natural and causal change, as well as internal and external change. LIR thus supports and explicates Fair's transference theory according to which, as discussed by Dowe [38] causation is a transfer of energy and/or momentum although causation by absence does not include any transfer of energy or momentum. Besides, in the context of the general theory of information, causation necessarily is a transfer of information [2].

Further detailed discussion of the subject of causality, (which has been called the "black hole" of philosophy), is beyond the scope of this paper. We believe that the major source of difficulties with standard views of the energetic aspects of cause, and the relation between cause and effect, have been due to the use of the conventional mutually exclusive categories of standard logic and category theory. In real complex situations, the reciprocal relations of feedback from operand to operator are of equivalent importance to the original *relational* property. Any human operator where the operation involves intentionality, operating as a cause, will be accompanied by a minority contradictorial tendency either *not* to operate or to cause the *opposite* effect. Logic in Reality permits a formalization of the parallel chains of causality [5]. Operation and cause are, accordingly, equivalent descriptions the choice of which will be defined by the specific focus of the subject of analysis.

## 4. *Psyche* as a *Natural* Operator

### *4.1. An Approach to a Theory of Mind*

Most of the difficulties in the vast variety of theories of mind and knowledge have been in finding the essence of "what it is" to be an entity with conscious awareness and causal efficacy. The concept of *natural* operators as a description of the operation of the LIR principle of dynamic opposition provides access to a new epistemology in which the classical entities of knower, knowing and known (knowledge) are all seen as processes in dialectic or contradictorial interaction.

In this Section, we are concerned with both individual mental processes and their totality, exposing their nature as operators, and choosing the term *psyche* (the Greek word for mind and soul) following Jung who used it for this totality. There is a striking similarity between his foundational view of opposition as an inherent principle of human nature [39] and dynamic opposition in LIR. As Jung wrote,

> "A psychological theory, if it is to be more than a technical makeshift, must base itself on the principle of opposition . . . There is no balance, no system of self-regulation, without opposition. The psyche is just such a self-regulating system."

Jung described the forces at work in terms of energies in language that is easy to compare with the LIR grounding in energy, and the *need* for opposition is clear from his emphasis that "there is no energy unless there is a tension of opposites".

We first note that members of *all* classifications of operators are instantiated in the psyche, providing a kind of rough measure of its existence as the most complex, natural object in the universe. To the suggestion that society, as an entity composed of multiple minds, is more complex, our answer is that many of the individual operations of society are less complex, involving fewer types of operators. Society as a collective entity or system (or 'being' in the sense of Minati and Pessa [40] instantiates emergent simplicity. In addition, the *psyche* is one of the basic concepts of depth psychology, comprising the forces in an individual that influence thought, behavior and the whole personality.

Freud [41] described the structure of the psyche as composed of three components:

- The *Id*, which represents the instinctual drives of an individual and functions mostly unconscious.
- The *Super-Ego*, which represents a person's implicit knowledge about social environment, including internalization of social norms, morality and other standards, functioning mostly unconscious.
- The *Ego*, which represents a person's conscience and serves to integrate the drives of the Id with the prohibitions of the Super-Ego.

In essence, Ego connects Id and Super-Ego, forming the following Fundamental Triad [33,42].

$$\text{Id} \xrightarrow{\text{Ego}} \text{Super-Ego} \tag{2}$$

Thus, in this context, the psyche controls an individual's functioning, behavior and personality. Consequently, psyche is a *natural* operator, with a compound structure built up from other simpler operators, forming a sophisticated hierarchy. Our intention is not to argue here for the validity of this descriptive system but simply to provide an interpretation in our "operator" language.

It is important to note, however, that the diagrams and discussions of the Fundamental Triad in this paper are univocal, without reference to bi-univocal relations, recursion and feedback involving operator and operand. The structure of such interactions is critical for any reasonable models of the mind, memory and complex social phenomena such as the economy. For discussions of these aspects, the reader is referred to Lupasco [43], Burgin [33,44] and Leydesdorff [45].

To further apply logical operations as dynamic processes, outlined in the previous sub-Section, to the human mental system, we start by looking at the dialectics of afferent and efferent systems in perception. Prior to excitation—a *natural* physical/biological operator—by internal or external stimuli, we assume that the afferent system is in a state of potentiality, maintained by the antagonistic actualization of the polarization or electrostatic equilibrium. Excitation results in a new actualization, potentializing the ionic equilibrium, the reception of an equivalent to heterogeneity of sensations. The new equilibrium state of perception appears, in its homogeneity, as something objective, exterior, an identity of which one can have 'knowledge', while sensations, although really belonging to the external world, *appear* interior to the senses and more subjective. The dialectics established in and by the afferent process is between the conscious mind of (or as) the 'knower', actualizing a series of energetic heterogeneities, and the 'known' displaced to the exterior in the potentiality of energetic homogeneity. This conception could be called 'pan-energetics', but it is not a pan-psychism, the mind appears as an aspect of the structuring and operation of energy.

The difference between actualizations that potentialize and potentializations that actualize is not continuous, and the pauses in the process, in the T-state, are what can be considered states of control. These constitute the dialectic of the psyche, which *becomes* what is generally called consciousness. There is thus in the LIR theory no 'consciousness of', no reification or objectification, only *that* which occupies the conscious mind, *that* which is potentiality itself, what is commonly called consciousness, with the capacity for causal change that justify its characterization as a *natural* operator. As a result,

relations between the knower, knowing and known become explicitly processual. Our approach to a theory of mind is discussed further below in Section 6.2.

### *4.2. The LIR Theory of Knowledge*

The LIR approach to knowledge purposefully includes it in reality. For instance, knowledge in a biological organism originates in, and can never be totally separated from, the set of processes that constitute its existence, including an organism's systems of perception and action. All these processes instantiate both reciprocal degrees of actuality and potentiality and different tendencies toward identity and diversity. The objects subject to these systems relations are the knower and a knowable or a known, as well as the exterior and an interior reality. The essential hypothesis is that the knower does not and cannot know himself in the process of knowing insofar as he/she knows or is in a process of knowing. Using a Leibnizian argument, we can show that for knowledge to exist, the knowable and the known must be neither totally identical to nor totally different from the knower. This is similar to the fundamental *Dynamic Knowledge Triad*, an important case of the Fundamental Triad described by Burgin [33,44] where the structure (4) is the actual form and the structure (5) is a potential form of the structure (3). In LIR, knower, knowing and known all instantiate both actuality and potentiality.

$$\text{Knower} \xrightarrow{\text{Knowing}} \text{Known/Knowable} \tag{3}$$

$$\text{Knower} \xrightarrow{\text{Knowing}} \text{Knower/Known} \tag{4}$$

$$\text{Knower} \xrightarrow{\text{Knowing}} \text{Knowable} \tag{5}$$

We assume, following the fundamental postulate of LIR, that existence and non-existence and the knowable and known in which they manifest themselves are dualities with contradictory terms, for one to be (predominantly) actual, the other must be (predominantly) potential. Thus when we know or do something, we do not know (or pay attention to) our knowing of it. We can focus, by another cognitive act, on our knowing, which does not know itself as such, and so on.

The core of this theory is that we *know* only what is potential—what is conceptual and 'ideal', the inverse of that which is relegated to the unconscious and according to which we act. The actual is always impossible to know, due to the nature of knowing itself, a concept that corresponds to the general idea according to which one cannot act, contemplate, etc. and see oneself act, contemplate, etc. at the same time. The known is neither totally identical nor totally other than the knower. The dualities are contradictory as one of them is (mostly) actual, while the other must be (mostly) potential. Although the roles of these dualities are changing, the two are not both fully actual at once, but as one is primarily actualized, the other is primarily potentialized.

In the present theory, the known is an element or entity or process that is contrary and contradictory to the knower. Together, they constitute antagonistic dynamisms in which alternating actualization and potentialization converts known to knower and *vice versa.* A known becomes a knower since 'knowledge' is not the knowledge that a knower 'has' but is a *process* that is active in the knower. As a part of an individual's total mental structure, this process, in a non-metaphorical sense, 'knows' or becomes a knower and accordingly, like knowing itself, is a *natural* operator as a consequence of the dialectics of contradiction.

We believe that the contribution of Lupasco, relating the terms used in philosophy since antiquity of knower and known to reality by focusing on *knowing* as a physical process is a major one that deserves more attention. The further discussion of the operator aspects of this process in the remainder of this paper, especially Section 6.3 on Intentionality and Section 7, should be read with this in mind. We can state already that our operators approach goes beyond any narrow functionalist conception of the mind.

## 5. Operators in Technology

### *5.1. Machines as Operators*

Usually machines are not treated as operators. They are operands (in our terminology) controlled by human operators. Here we extend our understanding of operators and this allows us to better characterize the role of machines in our society. Now, for example, a new tendency has arisen when machines are considered as operators but only when they are able to perform complex operations, and at least for a time work independently of people and operate, e.g., control some other machine. For instance, a robot can be an operator of another machine but traditionally it is not treated as an operator. A robot that explores the surface of the moon or planets must be able to walk on rough terrain in a harsh environment, receive instructions from remote operators about where to go next, and reach those commanded goals autonomously.

At the same time, it is customary to say that computers operate data. So, according to Definition 1, computers are operators. It may be less habitual but still acceptable to say that programs operate computers, but to hear that data operate computers seems rather strange. However, any manufacturing machine, e.g., loom or lathe, operates material from which the product of this machine is produced. Thus, this machine is an operator according to Definition 1.

This is especially true for robots, which become more and more abound. Industrial robots are found in a variety of locations including the automobile and manufacturing industries. Robots cut and shape fabricated parts, assemble machinery and inspect manufactured parts and perform many other manufacturing tasks. Outside the manufacturing world robots can be found in hazardous duty service, CAD/CAM design and prototyping, maintenance jobs, fighting fires, medical applications, military and agricultural operations and so on.

Some machines produce nothing, e.g., a car or a plane. So, if we base our judgment on Definition 1, a car or plane is not an operator. However, a car, as well as a plane, bus, train or ship, operates on things and people by taking them from one place to another. Nevertheless, there is an essential difference between humans and machines in their role of operators. What is considered unique in *humans* functioning as operators is the intentional relation between themselves and the object, tool or machine, which they operate or use as agents. In productive operation, there is thus an intentional relation between operators and the objects they create. We will therefore look briefly at the issue of intentionality in machines. In Section 6.2, we consider intentionality in humans.

### *5.2. Intentionality in Machines?*

Intentionality is an important property of human beings as operators, so, it is important to understand whether machines as operators also have intentionality. Some researchers tend to ascribe intentionality to such machines as computers, especially, when they have sophisticated software, e.g., softbots and other artificial intelligence programs. Other researchers completely deny a possibility for a machine to have intentionality. It is possible to suggest that such contrasting views are caused by a simplified understanding of intentionality, which exists now. Intentionality is considered as a binary property, which means that a system ether has intentionality or does not have —and nothing between these two possibilities.

To clarify the problem and eliminate this unnecessary restriction, we consider three kinds of intentionality:

- *Inserted* or *programmed intentionality*
- *Induced*, e.g., by persuasion, example or hypnosis, *intentionality*
- *Inherent intentionality*

Machines do not have inherent intentionality at all. For instance, a car goes only to the places directed by the driver. However, some machines, especially, those that can work autonomously, reach the first level of intentionality. There is a substantial debate in progress on whether machines can

reach higher levels of intentionality. The debate is not addressed here although some evidence for the positive answer is given below.

In the new direction in programming called programming by example, there is reason to assume that programming by example actually induces some kind of intentionality in the corresponding computer. If robots will be able to learn from examples, then it would be reasonable to consider that they acquire induced intentionality. Again, there have been endless speculations as to if and how artificial intelligence embodied in thinking machines might acquire all the traits of a human being, including intentionality. However, contemporary computers have only inserted/programmed intentionality. We will return to machine-related questions in Section 6.2 on Intentionality and Section 6.3 on recursive processes (Self-Control and Self-Regulation).

## 6. Human Operators

### *6.1. Basic Concepts*

In this Section, we study human operators, which epitomize almost all types of operators, especially, in the function-oriented classification. In fact, the idea that human beings are operators is so familiar that it rarely receives detailed discussions. It is a commonplace that "man as a "tool-making and tool-using" animal, but what is important from the standpoint of this paper is that humans, as biological systems, are *natural* operators in the sense of being evolving structures with causal power, *natural* and *naturalized*. At the same time, humans, as social units, are also *social* operators.

Obviously, the role of human operators is changing. Human operators are still required for the functioning and success of manufacturing systems, but advances in automation technology have caused fundamental changes in the way such systems operate. Functions of human operators have also evolved under the changing manufacturing environment. Human operators are currently performing more information processing, decision-making and control activities than ever before.

A brief historical remark: the classical view of man as a tool-maker, *homo faber*, indicated in Section 5, was developed by Bergson. Arendt proposed a subsequent social stage of man as *animal laborans*, man as worker [46]. Our conception of man as operator actually combines these two aspects and could be summarized by the term *homo operator*. This term was first used by Thomas Aquinas and more recent theologians with a rather negative connotation. (The term *homo operans* has also received a number of biological and theological definitions which will not be discussed here.) In our view, however, such a connotation is arbitrary. Ethical behavior or the lack of it supervenes on psychological and structural characteristics of an individual, some of which we discussed earlier, other than his or her functioning as an operator *per se.*

It is possible to consider all machines, from the simplest multiplier of force such as a lever or pulley to a computer, as operators. However, machines do nothing on their own but, like symbolic operators, have their origin in and are totally dependent on the *mental* and *natural* operators of human beings. The same remarks apply to recent man–machine hybrids. So, as Magnani [47] suggests, we exercise moral responsibility, giving them equivalent moral status. Such entities are, of course, paradigm examples of *hybrid* operators (Definition 17).

A simple model of a human operator is a person interacting with a machine in some kind of environment, for example, driving an automobile. The person and machine often are both modeled as information-processing devices, each with input component, central processor, and output component. The input component of a person consists of his/her sensors (e.g., eyes and ears) and the output component of a person consists of his/her effectors (e.g., hands and mouth). A computer and its ancillary devices has a similar operator structure. The environment can be characterized physically (e.g., vibration, noise, zero-gravity), cognitively (e.g., time pressure, uncertainty, risk), and/or organizationally (e.g., organizational structure, job design). This provides a convenient way for organizing some of the major concerns of human engineering: the selection and design of machine

displays and controls; the layout and design of workplaces; design for maintainability; and the design of the work environment.

No matter how important it may be to match an individual operator to a machine, some of the most challenging and complex human problems arise in the design of large man-machine systems and in the integration of human operators into these systems. Examples of such large systems are a modern jet airliner, an automated post office, an industrial plant, a nuclear submarine, and a space vehicle launch and recovery system.

Summarizing from the list of properties, and in line with the ability of LIR to explicate their relation, we point to the junction and Non-Separability, in human beings, of the internal properties of our operations: intelligence, capacities, and skills, each of which has its own enormous literature, and what is operated upon, the external aspects as operands.

In the case of human beings, it is useful to differentiate between operation and intentional use, as well as between tool and machine. When spectators watch a movie in a theater, they use this movie but do not operate it. At the same time, operation can take place unintentionally. In such situations, it is possible to operate something but not to use it. For instance, when an individual destroys something, she does not use that thing.

There is also an essential difference between tools and machines. In philosophy, the concepts of machines and mechanisms refer to essences of things, while *being a tool* is a role of an object. For instance, a car is a machine whether it is used or not. It becomes a tool only when it is used for achieving some goal. There is also a discussion whether a human being is a machine in this sense or not when instrumentalized as a tool by and for another individual or a group, becoming an operand in the narrowest sense, that is, with the destruction of ethical reciprocity. As humans are created by nature as biological objects, it is more relevant to assume that they are not machines. This once more demonstrates the difference between tools and machines.

Tools in the form of physical machines are extensions of human physical and mental capacities. We nevertheless can only present a few key examples from the entire range of human activities, differentiated by *attitude and complexity.* One extreme is the (literally mindless!) repetitive operation of a machine epitomized by Charlie Chaplin in the film *Modern Times.* At the other extreme is the surgeon executing delicate operations that are now possible only through human-machine symbiosis.

René Thom, in his catastrophe theory, mathematically describes the process of using a tool or weapon. For instance, the form of an axe or projectile with a beveled edge, for example, is imagined by the maker as appropriate for causing a swallow-tail catastrophe (one of the seven basic types) in the skull of the animal or enemy. "The mental vision of the catastrophe to be provoked in the adversary creates a secondary field, that of the fabrication of the axe [48]."

In LIR terms, the actual mental image process and the potential external effect as potential are dialectically connected. In this context, Lupasco [49] discussed the operation of tools and machines, constructions of human beings and extensions of their capacities, in terms of a dialectic alternation between induction and deduction in science. He wrote that "after operating (sic) inductively, bringing to light theoretical phenomenological identities, it (physics) becomes deductive, in order to verify the effective actualization of these identities . . . " Physics constructs machines and creates applied science such that this actualization implies that the objects fabricated should be rather considered fabricated *subjects*, since it is the agent, the operator-subject, the extension of the subject that is doing the actualizing of that theoretical identity. Deduction is the verification *operation* of inductive science.

What is unique in humans functioning as operators is the intentional relation between themselves and the object, e.g., tool or machine, they operate and use. In productive operation, there is also the intentional relation between operators and products they create. We will therefore look further at the structure of the intentional process started in Section 5.1 in relation to machines.

### *6.2. Intentionality and Non-Symbolic Representational Theories of the Human Mind*

The central problem for a philosophy of mind is to show how physical tokens in the form of the neuro-physiological processes occurring in the brain, can give rise to mental tokens that retain the properties of intentionality, "aboutness", individuality and some level of causal powers or functionality. The weak point in some current views [50] is that physical and mental tokens are treated as identical (identity theories of mind (ITM)). The LIR principle of opposition at all levels of perception, mental processing and action gives the logical and scientific basis for saying that something is the same and different, here physical and mental tokens, in dynamic opposition at the same time. At this stage, we simply state as a postulate that no theory of mind, philosophical or metaphysical, that is based on separability between physical and mental entities can provide adequate explanations of mental phenomena. This is not necessary in LIR, since the latter are described in terms of the contradictorial but physical process relations between the appropriate elements or entities.

Above we defined the human mind in its capacity for knowing, the psyche, as a *natural* operator. As we focus in this sub-Section on the totality of the human individual, we need to discuss another major aspect of human existence, namely, intentionality. Further explanation of intentionality and mental causation is critical, since the strong intuition of the completeness of causal exclusion requires that every physical state have a complete, that is, a necessary and sufficient physical cause. We should emphasize out that the following discussion refers only to what has been defined in the previous section as *inherent* intentionality.

In Brentano's conception, consciousness and mental phenomena are distinguished from physical by their intentionality. The characteristic property of the intentionality of mental states is their 'aboutness' or 'object-directedness', that is, consciousness of something. This involves critical issues regarding the object, the mental act of perception and the relation between them. A simple physicalist conception of, in particular, cause and the process involved in the perceptual acquisition of the object, is not sufficient to explain intentional relations. Our approach in terms of *symbolic* and *natural* operators provides a new perspective on the assumed presence and role of symbols in mental phenomena.

LIR demonstrates that there is something clearly missing or inadequate about the separation into mental and physical to begin with. The claim we make that offers a possible solution to the problem is that mental and physical phenomena are the same and different, and we will attempt to support this claim in the remainder of this Section.

LIR starts from the view that consciousness is an ensemble of dynamic processes, in which internal and external aspects cannot be totally separated from one another. Smith has criticized cognitive science for being reductionist specifically by its failure to preserve the properties of consciousness and intentionality, using the standard definition. "The phenomenological character of consciousness is not preserved, and Intentionality is not simply identical with the structure of a causal process . . . " [51]. LIR suggests the emergence of the phenomena of intentionality and qualia at the locus of the contradictorial interaction between the actualized subjective and potentialized objective parts of both incoming and outgoing stimuli.

In representationalist theories, internal entities of some sort stand for or correspond in some way to external processes and events. These mental representations explain or are explanatory devices for cognition in that they are, or correspond to intentional states, instances of intentionality considered as embodying the irreducible first-person properties that characterize consciousness, reasoning and qualia. The focus now shifts, therefore, to the representation of intentional states.

Without reproducing the entire debate, we note that in Fodor's intentional realism, a cognitive process is a causal sequence of intentional states. However, such states are not physical properties in the usual sense. To naturalize the initial intentional state, to which causal efficacy is ascribed, it is first transformed into a representational one, and this subsequent representational reconstruction is naturalized. The former is characterized as a symbol, similar to a linguistic symbol, but its intentionality is given by the original intentionality of the mental state. In other words, intentionality is identified with the semantic properties, including the capacity of physical causality, of a mental representation.

The language-of-thought hypothesis adds the concept that the system of mental symbols constructed above has a language-like structure, but the fundamental premise remains that a mental representation is a symbol, where a symbol is defined as something standing for something else, as noted.

The Fodorian construction implies that mental state and symbolic mental representation are identifiably separate entities, and in addition the second always exists in the relation between the state and the object. In other words, the idea that mental states could relate directly to objects without intermediaries is excluded. This depends on the idea that if a symbol makes a relation with something else possible, and is accordingly also an element, an intermediary one, all such relations must *ipso facto* be indirect. On the other hand, this is in contradiction with the accepted idea that some mental states are intuitive and accordingly in a direct relation with an intentional object.

Our preferred theory of intentionality is to involve non-symbolic mental representations, since in our view the above account of mental processes suffers from the need to introduce additional entities due to the lack of a principled categorial method of relating its critical concepts contradictorially. A mental phenomenon that is not something other than physical processes with emergent properties 'displays' its contradictorial origins in appearing to have symbolic and non-symbolic aspects, and being closer or farther from the center of attention at a particular time.

The difference of status between a representative content, in this picture, and a symbol can, first, be connected with the shift of attention that accompanies the movement from consciousness to consciousness of consciousness proposed above. Further, however, the underlying intuition is that representative content, although it is in a sense an intermediary in the intentional relation, is something other than a symbol because a symbol is an intended element as defined. The LIR conception of intentionality as a dynamic *mental* operator that avoids the reification of intentionality as a static posture of intent is still open, however, to the attack by anti-reductionists against all concepts of intentionality, namely, that they are unreliable. In the anti-reductionist, externalist theory of content, the thesis of intentionality need not be true. The reason given is that we do not know or are often wrong about the 'content' of intentional states. We will not argue these points here. In LIR, internal and external states are co-determining, and we do not require absolute truth for a workable phenomenological theory of intentionality.

In the LIR categorial ontology of mental process structures, intentionality thus appears as a kind of plan, composed of both actual and potential elements, more or less clear-cut or fuzzy (homogeneous or heterogeneous) in the contradictorial relation described above. Intentionality is thus best described, like knowledge, as a complex *natural* emergent operator, which emerges with respect to some primarily internal or external operand [52].

### *6.3. Self-Operation, Self-Operators, and Self-Organization*

Self-operation is a phenomenon that refers to the ability of human operators and organizations of humans to operate on themselves, that is, recursively. The term self-operation actually includes a number of processes that also take place at lower levels of reality and thus, self-operation is abundant in nature, society and technology. Among the many kinds of self-operation studied by researchers and used for practical purposes are self-modification, self-organization, self-regulation, self-management, self-replication, self-production, self-control, and self-programming. All of these processes in the broadest sense refer to properties of a system to change both its internal environment (structure) and external behavior (functioning). In general, all of the natural and social operators that execute these operations are *ipso facto* self-operators. In this paper, we will limit our discussion to self-organization, self-control and self-regulation.

#### 6.3.1. Self-Organization

We begin our discussion with the concept of self-organization in view of its importance for the understanding of all the reflexive processes, and because it illustrates the "operation" of the principles

of Logic in Reality [5]. The concept of self-organization was born in the 1960's in an attempt to establish a theory based on the standard logic of a system and its control.

The phenomenon of self-organization includes two sides (aspects): attributive, which is related to the inner structure of the system, and processual, which is related to the external behavior (functioning) of the system.

**Definition 21.** ***Self-organization*** *in a broad sense is the property of a system, which results in activity to better organize system's internal environment (structure) and external behavior (functioning).*

We thus include the above two aspects in our definition of self-organization: attributive and processual. In attributive self-organization, some systemic property is improved. In processual self-organization, some system functioning (behavior) becomes more organized.

There are two main approaches to self-organization in social systems. One of them is based on the model of a system composed of aggregated individuals where self-organization is generated by and results in the sum of the practices in the system induced by control, or self-control, in particular, and system logic. The second approach focuses on the practices of individuals deviating from the logic of a system, making the existing system fluctuate and transforming its structure. That is why, to exist, any organization has to self-organize itself.

As a result, self-organization is related to emergence of pattern or regular structures. Thus, the following definition is very often used:

**Definition 22.** ***Self-organization*** *is the process where a structure or pattern appears or grows in a system.*

The problem in all discussions of self-organization as a kind of self-operation is to establish why some systems have the capacity to self-organize or to express self-organization and others do not. In fact, on close inspection, all systems to which self-organization is attributed depend for their capacity on some set of either initial conditions or on-going input of energy and information that transforms the term "self"- into a contradiction, correctly, a "self-contradiction". This brings us to two extreme types of self-organization: independent and induced.

**Definition 23.** ***Independent self-organization*** *is a process where the organization (constraint, redundancy) of a system appears to spontaneously increase, i.e., without this increase being controlled by the environment or an encompassing or otherwise external system.*

A 100% independent self-organization is, in fact, an ideal operation which is never realized in the process of system functioning. It must always be accompanied, dialectically and functionally, by *non-self-organization* or, more exactly, hetero-organization [53].

**Definition 24.** ***Induced self-organization*** *or* ***hetero-organization*** *is a process where the organization (constraint, redundancy) of a system increases under the influence, e.g., pressure, persuasion or advice, from the environment or from an encompassing or otherwise external system.*

The problem to be resolved is then a proper definition of the interactions and differences between self- and hetero-organization. Indeed, an ordered structure capable of self-organization is nevertheless dependent on the input of external energy and information for it to form and persist [53]. It is not "spontaneously" generated within the system, even if it "spontaneously" forms. Once initiated, the process of self-organization does result in the creation of new entities. However, these require new (externally) available information for their further evolution. Hetero-organization refers, then, to a model for the delivery or introduction of this external information to the system.

The rigorous logical approach of LIR can be applied to the concept of self-organization. If one assumes a standard definition of a system, a self-organizing system is defined as distinguished by the

formation of some states or entities arising from the reciprocal or collective interactions (encounters) between its components, *quite independently of outside inputs.* LIR theory, however, states that the critical terms of 'self' and 'independent' involve question-begging assumptions, given the critical LIR categorial feature of Non-Separability discussed above.

Brenner [5] suggested that the critical step in the organization process is not spontaneous, in the sense of uncaused by outside agents, which the use of the particle "self-" without qualification implies. New organizational structures are the effective consequences of the potentialities residing in the components and/or introduced during the original constitution of the natural system or artificial experiment. This view supplements the discussion of self-organization in society proposed by Fuchs [54], in which the emphasis is on a dialectical, emergent transition from simple elements defined by one or two parameters to more complex process-like entities instantiating quality or meaning. Fuchs also suggested the need for a new functional "logic of self-organization" in another recent paper. The advantage of LIR for a theory of organized systems is that it provides, at least, a partial answer to the question of why some systems self-organize, or display autopoiësis, and others do not. LIR simply takes the theory of self-organization and grounds it in (at least) one lower level of reality, without the need for invoking any non-causal spontaneous processes.

We propose, accordingly, that self-organization is not, in and of itself, a 'self'-evident mode of system formation and change. All self-organizing systems also involve some degree of organization-by-external-agent, which is a case of hetero-organization and the two are dialectically related. Varela describes such a situation [55] when he states that coupled non-linear oscillators can give rise to kinds of self-organization that result in the emergence of neural structures from the component level. A local-global interdependence is necessary to understand the emergence. The components "attain relevance" through their relation with their global counterpart.

We further characterize self-organization, within the limitations discussed above as gradual or apparently spontaneous. Gradual self-organization is a basically a process of evolution where the effect of the environment is minimal, i.e., where the development of new, complex structures takes place primarily in and through the system itself, subject to control by the initial conditions. In many cases, it can be understood on the basis of the same variation and natural selection processes as other environmentally-driven evolutionary processes.

In processes of apparent spontaneous self-organization, fluctuations often operate as organizing forces when at the global level of a system, patterns emerge mostly from numerous interactions among the lower-level components of the system, for example, through the "order through fluctuations" mechanism discovered by Ilya Prigogine [36] in systems, which continuously export entropy in order to maintain their organization of dissipative structures.

As a definitive type of apparent self-organization, we may discuss self-organization in the restricted system sense [40,56].

**Definition 25.** ***Restricted self-organization*** *is the process where a structure or pattern appears in a system without a central authority or external element imposing it through planning and pressure, and is dependent on the prior potentialities of or initial conditions in or pertaining to the components of a system.*

Therefore, many processes of restricted self-organization are in part induced by other systems involving hetero-organization of a collective of different systems.

Self-organizing processes in the restricted sense, such as herd behavior, groupthink and others, are abound in sociology, economics, behavioral finance and anthropology, as well as in economy, politics, industry, medicine, and technology. Biological science studies the creation of structures by social animals, such as many mammals and social insects (bees, ants, and termites), flocking behavior (such as the formation of flocks by birds, schools of fish, etc.).

In economy, we can see such self-organization processes as: growth, competition, extinction of companies; functioning of financial markets and stock markets. In politics, we have revolutions, "self-dynamics", formation of public opinion and development of beliefs.

In social theory, the concept of self-organization was connected to self-referentiality and self-production by Luhmann [57], who treated the elements of a social system as self-producing communications when a communication produces further communications and hence a social system can reproduce itself as long as there is dynamic communication. At the same time, human beings are sensors in the environment of the social system. Based on these foundations Luhmann developed an evolutionary theory of society, using functional analysis and systems theory. These aspects of operators, however, will not be discussed further in this paper.

Various social structures, such as organizations and institutions, have the form of a network. Self-organization in such networks is triggered and fueled by an ideology or sociological force that is adhered to or shared by all participants in the network. Self-organization in human (social) and computer (technological) networks can give rise to a decentralized, distributed, self-healing systems, helping the actors in the network to protect their security by limiting influence of the entire system on individual actors. Self-organization emerges in the network as a distinctive synergistic behavior through combination of the behaviors of individual actors in the network.

The phenomenon of self-organization in the restricted sense also exists in many domains of nature, such as chemistry or biology. Examples of self-organization in the restricted sense in chemistry are molecular self-assembly, reaction-diffusion systems and oscillating chemical reactions, autocatalytic networks, liquid crystals, colloidal crystals, self-assembled monolayers, and microphase separation of block copolymers. Examples of self-organization in the restricted sense in biology are spontaneous folding of proteins and other biomacromolecules, formation of lipid bilayer membranes, pattern formation and morphogenesis, the coordination of human movement, the creation of structures by social animals discussed above.

The standard theoretical treatment of self-organization in the restricted sense is based both on microscopic, as well as macroscopic phenomenological approaches. Haken [56] formulated general principles of self-organization in the restricted sense or of self-organization as the reduction of complexity. They are based on general concepts, such as order parameters and the slaving principle. For instance, in large classes of systems their dynamics can be described by few order parameters.

In our view, however, the phenomenon of self-organization in the restricted sense, for which many more examples could be given in chemistry, biology and sociology are all dependent on the preexisting residual potentials present in the components of the system in its initial state. Since systems, accordingly, do not self-organize "by themselves", the possibilities for changing the evolution of the systems are limited.

The consequences, particularly in the social field are significant. For example, there are two main approaches to self-organization in social systems. One of them is based on the model of a system composed of aggregated individuals where self-organization is generated by and results in the sum of the practices in the system induced by control, or self-control, in particular, and the logic of the system logic. The second approach focuses on the practices of individuals as agents deviating from the logic of a system, making the existing system fluctuate and transforming its structure. The original capacities of such individuals to make changes should not, accordingly, be considered as a part of the self-organization that is alleged to occur.

#### 6.3.2. Self-Control

Proceeding from the general principles to self-operation and self-organization in human beings, we look at the process of self-control.

**Definition 26.** ***Self-control*** *in a broad sense is the property of a system to control itself, i.e., its internal processes and external behavior (functioning).*

The concepts of self-regulation (see next sub-Section) and self-control are very close to one another. However, the concept of self-regulation is more popular in system theory and cybernetics, while the concept of self-control is more popular in psychology. For instance, emotional self-control: presupposes keeping disruptive emotions and impulses under control. An example of self-control is when one manipulates one's own behavior by affecting states of deprivation or satiation. Psychological self-control means the ability to control one's emotions, behavior and desires in order to reach some goal, as well as the capacity of efficient behavior oriented at the future.

In society, self-control of an individual is directly related to the pressure/influence the individual may face. Thus, it is possible to distinguish three situations: good pressure/influence, bad pressure/influence and no pressure/influence. An example of a good pressure/influence is when an individual is in a competitive, yet non-judgmental and non-prejudicial environment and wants to be like those around. This makes the individual motivated and inspired to gain self-control. On the other hand, when an individual is in a judgmental and prejudicial environment, the individual may become depressed and unmotivated, losing self-control. In the third case, when an individual is free and there is no competition, self-control is based on how an individual may feel.

Skinner [58] gives a survey of nine categories of self-control methods. They include physical restraint and physical aid, changing the stimulus, drugs, operant conditioning, punishment depriving and satiating, manipulating emotional conditions, using aversive stimulation, and doing something else.

Manz et al. [59] argue that self-control is at the core of the organizational control process, expanding, in such a way, the view of control is developed in which importance of self-control, as well as external control mechanisms are recognized. For example, many companies are trying to become learning organizations often using self-control rather than relying on rules and regulations. Thus we can see that the difference between self-regulation and self-control is that self-control demands intentionality, while self-regulation, as a rule, does not need it.

#### 6.3.3. Self-Regulation

The next most general category within the framework of self-operation is self-regulation.

**Definition 27.** ***Self-regulation*** *in a broad sense is the property of a system to regulate its internal environment (state self-regulation) and external behavior or functioning (phase self-regulation) in order to maintain a stable, constant condition*.

Any self-regulating system is an operator, specifically, a *self-operator*. Self-regulating systems exist on all levels: cells in an organism, human organism, and many social organizations are self-regulating systems.

All self-regulation mechanisms have three interdependent basic components for the system feature, e.g., a system parameter, being regulated, as follows: (1) the *receptor system* is the sensing component that monitors and reflects changes in the system and its environment and sends information about these changes to the control unit; (2) the *control unit* (or *conceptor* in the sense of Burgin and Gladun [60] processes information that comes from the receptor, formatting instructions (operational information) to the effector; (3) the *effector system* is the acting component that changes in the system state, e.g., a system parameter, and/or system behavior (functioning).

Information plays a crucial role in self-regulation, which is based on the feedback of the system. It is possible to understand self-regulation through the interplay of positive and negative feedback cycles when some variations tend to reinforce themselves, while others tend to reduce themselves. Both types of feedback are important to self-regulation: positive feedback because it increases parameters of the system (up to the point where resources become insufficient) and negative feedback because it stabilizes these parameters.

Let us take the human organism as an example of a self-regulating system. In it, most homeostatic regulation is controlled by the release of hormones into the bloodstream, while other regulatory processes rely on simple diffusion to maintain a balance. The process of self-regulation proceeds as follows. The receptor system, which may consists of several components or even of many autonomous units, is sensing different stimuli. When a relevant stimulus comes, the receptor sends information to the control unit, finds the state/phase of the organism and determines an appropriate response to the stimulus. In the human organism, the control unit (control center) is the brain. Then the control center sends signals to the effector system or to a part of this system, which can be muscles, organs or other structures that receive signals from the control center. After receiving the signal, a change occurs to correct the deviation by either enhancing it with positive feedback or depressing it with negative feedback

Neural networks, both natural and artificial usually work in the same way, having the same components: receptors, conceptors and effectors [60] organized to maintain state self-regulation aimed at homeostasis.

## 7. Conclusions

### *7.1. A Unified Theory of Operators and Logic in Reality*

In this article, we have used two different but complementary strategies to arrive at a new coherent picture of operators in nature, mind and machines. First, we have defined and identified the *natural* operators that constitute and drive processes at the cognitive levels of reality. We have placed all operators in a comprehensive categorial framework, in which they are classified according to their form, media, target and function orientations and major general properties. The concept of *natural* operators at the level of the individual and collective human mind is given a new foundation. Second, we suggest that an applicable extension of logic, such as Logic in Reality (LIR), can make visible logical patterns in processes, in particular cognitive processes that have been previously considered irrational or disorganized, escaping any form of systematic or logical treatment. We show that the same structural logical principles are instantiated in the operators responsible in functioning of the human mind, the functioning of human society and the interaction between individual and group "minds".

From one standpoint, we may consider our approach as a 'Unified' Theory of Operators, at least from the point of view of a complete classification of operators. However, we do not wish to imply that our theory is somehow monolithic and 'closed'. Recent developments relevant to the problematic of unity have been well summarized by Jordi Cat in his article on the Unity of Science in the *Stanford Encyclopedia of Philosophy* [61]. Unification must not be created on the principle of 'unity for the sake of unity". Cat notes several positive, anti-reductionist trends in the philosophy of science, for example in moving from dependence on the concepts primarily taken from physics. One is struck, in fact, by the *diversity* of the sense, which can be given to the definition of unity! These definitions range from the possible single nature of the underlying 'stuff' of the universe, to the unification of the disciplines within science and with regard to these whether on is dealing with concepts or terms or the higher-level entity of theory and whether the relations are ones of reduction, explanation or logical inference. Cat suggests that the concept of unity can have a definite value at the practical level in science, to justify approaches and goals, and in philosophy to help choose what philosophical questions to pursue and what target areas to explore. The Unity of Science could suggest what science is to be referenced by philosophy as authoritative. He touches upon our point in the first paragraph by suggesting that "unities and unifications help us meet cognitive and practical demands upon our life . . . , contribute to our self-image and be seen as a source of aesthetic value, providing a grip on our intellectual imagination." Our approach extends this view: the value of the Unity of Science, self-referentially, is itself a unity of a sort that neglects, intentionally or not, the value of diversity. We believe it is necessary to do so explicitly. Informational processes, in our theory and logic, are composites of unities (identities) and diversities and all share in part the properties of their

opposites. We consider these remarks applicable to all theories of 'unity' whether of operators, science or knowledge.

### *7.2. Operators or Agents?*

The debate regarding agency has taken on further dimensions for theories of mind with the advent of complex computational capabilities that seem to confer some kind of autonomy on machine operators, the *artificial* or machine operators that have been discussed briefly in this paper. The problem is that that there is no exact unambiguous definition of an agent. Traditional definition assumes that an *agent* is a system who/that acts on behalf of or/and represents another system or other systems, e.g., a businessman who buys or sells for another in exchange for a commission. A more modern understanding treats an agent either as a system who/that exerts some force or effect or as a system who/that acts or has the power or authority to act.

According to the traditional definition, any agent is an operator because to accomplish his/her/its mission, the agent has to operate. However, there are operators that are not agents because they are aimed at their own goals. This gives us two types of operators:

- *Agent-operators who/that act/operate on behalf of or/and represent other systems*.
- *Free operators who/that are act/operate forming their goals and behavior by themselves*.

Following our discussion of the requirements for moral responsibility that human operators have and the principle, inherent in LIR of required indeterminacy in entities resulting from the chain of potentializations in natural evolution, implies that no machine that is the result of human determination can be considered completely autonomous. All the functionality involved in machines apparently capable of autonomy is the consequence of utilization of symbolic operators and operations. However, to some extent, the same conclusion is true for people, autonomy of which is always more or less restricted. There are even situations when people in a definite state, e.g., one equivalent to zombies, act like machines directed/operated by others.

We thus propose, as one main conclusion of this study, that only *natural* human operators should be designated as agents. All others, that is, all *symbolic* and machine operators are consequently, non-agents and should be so designated. An interesting problem is to consider groups and association of agents in the context of the operator theory developed in this paper because, typically, agents form groups and other collective agents, whose behavior, as in the case of groups discussed above, may not be totally reducible to that of individual members.

Our next intention is to develop the concept of operators in both theoretical and practical directions. Our approach, which uses the tools of the General Theory of Information and Logic in Reality, can be also applied to the categorization of the various types of symbolic operators—mathematical, logical and linguistic—being derivable from *natural* operators. In particular, the issue of *symbolic* operators in Information Technology and Computer Science is addressed elsewhere.

### *7.3. Operators in Philosophy*

Philosophical issues of our theory of operators arise in part due to changes in the foundations of knowledge, which both authors of this paper have discussed in different contexts. One focuses on the role and future of transdisciplinarity as a movement at the interface of science and philosophy. Another focuses on the impact on philosophy in general of the convergence in progress of the philosophy of information and information science.

In the preceding sections of this paper, we have provided definitions of operators primarily from the standpoints of mathematics, logic and science. The philosophical discussion was essentially limited to a classification of properties. Readers can see, we hope clearly, a thread of additional philosophical considerations running through this paper. In our analysis of both simple and complex operators, we take into account questions of causality, intentionality and meaning. Our discussion of the cognitive as well as physical role of operators thus raises further questions about their role in philosophy

itself; it is demonstrated in the Introduction that operators are basic for ontology and epistemology. In addition, the role of natural and cognitive operators cannot be included in a structureless concept of functionalism. The identification of natural operators in the cognitive domain suggests that a recursive analysis of the implications of operators for what philosophy 'is' is required. A proper philosophical discussion of complex operators must accept their ontological existence as dynamic structures. This amounts to a second order naturalization of our theory of operators.

Issues in the areas of the naturalization of phenomenology and the naturalization of the philosophy of science will be familiar to readers of this *Journal.* Naturalization means, roughly, 'bringing into science' from a domain, philosophy, that can be defined as a set of disciplines—logic, ontology, metaphysics, epistemology—and their use *via* reasoning and analysis to arrive at a viewpoint about what it is for human beings to be alive and think. The primary domain of naturalization has been that of the Philosophy of Science. In other areas, we note the naturalization of epistemology of Quine [62] although his discussion seems limited to linguistic logical elements. Another recent example is the naturalization of metaphysics by Ladyman, Ross and their colleagues [63]. We also note the application by Petitot-Cocorda [64] of his catastrophe theory to the naturalization of phenomenology, as well as a major compendium on the subject [65].

This study reflects the convergence of science and philosophy, at least in the area of information considered by Wu Kun, who writes that it is impossible to separate information – information science - from its philosophy or metaphilosophy [66]. Our tentative conclusion is therefore that there is a philosophical dimension to the natural operators, which we have observed acting in cognitive processes and involving information, that is, in all cognitive processes. This convergence is obviously not intended to imply an 'end' to philosophy or its conflation with science. Philosophy will continue to explore issues that arise, in particular, in relation to language and knowledge in their aspects as unique cognitive products of the human condition, with a substantial abstract content. However, the question of relations of that condition to the rest of the world logically requires retaining the scientific properties of that world to insure the validity of the comparison. There is a set of new and unique relationships, which are developing between the classical disciplines of science and philosophy based on a new understanding of the science and philosophy of information. The overall movement is philosophization of science and scientification of philosophy, as Wu has described it. An example of such scientification and naturalization of philosophy is the article by Brenner [67] on the naturalization of the philosophy of Rafael Capurro, who is a scientist and philosopher of information.

The role of operators is especially important in epistemology where cognitive and epistemic operators model cognitive processes providing efficient means for exploration of knowledge and information [68–71].

In conclusion, it is useful to understand that our new logical conception and categorization of operators and their functioning provides sound foundation for a scientification of philosophy. On the one hand, this conceptual paper provides a further basis for discussion of key issues in information and the philosophy of science in the emerging Information Society, since it describes both the quantitative and qualitative properties of information as both a *symbolic* and *natural* operator itself. On the other, our discussion of *natural* operators in machines, humans, organizations and knowledge provides a new perspective for the potential discussion of new models of society and a further basis for differentiation between human and non-human agents and their respective ethical characteristics.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Bickhard, M.H. The interactivist model. *Synthese* **2009**, *166*, 547–591. [CrossRef]
2. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
3. Nicolescu, B. *Manifesto of Transdisciplinarity*; State University of New York Press: Albany, NY, USA, 2002.

4. Bennett, M.R.; Hacker, P.M.S. *Philosophical Foundations of Neuroscience*; Blackwell Publishing: Malden, MA, USA, 2003.
5. Brenner, J.E. *Logic in Reality*; Springer: Dordrecht, The Netherlands, 2008.
6. Heisenberg, W. Über quantentheoretische Umdeutung kinematischer und mechanischer Beziehungen. *Z. Phys.* **1925**, *33*, 879–893. [CrossRef]
7. Jordan, T.F. *Linear Operators for Quantum Mechanics*; John Wiley & Sons, Inc.: NewYork, NY, USA, 1969.
8. Von Neumann, J. *Mathematical Foundations of Quantum Mechanics*; Princeton University Press: Princeton, NJ, USA, 1955.
9. Davies, E.B.; Lewis, J.T. An Operational Approach to Quantum Probability. *Commun. Math. Phys.* **1970**, *17*, 239–260. [CrossRef]
10. Kraus, K. *States, Effects and Operations*; Springer: Berlin, Germany, 1983.
11. Ludwig, G. Versuch einer axiomatischen Grundlegung der Quanten mechanik und allgemeinerer physikalischer Theorien. *Z. Phys.* **1964**, *181*, 233–260. [CrossRef]
12. Scarrott, G.G. The Nature of Information. *Comput. J.* **1989**, *32*, 262–266. [CrossRef]
13. Reading, A. The Biological Nature of Meaningful Information. *Biol. Theory* **2006**, *1*, 243–249. [CrossRef]
14. Csanyi, V. *Evolutionary Systems and Society*; Duke University Press: Durham, NC, USA; London, UK, 1989.
15. Kampis, G.A. Causal Model of Evolution. 1991. Available online: http://www.jaist.ac.jp/~g-kampis/SEAL02/A_Causal_Model_of_Evolution.htm (accessed on 21 March 2017).
16. Burgin, M.; Simon, I. Information, Energy, and Evolution. Preprintin Biology 2359, Cogprints. 2001. Available online: http://cogprints.ecs.soton.ac.uk (accessed on 21 March 2017).
17. Smith, J.M.; Szathmary, E. *The Origins of Life: From the Birth of Life to the Origin of Language I*; Oxford University Press: Oxford, UK, 1999.
18. Deacon, T.W. *Incomplete Nature. How Mind Evolved from Matter*; W.W. Norton & Co.: London, UK; New York, NY, USA, 2012.
19. Crutchfield, J.P. Information and its Metric. In *Nonlinear Structures in Physical Systems—Pattern Formation, Chaos and Waves*; Springer: New York, NY, USA, 1990.
20. Landauer, R. Information is Inevitably Physical. In *Feynman and Computation: Exploring the Limits of Computers*; Westview Press: Oxford, UK, 2002; pp. 76–92.
21. Kaye, D. The nature of information. *Libr. Rev.* **1995**, *44*, 37–48. [CrossRef]
22. Sinden, R. *DNA Structure and Function*; Elsevier: Amsterdam, The Netherlands, 1994.
23. Loewenstein, W.R. *The Touchstone of Life: Molecular Information, Cell Communication, and the Foundation of Life*; OxfordUniversity Press: Oxford, UK; New York, NY, USA, 1999.
24. Smolin, L. *The Life of the Cosmos*; Oxford University Press: Oxford, UK; New York, NY, USA, 1999.
25. Stonier, T. Towards a new theory of information. *J. Inf. Sci.* **1991**, *17*, 257–263. [CrossRef]
26. Bekenstein, J.D. Information in the holographic universe. *Sci. Am.* **2003**, *289*, 58–65. [CrossRef] [PubMed]
27. Wheeler, J.A. Information, Physics, Quantum: The Search for Links. In *Complexity, Entropy, and the Physics of Information*; Zurek, W., Ed.; Addison-Wesley: Redwood City, CA, USA, 1990; pp. 3–28.
28. Brenner, J.E. The Philosophical Logic of Stéphane Lupasco. *Log. Log. Philos.* **2010**, *19*, 243–285.
29. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
30. Jacquette, D. Philosophy of Logic. In *Handbook of the Philosophy of Science Series*; Gabbay, D., Woods, J., Thagard, P., Eds.; North-Holland Press (Elsevier): Amsterdam, The Netherlands, 2007.
31. Brenner, J.E. Stéphane Lupasco et la Rejonction Métalogique. In *A la Confluence de deux Cultures: Lupasco Aujourd'hui, Proceedings of the International UNESCO Colloquium, Paris, France, 24 March 2010*; Editions Oxus: Paris, France, 2010; pp. 250–285.
32. Brenner, J.E. Processing Reality: A Logical Offering. *Log. Philos.* **2005**, *14*, 165–189.
33. Burgin, M. *Theory of Named Sets*; Nova Science Publishers: New York, NY, USA, 2011.
34. Ghils, P. *Les Tensions du Langage*; Peter Lang: Berne/Berlin, Germany, 1994.
35. Jakobson, R. *Language in Literature*; Belknap Press: Cambridge, MA, USA, 1987.
36. Prigogine, I. *From Being to Becoming: Time and Complexity in Physical Systems*; Freeman & Co.: San Francisco, CA, USA, 1980.
37. Woodward, J. Causation and Manipulability. In *Stanford Encyclopedia of Philosophy (Fall 2001 Edition)*; Zalta, E.N., Ed.; Stanford University: Stanford, CA, USA, 2001. Available online: http://plato.stanford.edu/archives/fall2001/entries/causation-mani/ (accessed on 21 March 2017).

38. Dowe, P. Causal Processes. In *The Stanford Encyclopedia of Philosophy (Fall 2008 Edition)*; Zalta, E.N., Ed.; Stanford University: Stanford, CA, USA, 2008. Available online: http://plato.standford.edu/archives/fall2008/entries/causation-process/ (accessed on 21 March 2017).
39. Jung, C.G. *Two Essays on Analytical Psychology*; Adler, G., Hull, R.F.C., Eds.; World Publishing (Meridian): New York, NY, USA; Cleveland, OH, USA, 1971; pp. 63–71.
40. Minati, G.; Pessa, E. *Collective Beings*; Springer: Dordrecht, The Netherlands, 2006.
41. Freud, S. *The Ego and the Id*; The Hogarth Press Ltd.: London, UK, 1949.
42. Burgin, M. Structural traits of psychoanalysis. *Visnik Natl. Acad. Sci. Ukr.* **1994**, *2*, 45–50. (In Ukrainian)
43. Lupasco, S. *L'énergie et la Matière Vivante*; Éditions du Rocher: Monaco-Ville, Monaco, 1986.
44. Burgin, M. Triad as a Fundamental Structure in Human Culture. *Stud. Culturol.* **1993**, *2*, 51–63.
45. Leydesdorff, L. *The Knowledge-Based Economy: Modeled, Measured, Simulated*; Universal Publishers: Boca Raton, FL, USA, 2006.
46. D'Entreves, M.P. Hannah Arendt. In *The Stanford Encyclopedia of Philosophy (Fall 2008 Edition)*; Zalta, E.N., Ed.; Stanford University: Stanford, CA, USA, 2008. Available online: http://plato.stanford.edu/archives/fall2008/entries/arendt/ (accessed on 21 March 2017).
47. Magnani, L. *Morality in a Technological World: Knowledge as Duty*; Cambridge University Press: New York, NY, USA, 2007.
48. Thom, R. *Stabilité Structurelle et Morphogénèse*; W.A. Benjamin, Inc.: Reading, MA, USA, 1972.
49. Lupasco, S. *Logique et Contradiction*; Presses Universitaires de France: Paris, France, 1947.
50. Esfeld, M.; Lam, V. Holism and Structural Realism. In *Worldviews Science and Us: Studies of Analytical Metaphysics*; Vanderbeeken, R., Weber, E., Eds.; World Scientific: Singapore, 2010.
51. Smith, D.W. *Intentionality Naturalized? In Naturalizing Phenomenology: Issues in Contemporary Phenomenology and Cognitive Science*; Petitot, J., Varela, F.J., Pachoud, B., Roy, J.-M., Eds.; Stanford University Press: Stanford, CA, USA, 1999.
52. Seibt, J. Forms of interaction in General Process Theory. *Synthese* **2009**, *166*, 479–512. [CrossRef]
53. Wu, K. The Basic Theory of Philosophy of Information. In Proceedings of the 4th International Conference on the Foundations of Information Science, Beijing, China, 21–24 August 2010.
54. Fuchs, C. The Dialectic of the Nature-Society System. *Triple C* **2006**, *4*, 1–39.
55. Varela, F.J. The Specious Present. In *Naturalizing Phenomenology: Issues in Contemporary Phenomenology and Cognitive Science*; Petitot, J., Varela, F.J., Pachoud, B., Roy, J.-M., Eds.; Stanford University Press: Stanford, CA, USA, 1999.
56. Haken, H. *Advanced Synergetics: Instability Hierarchies of Self-Organizing Systems and Devices*; Springer: New York, NY, USA, 1993.
57. Luhmann, N. *Social Systems*; Stanford University Press: Stanford, CA, USA, 1995.
58. Skinner, B.F. *Science and Human Behavior*; Simon & Schuster: New York, NY, USA, 1953.
59. Manz, C.C.; Mossholder, K.W.; Luthanvs, F. Controlling Information Systems Development Projects: The View from the Client. *Manag. Sci.* **2002**, *48*, 484–498.
60. Burgin, M.; Gladun, V. Mathematical Foundations of Semantic Networks Theory. *Lect. Notes Comput. Sci.* **1989**, *364*, 117–135.
61. Cat, J. The Unity of Science. In *Stanford Encyclopedia of Philosophy (Winter 2014 Edition)*; Zalta, E.N., Ed.; Stanford University: Stanford, CA, USA, 2014; Available online: http://plato.stanford.edu/archives/win2014/entries/scientific-unity/ (accessed on 21 March 2017).
62. Van Quine, W.O. Epistemology Naturalized. In *Ontological Relativity and Other Essays*; Columbia University Press: New York, NY, USA, 1969; Chapter3.
63. Ladyman, J.; Ross, D. *Every Thing Must Go: Metaphysics Naturalized*; Oxford University Press: Oxford, UK, 2007.
64. Petitot-Cocorda, J. *Physique du Sens*; CNRS Éditions: Paris, France, 1992.
65. Petitot, J.; Varela, F.J.; Pachoud, B.; Roy, J.-M. (Eds.) *Naturalizing Phenomenology: Issues in Contemporary Phenomenology and Cognitive Science*; Stanford University Press: Stanford, CA, USA, 1999.
66. Wu, K. The Interaction and Convergence of the Philosophy and Science of Information. *Philosophies* **2016**, *1*, 228–244. [CrossRef]
67. Brenner, J.E. The 'Naturalization' of the Philosophy of Rafael Capurro. In *A Festschrift in Honor of Rafael Capurro*; Kelly, M., Bielby, J., Eds.; Springer: Wiesbaden, Germany, 2016; pp. 45–64.

68. Burgin, M. Epistemic Information in Stratified M-Spaces. *Information* **2011**, *2*, 697–726. [CrossRef]
69. Burgin, M. Weighted E-Spaces and Epistemic Information Operators. *Information* **2014**, *5*, 357–388. [CrossRef]
70. Burgin, M. *Theory of Knowledge: Structures and Processes*; World Scientific: New York, NY, USA; London, UK; Singapore, 2016.
71. Burgin, M.; Mikkilineni, R.; Mittal, S. Knowledge processing as structure transformation. *Proceedings* **2017**, *1*, 212. [CrossRef]

© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

*proceedings*

MDPI

*Proceedings*

# Problems in the System of Scientific Knowledge †

**Mark Burgin**

Department of Computer Science, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

† Presented at the IS4SI 2017 Summit DIGITALISATION FOR A SUSTAINABLE SOCIETY, Gothenburg, Sweden, 12–16 June 2017.

Published: 8 June 2017

**Abstract:** Problems play a crucial role in science. However, to correctly understand this role, it is necessary to have an adequate model of scientific knowledge. Here we use the most advanced and complete model called the modal stratified bond model of comprehensive knowledge systems. According to this model, problems are a specific kind of knowledge called erotetic knowledge generating the extensive erotetic system (modality) of scientific knowledge. Here we give a brief exposition of this system analyzing properties and aspects with the aim to determine the best form of problem description.

**Keywords:** problem; theory; science; system; modality; scientific theory; information

## 1. Introduction

To properly figure out the place and role of problems in science, it is necessary to use a scientific/mathematical model of the system of scientific knowledge. Only such a model can provide means for exploring important problem traits and functioning, interaction of problems with other elements of scientific knowledge and influence of problems on the development of science in general and specific scientific fields, in particular. Here we are mostly interested in problems in and related to information science as representatives of a special type of knowledge, which called *erotetic knowledge*.

The most widespread image of scientific knowledge that people have is a collection of statements about studied objects, processes and phenomena. This image was formalized and studied in the *standard model* (also called the *positivist model*) of a scientific theory, which utilized the logical dimension to represent a scientific theory as a system of propositions (cf., for example, [1,2]).

Another popular approach to description of the scientific theory structure was the *structuralist model* (*reconstruction*) of a scientific theory, which used the set-theoretical dimension to represent a scientific theory as a system of models of the theory domain (cf., for example, [3,4]).

These and other approaches in methodology of science were unified in the *structure-nominative model* of a scientific theory, which gave a four-dimensional knowledge representation including the logical and set-theoretical dimensions [5–7].

The further development of the structure-nominative model of a scientific theory brought forth the most advanced is the *modal stratified bond model* of comprehensive knowledge systems, which encompasses the structure-nominative model and provides a nine-dimensional knowledge representation [8]. In this context, advanced scientific theories are examples of comprehensive knowledge systems. Three of these dimensions, which are related to problems, are analyzed in the next section.

## 2. Erotetic Systems as Intrinsic Components of Comprehensive Knowledge Systems

At first, let us define a problem.

*A problem* is absence of something, e.g., of knowledge or information, and explicit representation of this absence.

The place of scientific problems is determined by models of scientific knowledge. The modal stratified bond model of scientific knowledge as a whole is stratified into several subsystems forming three *epistemic directions*—the modal, systemic and hierarchical directions.

The *modal direction* (*stratification*) of scientific knowledge contains three classes:

- *Assertoric knowledge* consists of epistemic structures with implicit or explicit affirmation of being knowledge, such as theorems, propositions and lemmas.
- *Hypothetic* or *heuristic knowledge* consists of epistemic structures with implicit or explicit supposition that they may be knowledge, such as conjectures and hypotheses.
- *Erotetic knowledge* consists of epistemic structures that express lack of knowledge, such as problems, questions and dilemmas.

For instance, according to the modal stratification of scientific knowledge [8], the horizontal structure of each subsystem of an advanced knowledge system, e.g., the logic-linguistic subsystem of a scientific theory, which includes the descriptive knowledge of this theory, has the form presented in Figure 1.

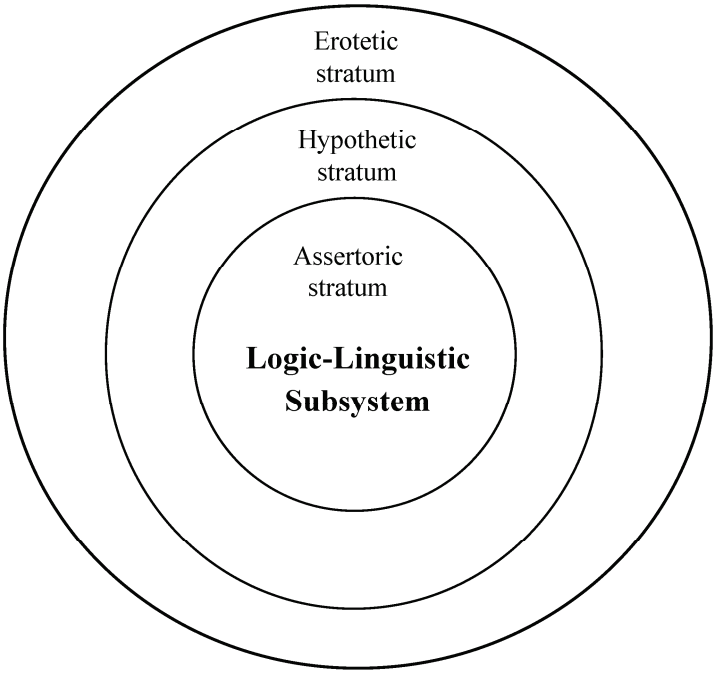



**Figure1.** The modal stratification of the logic-linguistic subsystem of scientific knowledge.

The *hierarchical direction* of scientific knowledge comprises three levels:

1. The *componential level*
2. The *attributed level*
3. The *productive level*

The *systemic direction* of scientific knowledge is differentiated into three categories:

- *Descriptive knowledge*
- *Representational knowledge*
- *Operational knowledge*

As we can see, various types and kinds of problems form the erotetic system (stratum or modality) of scientific knowledge with respect to the model direction. Two other epistemic directions—systemic and hierarchical—disclose the structure of the erotetic system and the place problems in it. Namely, we have the following components of the erotetic system:

- *Descriptive knowledge* develops different forms of concepts and terms on the componential level, utilizes erotetic languages and grammars on the attributed level and employs erotetic logical calculi and logical varieties [8] on the productive level.
- *Representational knowledge* constructs various forms of properties and names for problems on the componential level, elaborates a variety of models of problems on the attributed level and employs systems of models for problems on the productive level.
- *Operational knowledge* builds various operations and scales for problems on the componential level, elaborates a variety of algorithms, procedures, automata, estimates, measures, norms and values on the attributed level and employs operational and assessment algebras and calculi on the productive level.

Going in the systemic direction, we see that a problem must be described in some language. There are three basic forms of such descriptions:

- A *question* explaining what is absent and requiring the absent essence
- A *task* explaining what is absent and instructing to get (obtain) the absent essence
- A *conjecture* tentatively describing the absent essence

As an example, let us consider three forms of the following problem:

Question: *What is the comprehensive definition of information?*
Task: *Construct or find the comprehensive definition of information.*
Conjecture: *The general theory of information gives the comprehensive definition of information.*

The representational knowledge related to this problem includes the meaning of terms:

- The meaning of the word *information*
- The meaning of the word *definition*
- The meaning of the word *comprehensive*

In turn, the meaning of the word *information* includes existing definitions and descriptions of information and related phenomena; usage of the word *information* now and in the historical perspective; and relations of the notion of information to other notions and concepts.

The representational knowledge related to this problem also includes relations of this problem to other problems, for example, relations to the problem of adequate measuring information.

The operational knowledge related to this problem includes different operations demanded to solve this problem. For instance, at first, it is necessary to use operations of construction or/and search. After some linguistic expression is found, it is necessary to show (prove) that it a definition and not only some description. The next step includes operations of demonstration (proving) that this is a definition of information. The next step includes demonstration (proving) that this definition of information is comprehensive. The last step demands analysis of a variety of different approaches and directions in information theory. Many of them are described in [9] but there are even more.

Unfortunately, there is a negative tendency in information studies when researchers take only Shannon's theory of communication (information), demonstrate that it does not represent many types and kinds of information and then suggest their own definition, or more often, a description of information. This situation represents the straw man fallacy.

All this gives only the first approximation of the structure of problems. To obtain a more exact and detailed structure of problems, it is necessary to use the hierarchical dimension and take into account existing types of operational knowledge. However, when a problem is formulated for experts in some area, many parts of its structure remains implicit because it is implied that experts understand them. It is possible to find more information about these structures in the book "Theory of Knowledge" [8].

## 3. Problem Structure and Its Presentation

In addition to forms of problems, we analyze the structure of problems with the goal of development models for representation articles in the problem book. The extended problem structure spans in three directions (dimensions):

- The *temporal problem dimension* reveals the history of the problem, its current standing and its projection on the development of the scientific field.
- The *relational problem dimension* reflects relations of the problem to its domain, associated problems, as well as to other elements of scientific knowledge, such as theorems or procedures, and other fields of knowledge and domains of reality.
- The *innate problem dimension* describes essential properties, traits and parameters of the problem.

For instance, the problem *What is a comprehensive definition of information*? is related to such objects and systems as the diversity of information circulating in society (a part of the problem domain), to the associated problem of finding an adequate definition of knowledge, to algorithms, procedures and information processing systems, and to physics, biology and psychology.

A solution of a problem is an important element of the innate problem dimension for any problem. Thus, when problem is formulated, it might be useful to describe desirable and expected properties of the problem solution. However, it is necessary to understand that some problems have easily identifiable solutions while solutions of other problems can be vague and uncertain.

The form of a problem is an important property of this problem. Another important property is its complexity. For instance, complexity of mathematical problems is studied in [10], while complexity of algorithmic problems is studied in [11].

The analysis of the problem structure allows developing a tentative format of articles representing problems in the future book "Gothenburg Book: A Compendium of Problems in Information Studies", the primary goal of which is to stimulate the further development of information science, organize its functioning on sound foundations and create means for its better application and utilization.

Thus, it is possible to suggest three types of problem exposition (representation) in the book:

1. *Pure problem*
2. *Annotated problem*
3. *Expanded problem*

*Pure problem exposition* must have three basic components:

1. The Name of the problem (It may coincide with the problem Description)
2. The Name of the problem author(s), or in general, identification of the problem source
3. The problem Description (verbalization)

At the same time, pure problem exposition can include other materials related to the problem.

*Annotated problem exposition* has the following components:

1. The Name of the problem (It may coincide with the problem Description)
2. The Name of the problem author(s), or in general, identification of the problem source
3. The problem Description (verbalization)
4. Explanation of the terms used in the problem Description
5. Existing sources (if any), where this problem is considered
6. Other considerations related to this problem

It is also possible to suggest the following structure for an *expanded problem exposition*.

1. The Name of the problem (as the title)
2. Author(s) who is (are) suggesting this problem
3. The problem Description (verbalization) and analysis
4. Origin of the problem and the history of approaches to its solution
5. Contemporary state (Research related to the problem)

6. Desirable and/or expected properties of the problem solution
7. Relations to other problems
8. Relations to other scientific and/or practical domains
9. General remarks
10. References and bibliography on the problem

The author of the problem can also include other materials in such a text.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Carnap, R. *Der Logische Aufbau der Welt. Scheinprobleme in der Philosophie, Berlin, 1928 (English Translation: The Logical Structure of the World*; Pseudoproblems in Philosophy: London, UK; Berkeley, CA, USA, 1967.
2. Suppe, F. The Positivist Model of Scientific Theories. In *Scientific Inquiry*; Oxford University Press: New York, NY, USA, 1999; pp. 16–24.
3. Sneed, J.D. *The Logical Structure of Mathematical Physics*; D. Reidel: Dordrecht, The Netherlands, 1971.
4. Balzer, W.; Moulines, C.U.; Sneed, J.D. *An Architectonic for Science. The Structuralist Program*; Reidel: Dordrecht, The Netherlands, 1987.
5. Burgin, M.; Kuznetsov, V. The structure-nominative reconstruction of scientific knowledge. *Epistemologia* **1988**, *11*, 235–254.
6. Balzer, W.; Burgin, M.; Kuznetsov, V. Reduction and the Structure-Nominative View of Theories. In Proceedings of the 9th International Congress of Logic, Methodology and Philosophy of Science, Uppsala, Sweden, 7–14 August 1991; Volume 2, p. 6
7. Burgin, M.; Kuznetsov, V. *Introduction to Modern Exact Methodology of Science*; International Science Foundation: Moscow, Russia, 1994. (In Russian)
8. Burgin, M. *Theory of Knowledge*: *Structures and Processes*; World Scientific: New York, NY, USA; London, UK; Singapore, 2016.
9. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
10. Burgin, M.; Calude, C.S.; Calude, E. Inductive Complexity Measures for Mathematical Problems. *Int. J. Found. Comput. Sci.* **2013**, *24*, 487–500.
11. Burgin, M. Algorithmic Complexity of Computational Problems. *Int. J. Comput. Inf. Technol.* **2010**, *2*, 149–187.

© 2017 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

*Proceedings*

# Principles of General Ecology †

**Mark Burgin**

Department of Computer Science, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

† Presented at the IS4SI 2017 Summit DIGITALISATION FOR A SUSTAINABLE SOCIETY, Gothenburg, Sweden, 12–16 June 2017.

Published: 9 June 2017

**Abstract:** Ecological systems are studied in many areas. There are different ecological areas: plant ecology, animal ecology, natural ecology, human ecology, industrial ecology, information ecology, ecology of mind, knowledge ecology and so on. Here we develop a unifying approach to ecological studies developing general ecology, which encompasses and organizes different directions on a reliable foundation. Information ecology as one of the basic areas in ecological studies is placed in a resourceful environment allowing its accelerated development and advanced expansion.

**Keywords:** information; system; ecology; ecological system; world; component; technology

## 1. Introduction: A Historical Perspective

The term *ecology* (*Ökologie* in German) was coined in 1866 by the German scientist Ernst Haeckel (1834–1919) from two Greek words *oikos*, which means *house*, or more generally, *habitat* or place of living and logos, which was used in ancient Greece denoting such concepts as *order*, *meaning*, *foundation* or *mind* [1]. Haeckel's initiative instigated an approach, where European botanists investigated plant communities related to definite territories and their interdependencies, giving rise to the science of ecology, which was dealing not only with plants but also with other living beings.

In the contemporary science, ecology is a holistic study of living systems in relation to their environment by explicating patterns of, processes in and relationships between these systems.

At the same time, ecology as a whole contains such subdisciplines as plant ecology and animal ecology.

*Plant ecology* studies the distribution and abundance of plants, the effects of environmental factors upon the abundance of plants, and the interactions among and between plants and other organisms [2].

*Animal ecology* is the scientific study of animals and how they related to and interact with each other, as well as with their environment, determining the distribution and abundance of organisms.

Together these two areas form *natural ecology*, whereas researchers also created other ecological fields. One of them is *human ecology*, which is an interdisciplinary and transdisciplinary study of the relationships between humans and their natural, social, and technological environments involving a variety of disciplines: geography, sociology, psychology, anthropology, zoology, epidemiology, public health, home economics, and natural ecology, among others [3].

While ecology has traditionally dealt only with natural systems, the new field of *industrial ecology* studies industrial products as part of larger systems and processes including industrial behavior and biogeochemical cycles as a part of a system and aiming at reduction of the environmental impacts of production, consumption, and disposal.

Chinese scientist Yixin Zhong initiated *information ecology* [4,5]. This discipline is essentially important for information studies as a holistic approach to the existence and functioning of information processing systems, as well as for better understanding of information processes in all

spheres of reality. If ecology of plants studies structures and processes in systems of plants, information ecology studies structures and processes in organizations of information processing systems and formations.

One more ecological area is *ecology of mind* suggested by Bateson [6].

Researchers also study *knowledge ecology* [7,8], which is an approach to knowledge management aimed at fostering the dynamic evolution of knowledge interactions between systems to advance decision-making and innovation by means of enhanced evolutionary networks of collaboration. In contrast to purely instructional management, which attempts either to manage or to direct outcomes, knowledge ecosystems advocate that knowledge strategies should focus more on enabling flexible self-organization and self-improvement in response to changing environments.

In addition, American anarchist and libertarian socialist author Murray Bookchin introduced *social ecology* as a critical study of society [9].

## 2. Principles and Structures of General Ecology

Existence of different ecological disciplines needs a common foundation and presented in this work *general ecology* provides such a unifying foundation for all ecological studies.

The concept of *ecosystem* proposed by the English ecologist Arthur Tansley is central for different ecological disciplines. That is why we start our exposition with defining this concept in the most general context. To do this, we describe how the global structure of the world affects the organization of ecosystems.

The large-scale structure of the world is represented by the *Existential Triad* [10], which is presented in Figure 1.

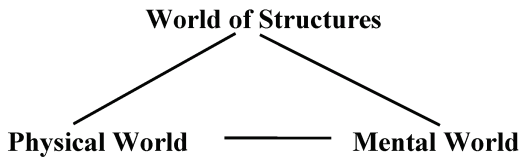



**Figure1.** The Existential Triad of the World.

The three worlds from the Existential Triad are not separate realities: they interact and intersect. Individual mentality is based on the brain, which is a material thing, while in the opinion of many physicists mentality influences physical world (cf., for example, [11]). At the same time, our knowledge of the physical world largely depends on interaction between mental and material worlds.

Note that not only people but also all information processing systems have their mentality. Let us look at a computer. The content of the computer's memory can be naturally treated as the mentality of this computer. For instance, the operating system is a part of the mentality of the computer.

The World of Structures consists of various forms and types of structures perfectly matching to Plato's World of Ideas/Forms because ideas or forms are correctly associated with structures. Structures exist like material things, such as tables, chairs, or buildings do, and form the *structural level* of the world. When it is necessary to learn or to create a system or to start a process, it is done, as a rule, by means of knowledge of the corresponding structure. Structures mold things in their being and comprehension.

The global structure of the world induces three types of ecosystems:

- *Physical ecosystem* includes physical systems and processes as its elements and components
- *Mental ecosystem* includes mental systems and processes as its elements and components
- *Structural ecosystem* includes physical systems and processes as its elements and components

When all three components of the world stratification are combined in one system, we have a *total ecosystem*. Such a total ecosystem has three constituents:

- The physical constituent of the ecological system and its environment
- The structural constituent of the ecological system and its environment

➢ The mental constituent of the ecological system and its environment

An ecosystem is delineated by three parameters:

– A region in the space, i.e., it is assumed that all elements and components of an ecological system belong to a definite region in the space
– The primary types of its elements/components, i.e., it is determined what elements and components of given ecological system are considered the most important from the point of view of ecological studies
– The basic types of connections between its elements/components including processes as dynamic connections, i.e., it is determined what connections, ties and processes in given ecological system are considered the most important from the point of view of ecological studies

For instance, in a natural ecosystem, living organisms form the primary type of elements and a chosen area on the Earth shapes the region in the space. In this context, a natural ecosystem is composed of the dynamically interacting parts including all living organisms in a given area, which interact with each other and with their non-living environment.

In an information ecosystem, information processing systems form the primary type of elements and a chosen area on the Earth (may be the whole Earth) shapes the region in the space in which information processing systems are interacting with each other, and also with their environments. In addition, studies of information ecosystems concentrates on information processes going in the system.

Note that there are different kinds of information processing systems: technical information processing systems, living information processing systems, human information processing systems and so on.

Three grades of (types of) elements/components:

- Primary or leading elements/components
- Secondary or auxiliary elements/components
- Tertiary or background elements/components

Ecological studies are aimed at understanding existence and functioning of the primary elements/components of ecosystems, as well as basic connections, ties and processes in these ecosystems.

A physical ecosystem contains parts, elements and components of three kinds:

- Natural parts, elements and components, which include physical systems and processes in nature
- Technological parts, elements and components, which include technological systems and processes
- Social parts, elements and components, which include social systems and processes

In a physical ecosystem, it is possible to consider only physical processes or also to take into account mental and information processes.

A mental ecosystem contains parts, elements and components of three kinds:

- Natural parts, elements and components, which include and comprise mentality and its components of living beings
- Technological parts, elements and components, which include and comprise mentality and its components of technical devices
- Social parts, elements and components, which include and comprise mentality and its components of groups, communities and societies of living beings and technical devices

In a mental ecosystem, it is possible to consider only mental processes or also to take into account information processes.

A structural ecosystem contains parts, elements and components of three kinds:

- Natural parts, elements and components, which include structures of physical systems and processes
- Technological parts, elements and components, which include structures of technological systems and processes
- Social parts, elements and components, which include structures of social systems and processes

## 3. Conclusions

The general ecology standpoint shows that it is possible to study information ecosystems either as physical ecosystems or as mental ecosystems or as structural ecosystems. It gives three perspectives at information ecosystems allowing researchers to obtain better knowledge and understanding of these systems. One more possibility is to study total information ecosystems combining all three perspectives in one model.

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Odum, E.P. *Fundamentals of Ecology*; Cengage Learning: Boston, MA, USA, 2004.
2. Weaver, J.E.; Clements, F.E. *Plant Ecology*; McGraw-Hill Book Company: New York, NY, USA, 1938.
3. Young, G.L. Human ecology as an interdisciplinary concept: A critical inquiry. *Adv. Ecol. Res.* **1974**, *8*, 1–105.
4. Zhong, Y.X. *Principles of Information Science*; BUPT Press: Beijing, China, 1988. (In Chinese)
5. Zhong, Y.X. The Law of "Information Conversion and Intelligence Creation". In *Information Studies and the Quest for Transdisciplinarity*; World Scientific: New York, NY, USA; London, UK; Singapore, 2017.
6. Bateson, G. *Steps to an Ecology of Mind*; Jason Aronson Inc.: St. Albans, UK, 1973.
7. Bray, D.A. Knowledge Ecosystems: A Theoretical Lens for Organizations Confronting Hyperturbulent Environments. In *Organizational Dynamics of Technology-Based Innovation*: *Diversifying the Research Agenda*; Springer: Medford, MA, USA, 2007; pp. 457–462.
8. Shrivastava, P. *Knowledge Ecology*: *Knowledge Ecosystems for Business Education and Training*; 1998. Available online: http://www.facstaff.bucknell.edu/shrivast/KnowledgeEcology.html (accessed on 18 May 2017).
9. Bookchin, M. *The Ecology of Freedom*; AK Press: Stirling, UK, 2005.
10. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
11. Herbert, N. *Quantum Reality*: *Beyond the New Physics*; Anchor Books: New York, NY, USA, 1987.

© 2017 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

*proceedings*

MDPI

*Proceedings*

# Bidirectional Named Sets as Structural Models of Interpersonal Communication †

**Mark Burgin**

Department of Computer Science, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

† Presented at the IS4SI 2017 Summit DIGITALISATION FOR A SUSTAINABLE SOCIETY, Gothenburg, Sweden, 12–16 June 2017.

Published: 8 June 2017

**Abstract:** Treating communication as information exchange between systems, we employ the most fundamental structure in mathematics, nature and cognition, which is called a named set or a fundamental triad because it has been useful in a variety of areas such as networks and networking, physics, information theory, mathematics, logic, database theory and practice, artificial intelligence, mathematical linguistics, epistemology and methodology of science, to mention but a few. Here we use structural models based on the theory of named sets for description and analysis of interpersonal communication explicating its structural regularities.

**Keywords:** information; structure; structural information; interaction; correctness; knowledge

## 1. Introduction

There are different models of communication (cf., for example, [1–3]). They distinguish two basic types of communication—interpersonal communication and intrapersonal communication. Interpersonal communication is information exchange between different systems. For instance, human-computer communication is interpersonal. Intrapersonal communication is information exchange in one system. For instance, communication in the World Wide Web is intrapersonal with respect to the World Wide Web. Here we use structural models based on the theory of named sets [4] for description and analysis of interpersonal communication explicating its structural regularities.

Being the most fundamental structure in mathematics, nature and cognition (as it is demonstrated in [4–6], named sets (also called fundamental triads) have been useful in a variety of areas such as networks and networking, physics, information theory, mathematics, logic, database theory and practice, artificial intelligence, mathematical linguistics, epistemology and methodology of science, to mention but a few. Application of named set theory to communication theory is based on the fact that communication as an information exchange has the intrinsic structure of a named set.

In the next section, we consider elements of named set theory necessary for communication studies, which are exposed in Section 3.

## 2. Named Sets and Fundamental Triads

We consider three primary types of named sets and fundamental triads [4].

A basic fundamental triad or a basic named set has the following form (1).

$$X \xrightarrow{\quad f \quad} N \tag{1}$$

It is a triad $\mathbf{X} = (X, f, I)$, in which $X$ and $N$ are two objects and $f$ is a correspondence (e.g., a binary relation) between $X$ and $I$. With respect to $\mathbf{X}$, $X$ is called the *support* of $\mathbf{X}$, $N$ is called the *component of*

*names* (*reflector*) or *set of names* of **X**, and *f* is called the *naming correspondence* (*reflection*) of **X**. Note that here, *f* is not necessarily a mapping or a function.

The standard example is a basic named set (fundamental triad), in which *X* consists of people, *N* consists of their names and *f* is the correspondence between people and their names. Another example is a basic named set (fundamental triad), in which *X* consists of things, *N* consists of their names and *f* is the correspondence between things and their names [7].

It is necessary to make a distinction between triples and triads. A *triple* is any set with three elements, while a *triad* is a system of three connected elements (components). It is worthy of note that mathematicians introduced the concept of a triple in an abstract category [8]. In essence, such a triple is a triad that consists of three fundamental triads and thus is a triad of the second order [4]. Understanding of the complex nature of the categorical triple made mathematicians to change the name of this structure and now it is always called a *monad* [9]. Interestingly, this shows connection between fundamental triads and Leibniz monads.

A *bidirectional fundamental triad or a bidirectional named set* has the following form (2).

$$X \xleftrightarrow{\quad f \quad} N \tag{2}$$

It is also a triad **D** = (*X*, *f*, *Y*), in which the naming relation *f* goes in two directions.

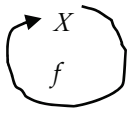
We have an example of a bidirectional named set when two people are exchanging messages, e.g., be e-mails, messaging or talking to one to another. In this case, *X* and *Z* are people while *f* and *g* are messages that go from one person to another.

Note that when mathematicians or computer scientists use connections without direction such as those that are used, for example, in general graphs [10], these connections actually have both directions and are more explicitly represented by the union of directed connections *h* and *g*.

A *cyclic fundamental triad or a cyclic named set* has the following form (3)

$$X \xrightarrow{\quad f \quad} X \tag{3}$$

The following graphic form (4) can also describe it.




(4)

An example of a cyclic named set is a subatomic particle, such as an electron, which acts on itself (cf., for example, [11]).

Another example of a cyclic named set is a computer network. In it, *X* consists of computers and *f* contains all connections between them.

Let us obtain some simple properties of named sets related to their compositions.

**Proposition 1.**

*(a) The sequential composition of basic named sets is a basic named set.*
*(b) The sequential composition of bidirectional named sets is a bidirectional named set.*
*(c) The sequential composition of cyclic named sets is a cyclic named set.*

In many cases, a bidirectional named set can be decomposed into the inverse composition of two basic named sets as it is demonstrated in the following diagram, which is a decomposition of Diagram (2).

$$X \underset{g}{\overset{h}{\rightleftarrows}} Z \tag{5}$$

Here $f = [h, g]$ or $f = h \cup g$. Thus, a decomposed bidirectional named set D is denoted by D = (*X*, [*h*, *g*], *Z*) and has two components, which are basic named sets:

- The direct component **X** = (*X*, *f*, *Z*)
- The inverse component **Y** = (*Z*, *g*, *X*)

**Proposition 2.** *A bidirectional named set **D** is equal to the inverse composition of its components **X** and **Y**.*

For instance, the set-theoretical bidirectional named set

$$X = (X = \{1, 2, 3\}, f, Z = \{a, b\})$$

with the naming correspondence (a binary relation in this case)

$$f = \{(1, a), (3, b), (2, a), (b, 1), (a, 3)\}$$

is decomposable the inverse composition of the direct component $Z = (X, h, Z)$ and inverse component $Y = (Z, g, X)$ where

$$h = \{(1, a), (3, b), (2, a)\} \subseteq X \times Y$$

and

$$g = \{(b, 1), (a, 3)\} \subseteq Y \times X$$

We see that in this case, $f = h \cup g$.

This shows how it is possible to construct bidirectional named sets using inverse composition of basic named sets [4].

*Inverse composition* of basic named sets $X = (X, f, I)$ and $Y = (Y, g, J)$ is defined as

$$\mathbf{X} \between \mathbf{Y} = (X, f, I) \between (Y, g, J) = (X \cup J, f \cup g^{-1}, I \cup Y)$$

**Proposition 3.** *The inverse composition of named sets **X** and **Y** is equal to the sequential composition of X and the involution **Y**° of **Y**, i.e.,*

$$\mathbf{X} \between \mathbf{Y} = \mathbf{X} \circ \mathbf{Y}^{\circ}$$

Although any bidirectional named set is the inverse composition of basic named sets, it is a fundamental structure such as a set, graph, category, fuzzy set or multiset. In more detail, relations between basic and bidirectional named sets are studied elsewhere.

There are also other compositions of named sets [4].

If $X = (X, r, I)$ and $Y = (Y, q, J)$ are named sets, then their *sequential composition* $X \circ Y$ is the named set $(X, r_{\circ}q_{\circ}, J)$ where $r_{\circ} = r \cap (X \times (I \cap Y))$ and $q_{\circ} = q \cap ((I \cap Y) \times J)$.

**Example 1.** *Superposition of functions is the sequential composition of the corresponding named sets in the case when $I = Y$.*

**Example 2.** *Composition of morphisms in categories is the sequential composition of the corresponding named sets.*

If $X = (X, r, I)$ and $Y = (Y, q, J)$ are named sets, then their *parallel composition* $X \oplus Y$ is

$$\mathbf{X} \oplus \mathbf{Y} = (X, f, I) \oplus (Y, g, J) = (X \cup J, f \cup g, I \cup Y)$$

An important special case of inverse composition is cyclic composition of named sets.

If $X = (X, r, Y)$ and $Y = (Y, q, X)$ are two named sets, in which the support of X coincides with the reflector of Y and the support of Y coincides with the reflector of X, then their *cyclic composition* has the form

$$\mathbf{X} \copyright \mathbf{Y} = (X, r, Y) \copyright (Y, q, X) = (X, r \circ q, X)$$

If $X = (X, r, Y)$ and $Y = (Y, q, Z)$ are two named sets, then their *chain composition* is a named set chain (Burgin, 2011) and has the form

$$V = [\mathbf{X}, \mathbf{Y}]$$

## 3. Interpersonal Communication

People understand communication either as a process of information exchange or as a result of such a process. In addition, communication can include exchange of ideas, thoughts and/or opinions. However, everything that is transmitted in communication comes through information exchange. As a result, it is natural to treat communication as a system of information transmissions, which can be organized in a sequence or go concurrently. The action of information transmission has the structure of a basic named set (6), in which its support and reflector have the roles of a sender and receiver.

$$\text{Sender (Source) o} \xrightarrow{\;t\;} \text{o Receiver (Sink)} \tag{6}$$

Communication as a pure exchange of information in the form of messages has the structure (7) of a decomposed bidirectional named set, in which the naming relation represents messaging.

$$\text{Sender/Receiver o} \underset{g}{\overset{f}{\rightleftarrows}} \text{o Sender/Receiver} \tag{7}$$

Here each participant acquires the two-folded role of a Sender/Receiver.

Note that both the Sender and Receiver can be not only individuals but also groups of individuals, devices, e.g., computers, birds, animals and other living beings.

Connections *t*, *f* and *g* between the Sender and the Receiver have three components:

- *Communication space*, e.g., a channel or a system of channels, is the medium, in which communication goes and which allows sending messages from the sender to the receiver
- A system of *messages*, e.g., one message, where a message is the object sent (transmitted)
- A *context* consists of conditions (environment) in which communication goes

Messages are carriers of information and usually have three components:

- The *physical component* of a message, e.g., electrical signals or piece of paper with some text.
- The *structural component* of a message, e.g., text or picture
- The *mental component* of a message, e.g., the meaning of a text

Contexts usually have one of the following three types:

- *Individual contexts*
- *Group contexts*
- *General contexts*

A context determines how information is transmitted and also have three components:

- The *physical component* of a context, e.g., conditions, in which information is sent and received
- The *structural component* of a context, e.g., language of the message
- The *mental component* of a context, e.g., knowledge of the sender and receiver

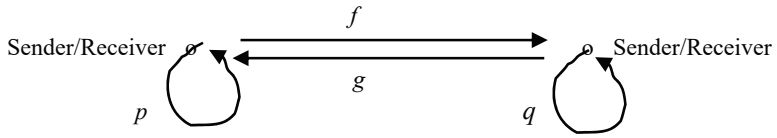
Besides, there is such a phenomenon as self-communication. Self-communication is communication self-directed in the elaboration and sending of the message, self-selected in the reception of the message, and self-defined in terms of the formation of the communication space. Self-communication is special case of intrapersonal communication. Self-communication is represented by the following cyclic named set (8), in which X is both the Sender and the Receiver.

$$\overset{X}{\underset{f}{\circlearrowright}} \tag{8}$$

Another important case of intrapersonal communication represented by a cyclic named set is communication in networks, such as the Internet, where each node can be both a receiver and sender. This process is also naturally modeled by a cyclic named set, in which X is the whole network.

A combination of network communication and self-communication is called mass self-communication.

Enhanced communication as an exchange of information with information processing, which includes information organization, is the sequential composition (9) of two cyclic and one bidirectional named sets.



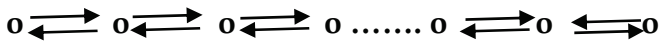

(9)

According to the general theory of information, *information* for a system *R* is a capacity to change an infological system IF(*R*) of the system *R*.

There are three basic forms of information organization:

- *Quantization* by determining units of information and then measuring or counting these units
- *Qualification*, in which information is represented in an explicit form pertinent to the problem or situation, e.g., by modeling or describing
- *Categorization*, e.g., classification or clustering

All considered above types and schemas represented *direct communication*. *Mediated communication* has the structure of a chain (10) of bidirectional named sets (fundamental triads):



(10)

In a *sequential communication*, all messages are linearly ordered in time and transmission of each of them does not intersect in time with transmission of another one.

There are different types of messages in communication:

– The *starting message* $m_0$ is the first message in communication.
– The *concluding message* $m_t$ is the last message in communication.
– A *feedback message* is an outcoming message caused by some incoming message.
– A *direct feedback* message is an outcoming message caused by the previous incoming message.
– An *initializing message* is a message that is not a feedback message.

Messages allow construction of characteristic structures of communication.

*Communication thread L* consists of an initializing message called the *root* of *L* and a sequence of direct feedback messages, in which the first element is the feedback to the initializing message and each next element is the direct feedback to the previous element.

*Communication leaping thread H* consists of an initializing message called the *root* of *H* and a sequence of direct feedback messages, in which the first element is the feedback to the initializing message and each next element is the feedback to the previous element but not necessarily direct feedback.

Note that a communication thread is also a communication leaping thread.

*Communication multithread* is the union of communication threads with a common root.

*Communication leaping multithread* is the union of communication threads, which have a common root and can be leaping.

Note that a communication thread is also a communication multithread and a communication multithread is also a communication leaping multithread.

*Communication hyperthread* is the union of communication threads.

*Communication leaping hyperthread* is the union of communication threads, some of which can be leaping.

This gives us different types of communication.

– Communication is *linear* if it consists of a single communication thread.
– Communication is *branching* if it contains, at least, one communication multithread.
– The most general is *concurrent communication*.

Two communication threads are *disjoint* if each of them does not have a feedback message to a message from another one.

Communication is *disjoint* if it consists of disjoint communication threads.

Explication of structural peculiarities of communication is aimed at better organization of interpersonal communication in both human society and networks of artificial devices such as cell phone networks or the Internet.

**Conflicts of Interest**: The author declares no conflict of interest.

## References

1. Harrah, D. *Communication*: *A Logical Model*; Cambridge University Press: Cambridge, UK, 1967.
2. Keltner, J.W. *Interpersonal Speech-Communication*: *Elements and Structures*; Wadsworth P.C.: Belmont, CA, USA, 1970.
3. Burgin, M.; Neishtadt, L. *Communication and Discourse in Teachers Professional Activity*; Daugavpils Pedagogical Institute: Daugavpils, Latvia, 1993.
4. Burgin, M. *Theory of Named Sets*; Mathematics Research Developments, Nova Science: New York, NY, USA, 2011.
5. Burgin M. Theory of Named Sets as a Foundational Basis for Mathematics. In *Structures in Mathematical Theories*; D. Reidel Publishing Company: San Sebastian, Spain, 1990; pp. 417–420. Available online: http://www.blogg.org/blog-30140-date-2005-10-26.html.
6. Burgin, M. Named Sets as a Basic Tool in Epistemology. *Epistemologia* **1995**, *XVIII*, 87–110.
7. Dalla Chiara, M.L.; di Francia, T.G. Individuals, kinds and names in physics. In *Bridging the Gap*: *Philosophy, Mathematics, Physics*; Kluwer: Dordrecht, The Netherlands, 1993; pp. 261–283.
8. Barr, M.; Wells, C. *Toposes, Triples, and Theories*; Grundlehren der Math. Wissenschaften, v. 278; Springer: Berlin, Germany, 1985.
9. Wadler, P. Comprehending monads. *Math. Struct. Comput. Sci.* **1992**, *2*, 461–493.
10. Berge, C. *Graphs and Hypergraph*; North Holland P.C.: Amsterdam, The Netherlands; New York, NY, USA, 1973.
11. Close, F. *The Infinity Puzzle*: *Quantum Field Theory and the Hunt for an Orderly Universe*; Basic Books: New York, NY, USA, 2011.

© 2017 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

information

MDPI

Article

# Information Ecology in the Context of General Ecology

**Mark Burgin and Yixin Zhong**

1 Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA
2 School of Computer, University of Posts and Telecom, Beijing 100876, China; zyx@bupt.edu.cn
* Correspondence: mburgin@math.ucla.edu; Tel.: +1-310-251-4897

Received: 16 January 2018; Accepted: 24 February 2018; Published: 8 March 2018

**Abstract:** The ecological approach studied in this paper is a new level of information studies. It allows for achieving a better understanding of information processes in society as well as more efficient creation of information processing systems. At first, in Section 2, we describe and analyze ecological studies in different areas ranging from biology to technology to sociology to knowledge and information. Then, in Section 3, we present elements of general ecology building methodological and philosophical foundation for information ecology. In Sections 4 and 5, we elaborate a concise definition of information ecology and further develop information ecology as a methodological base for information studies in general based on the concepts and principles of the general theory of information.

**Keywords:** information; ecology; process; environment; interrelations; subject information; object information; model; structure; pattern

## 1. Introduction

Information is a very complex phenomenon. It is possible to study information on several levels.

The first level is oriented at information as a fundamental phenomenon in nature, society and technology exploring properties and relations of information. Examples of theories from the first level are the general theory of information [1] and qualitative information theory [2]. The fundamental essence and intrinsic nature of information is of the primary importance for scientific research. There have been many discussions and various suggestions related to these issues. The most advanced and comprehensive answer is obtained in the general theory of information.

The research regarding the second level investigates information processes. Shannon's information theory [3,4], theories of information flow developed by Dretske, Barwise and Seligman [5–7] and operator information theory [8,9] are examples of theories from the second level. Dynamics of information has prevalent significance for contemporary society because humankind came to the information era.

On the third level, researchers investigate information systems and processes going in these systems making emphasis on the products and services of information process and the interrelations between information and the products/services in the information environment. The ecological approach to information belongs to this level [10–13].

The research in information ecology is based on the global view on existing interrelations between information, knowledge, data and information processing systems. Ecological approach in information studies presupposes synthesis of knowledge and cognition from the first two levels developing the most comprehensive and wide-ranging picture of information reality.

This paper is aimed at building foundations of information ecology using principles of general ecology developed by Burgin [14] because any science in general and information ecology, in particular, need sound and flexible foundations.

It is necessary to understand that there are many interesting and important problems related to studies of information. However, because information is one of the basic phenomena of the world, everything is more or less related to information. The scope of information ecology does not include all phenomena of the world. By the same token, celestial mechanics studies dynamics of planets but tells nothing about people living on one of these planets. Plant ecology does not discuss atoms and molecules although all plants consist of atoms and molecules. This is the scientific approach when each science has its own domain and we build information ecology as a scientific discipline.

That is why, for example, here we do not discuss the diversity of definitions of information suggested by different researchers but base our work on the axiomatic definition from the general theory of information as the most advanced and encompassing theoretical model of information [1].

## 2. A Brief History of Ecological Studies

The basic for ecological studies concept of ecosystem appeared in 1864 in the book "Man and Nature" by George Perkins Marsh. However, according to [15], the term ecosystem was derived much later by Arthur Roy Clapham by Tansley's request [16].

German scientist Ernst Haeckel (1834–1919) coined the term *ecology*, or *Ökologie* in German, in 1866 constructing it from two Greek words. The first of them *oikos* is in English is translated as *house*, or more generally, *habitat* or place of living. At the same time, the second word, *logos* was used by ancient Greeks signifying such concepts as *order*, *meaning*, *foundation* or *mind* [17]. Haeckel's approach encouraged European botanists to study plant populations related to definite areas and their interdependencies. This gave birth to the early science of ecology, which was studying not only plants but also animals and other living beings. Later this field has become biological ecology comprising more living organisms.

In the contemporary context, ecology is mostly understood as a holistic exploration of living systems with their connections to their environment by finding patterns of, processes in and interrelationships between these systems and their surroundings.

It is important that ecology, or more exactly, biological ecology, as a whole contains two subfields—plant ecology and animal ecology.

*Plant ecology* explores the allocation and profusion of plants, the impact of environmental factors upon systems of plants, and the interactions among and between plants and other organisms [18].

*Animal ecology* is a scientific field that explores the allocation and profusion of animals interacting with each other, as well as with their surroundings, which controls the dispersal and abundance of living organisms.

Together these two scientific fields constitute *natural ecology*, or *biological ecology*, which is often simply called ecology. Later researchers also created other ecological disciplines. One of the most basic is *human ecology*. It is defined as an interdisciplinary and transdisciplinary research of the relationships between humans and their natural, social, and technological surroundings. It engages an assortment of disciplines such as sociology, anthropology, psychology, zoology, public health, epidemiology, home economics, geography, and natural ecology, among others [19].

In contrast to traditional ecology, which is dealing only with natural systems, the new field of *industrial ecology* emerged. It studies industrial manufacturing as part of larger organizations and processes, which include industrial functioning and biogeochemical reactions as a part of an arrangement, and aims at reduction of the negative environmental effects of manufacturing, expenditure, and disposal.

There is also *business ecology*, which studies business organizations in the context of *business ecosystems*. Theyare dynamic networks of mutually dependent organizations that rely on each other. Such components as compartments separated by the organization's walls, distribution channels and direct suppliers are investigated in business ecology. Other constituents that can have a significant effect

on the core business such as trade associations, direct customers, standards bodies, regulatory bodies, suppliers of complementary products, unions, investors, are also included in business ecosystems.

Yixin Zhong initiated studies of *information ecology* in China [10]. Independently, Western researchers started their exploration of information ecology [12,13]. Information ecology has the foremost importance for information area as a holistic methodology to knowledge acquisition about the nature and behavior of information processing systems, as well as for improved comprehension of information processes in all realms of the world. If ecology of animals studies the processes and interrelations in systems of animals, information ecology investigates the processes and interrelations in associations of information processing systems and configurations.

Researchers also study *knowledge ecology* [20–23]. It is a methodology of knowledge management, which is aimed at advancing the dynamic development of knowledge contacts between organizations to improve decision-making and innovation using superior evolutionary networks of cooperation. In comparison with the merely directive-oriented management, which tries either to administer or to control outcomes, knowledge ecosystems promote knowledge strategies that concentrate more on enabling adaptable self-organization and self-improvement reacting to changing surroundings.

Introduction and utilization of the concept *digital ecosystem* instigated the development of *digital ecology* [24]. A digital ecosystem is a distributed, flexible, adaptive, amenable socio-technical system with features of self-organization, self-operation, scalability and sustainability motivated by the concept of natural ecosystems.

Specialization of the concept digital ecosystem brought forth digital health ecosystems [25,26], digital service ecosystems [27], digital transportation ecosystems [28] and digital business ecosystems [29]. The latter are defined as combinations of digital systems used in the business, the people that interact with them, and related business processes and technology environment. A digital business ecosystem emerges in coupling of digital business ecosystems to the socio-economic system of its users [30].

Mark Burgin introduced and studied *techno-ecology* [31].

Damiani, Uden and Trisnawaty introduced E-learning ecosystems [32].

Gregory Bateson introduced *ecology of mind* [33].

In a similar way, American anarchist and libertarian socialist author Murray Bookchin originated the field called *social ecology*, which is a critical study of the existing society [34].

## 3. Fundamentals of General Ecology

The fundamental concept of *ecosystem* (*ecological system*), which is pivotal for different ecological disciplines, waspropposed by the English ecologist Arthur Tansley in the context of natural ecology. Here we define this concept in a much broader context comprising the existing diversity of ecological disciplines. With this goal in mind, we demonstrate the impact of the global structure of the world on the organization of ecosystems.

There are different models of the world where people live. Some assume that only material (physical) reality exists. Others add to it individual mentality called Mind. Here we based our study on the most advanced large-scale structure of the world, which has the form of the *Existential Triad* [1]. It is given in Figure 1 being scientifically elucidated in the book [35].

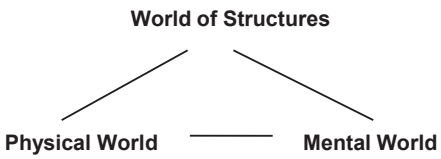



**Figure 1.** A graphical representation of the Existential Triad of the World.

The three worlds, which constitute the Existential Triad, are not disconnected realities: they interact and traverse. The mental world includes individual mentality. It is based on the brain, which is

a material thing, but includes mind. Mind-Brain relations have been discussed by many philosophers in the context of the mind-body problem and there is no mutual consent on this problem. However, here we are not going to discuss all different approaches and opinions in this area because this is not the goal of this paper. Besides, it is necessary to understand that this problem belongs to the second level of information and cognitive studies while information ecology bring these studies on the higher, third level. At the same time, many physicists think that mentality impacts physical world (cf., for example, [36]). This is only a working hypothesis but it is definitely established that our knowledge of the physical reality principally depends on interface between mental and physical worlds. Note that in the context of the Existential Triad of the World, there is no fundamental distinction between material and physical worlds.

However, not only people but also all forms of information processing systems have their specific mentality. For instance, the content of the computer's memory can be logically considered as the inherent mentality of this computer. Thus, the operating system or word processor is a basic part of the mentality of the computer.

The World of Structures provides a scientific embodiment of Plato's World of Ideas/Forms because ideas or forms are properly associated with structures [37]. As it is demonstrated in [35], existence of structures is validated by observations and experiments in a similar way as it is done for material things, such as tables, chairs, or buildings. As a result, structures form the *structural level* and *component* of the world. When it is necessary to, it is done, as a rule, by means of knowledge of structures is vital for learning or creating systems and processes. Structures, which also include different interrelations and geometric forms, shape material things in their existence and comprehension.

Ecological approach allows explication of homological processes in the physical and mental worlds. In the physical world, the growth of organization and complexity goes through the following stages:

**Energy** begets **matter**, while **matter** begets **life**

In the mental world, the growth of organization and complexity goes through the following stages:

**Information** begets **knowledge**, while **knowledge** begets **intelligence**

These stages form two triadic structures:

$$\textbf{energy} \xrightarrow{\text{begets}} \textbf{matter} \xrightarrow{\text{begets}} \textbf{life} \tag{1}$$

$$\textbf{Information} \xrightarrow{\text{begets}} \textbf{knowledge} \xrightarrow{\text{begets}} \textbf{intelligence} \tag{2}$$

In more detail, these relations are validated, studied and explained in the book [1].

The global structure of the world in the form of the Existential Triad brings on three categories of ecosystems:

- *Physical ecosystems* comprise physical systems, arrangements and processes as its elements, parts and components
- *Mental ecosystems* encompass mental schemas, arrangements and processes as its elements, parts and components
- *Structure ecosystems* consist of structures of physical systems, arrangements and processes as its elements, parts and components

For instance, a mathematical model of a material ecosystem is a structure ecosystem, while a mental model of a material ecosystem is a mental ecosystem. Other examples of structure ecosystems are mathematical models of scientific theories when the processes going in these systems are included in the model (cf. [38–41]).

Combining all three components of the world stratification in one system, we come to a *total ecosystem*. Total ecosystems have three basic constituents:

– The *physical constituent* of the ecological system and its environment comprises physical systems, arrangements and processes
– The *structural constituent* of the ecological system and its environment consists of structures of physical systems, arrangements and processes
– The *mental constituent* of the ecological system and its environment encompasses mental schemas, arrangements and processes

A general ecosystem is determined by three characteristics:

– A section in the physical (mental or structural) space (or a region on a scientific sphere), i.e., it is presumed that all elements and components of an ecological system belong to a definite section in the space (or a definite region on a scientific sphere).
– The key categories of its elements, parts and components, i.e., it is determined what elements, parts and components of the specified ecological system are treated as the most imperative from the perspective of ecological studies.
– The principal sorts of connections and interrelations between its elements, parts and components, including processes as dynamic connections, i.e., it is determined what relationships, ties and processes in the specified ecological system are treated the most basic from the perspective of ecological studies.

For instance, living beings constitute the primary class of elements in natural ecosystems and a selected area on the Earth molds the section in the space of these ecosystems. In this framework, a natural ecosystem is built of the dynamically interacting constituents including primary living beings in a given area, which interact with each other and with their non-living surrounding.

To formally define an ecosystem, we consider three classes:

– A class *R* of regions in the space, for example, forest region or desert region on the Earth
– A class *E* of elements and components, for example, animals living in the region, plants growing in the region and abiotic elements, such as rivers or mountains, in the region
– A class *C* of ties and relations between its elements/components including processes as dynamic ties

**Definition 1.** *An (R, E, C)-ecosystem is a system of elements from E, which belong to a region r from R and tied/interrelated in the system by connections from C.*

Information processing systems constitute the primary type of elements in an information ecosystem and a selected area on the Earth (may be the whole Earth) forms the section in the space where information processing systems are interacting with each other as well as with their surroundings. Besides, investigation of information ecosystems contemplates information processes going in or related to the system.

It is necessary to remark that there are various kinds and sorts of information processing systems such as living information processing systems, technical information processing systems, human information processing systems, digital information processing systems and so on.

There are three grades of (types of) parts, components and elements in information ecosystems:

- *Primary* or *leading* parts, components and elements
- *Secondary* or *auxiliary* parts, components and elements
- *Tertiary* or *background* parts, components and elements

Ecological studies in all fields are oriented at understanding existence and behavior of the primary parts, components and elements of ecosystems, as well as critical connections, ties, interrelations and processes in considered ecosystems.

Primary and other elements of ecosystems belong to populations of similar entities. For instance, in natural ecosystems, elements belong to populations of human beings, animals, plants and viruses.

## 4. Methodological Issues of Information Studies

It is well-known that information is a category of critically important resources for human beings as well as for all kinds of living beings. As a result, humans have information processing and utilization organs, which are able to handle information: (1) the sensing organs receive information from the environment; (2) the nervous system including the brain processes and transmitting information from one point to another within human body producing knowledge and intelligent strategies for problem solving; and (3) action organs, which are controlled by information coming from the nervous system, execute strategies developed in the brain. In addition, people have created an abundance of technical information devices and instruments for enhancing functions of human information organs in information handling. Examples of such systems are different kinds of sensors, communication systems and networks for information transmission and distribution, computer systems for information processing, artificial intelligence systems producing knowledge and making decisions, control systems for strategy execution, and so on.

However, there is contrast between functioning of the human information organs and operations in technical information systems. On the one hand, assuming a satisfactory state of the whole organism, human information organs have always been working effectively and smoothly with well-balanced interrelationships among them continuously facilitating the development of people. On the other hand, various kinds of technical information systems constructed for enhancing human information organs were not able to achieve sufficient harmony between technical systems, human beings and the environment.

The problem lies in establishing relevant connections between technical information systems. At the beginning, many such systems, e.g., computers were designed without interconnections between them. Later with the advance of local and global computer networks such as the Internet and local networks, technical information systems have become more and more interconnected but they still did not achieve harmony and efficiency of information processing system of human beings.

## 5. Ecological Approach as a New Methodology for Information Studies

Let us consider information ecology as a methodology for information studies. It will allow us to understand what progresses could be achieved based on this methodology and what opportunities can be gained in applications.

### *5.1. The Concept of Information Ecology*

As usual, the definition for a new terminology is necessarily needed first so that the understanding and the application of the terminology in the future can be precise as well as reliable. The definition of information ecology shall, of course, keep the spirit of the definition of the general ecology.

**Definition 2.** *Information ecology is a holistic study of information processing systems in the context of their surroundings by explicating patterns of, processes in and interrelationships between these systems and their components in the context of their environment.*

The efficacy of this definition is demonstrated by application of ecological principles in other areas, such as plant ecology, animal ecology and other kinds of ecology considered in Section 2. For instance, importance of relationships between the system and its environment is well explained in the book [17] and it is not necessary to repeat those arguments here.

Defining information ecology, we do not define information or information processing because the definition of information and information processing belongs to Level 1 of information studies, while information ecology is Level 3 of information studies. That is why in this paper, we do not discuss different approaches to the definition of information but take as the basic the definition of information from the general theory of information, the most detailed exposition of which is given in the book [1].

From the perspective of scientific research, there are two basic approaches to the studied phenomena—reductionism and holism. The major feature of reductionism is dividing a complex system into elements and reducing the theory of the system to the theory of its elements. Holism goes in the opposite direction assuming that basic properties and functioning of a complex system cannot be deduced from properties and behavior of its elements.

Ecological theory in general and information ecology in particular base their studies on holistic approach because relations, connections and processes uniting separate elements in an integrated system produce synergy making the whole system irreducible to its elements. Ignoring these regularities of system science can cause various misconceptions and result in waste of time and energy in human society.

### *5.2. Information Ecology as a Research Model*

A critical issue of any research in general and information studies, in particular, is correct selection of an adequate basic structure for representation, modeling and exploration of the research domain. An important innovation of information ecology is introduction of the triadic structure "subject–object-interaction" as the basic system of information studies [42,43]. This structure is described in Figure 2.

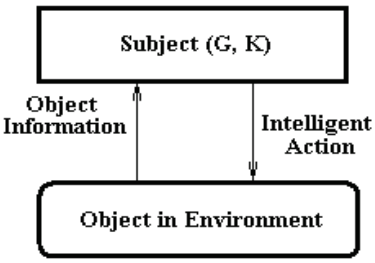



**Figure 2.** The triadic informational structure "subject–object-interaction".

The structure "subject–object-interaction" consists of three components:

- *Subject* can be any intelligent system possessing knowledge and goals and treated from the information perspective although the most typical, most characteristic, and thus, the most meaningful, subjects are human beings who have complex goals and enough knowledge.
- *Object* can be any information processing system imbedded in its environment.
- *Interaction* consists of processes that go between the Subject and the Object.

Triadic structures, such as "subject–object-interaction" are very important. As Lao Tse writes:

*The Way produces one;*
*one begets two;*
*two begets three;*
*and three brings about the whole world.*

Lao Tse, *Tao TeChing*

In a more general context, the structure "subject–object-interaction" is an important kind of bidirectional fundamental triad, which is also called a bidirectional named set [44,45]. Identifying

"subject–object-interaction" as a fundamental triad or a named set allows using mathematical theory of named sets developed in [44] for modeling and exploration of cognitive processes.

This shows fundamentality of "subject–object-interaction" because it is proved [44,46,47] that fundamental triads constitute the most critical structure in nature, mathematics, society and cognition supporting the insight of Lao Tse. In addition, it is demonstrated that fundamental triadsare indispensable for a diversity of domains including information theory, mathematics, chemistry, physics, networks and networking, logic, AI, database theory and practice, mathematical linguistics, biology, epistemology, philosophy and methodology of science, to mention a few.

In the graphical representation (cf. Figure 3), a *basic fundamental triad* or a *basic named set* has the following appearance:

$$A \xrightarrow{\quad c \quad} B$$

**Figure 3.** A basic fundamental triad or a basic named set.

In the symbolic representation, a basic fundamental triad has the form $\mathbf{X} = (A, c, B)$ where $A$ and $B$ are two objects and $c$ is a tie or correspondence (e.g., a binary relation) between $A$ and $B$. In fundamental triad $\mathbf{X}$, the object $A$ is called the *support* of $\mathbf{X}$, the object $B$ is called the *component of names* (*reflector*) or *set of names* of $\mathbf{X}$, and $c$ is called the *naming correspondence* (or *reflection*) of the triad $\mathbf{X}$. It is necessary to understand that in the fundamental triad $\mathbf{X}$, $c$ is not always a function or a mapping.

In the typical example of a basic named set (fundamental triad), the object $A$ consists of people, the object $B$ consists of their names and $c$ is the connection between people and their names. Another example is a basic named set (fundamental triad) the object $A$ comprises books, the object $B$ comprises their titles and $c$ is the association between books and their titles.

In the graphical representation (cf. Figure 4), a *bidirectional fundamental triad* or a *bidirectional named set* has the following appearance:

$$A \xleftrightarrow{\quad c \quad} B$$

**Figure 4.** A basic fundamental triad or a basic named set.

In the symbolic representation, a bidirectional fundamental triad has the form $\mathbf{X} = (A, c, B)$ where $A$ and $B$ are two objects and $c$ is a tie or correspondence (e.g., a binary relation) between $A$ and $B$ which goes in two directions.

An important example of a bidirectional named set (a bidirectional fundamental triad) is given by two people who are communicating, i.e., exchanging messages, e.g., by e-mails, using messaging or talking to one to another. In this bidirectional fundamental triad, $A$ and $B$ are people while $c$ consists of messages going from one person to another.

In situations when mathematicians, physicists or computer scientists utilize connections or ties without definite direction, such as edges in general graphs, in essence, these ties have both directions as in a bidirectional fundamental triad.

Thus, we can see that the structure "subject–object interaction" (see Figure 2) is in fact a comprising model in the world of information providing efficient means for information studies in general and information ecology as their methodology.

The triad "subject–object interaction" explicates the situation when the object contains information for the subject and it is natural to call it "object information". In the process of interaction, the subject, having received the object information, often acts on the object physically, structurally or/and mentally.

The action produced by the subject should be sufficiently intelligent to avoid certain risks. Intelligent action must satisfy two conditions:

(1) It has to be aimed at achieving the subject's goal;
(2) The interaction of the system and the environment should be harmonious.

Another point worth of mentioning is that the subject in the triad "subject–object interaction" is not necessarily a single individual person. Instead, it may be a group people, an organization, or even the entire society as long as it has the common goal and knowledge.

While the model of information studies based on the triad "subject–object interaction" is fundamental, it is extremely abstract. To be closer to the scientific research in information discipline and even more to information processing practice, it must be specified in more detail giving an expanded picture and efficient structure of information processing systems and their dynamics in an adequate environment.

Living beings, such as people, animals and plants, consume energy, for example, energy coming from the Sun, and material things, such as water or air, with the goal to produce energy and substance for sustaining themselves and producing other material things. In a similar way, information processing systems consume information and structural objects, such as data and knowledge for producing new information and knowledge and achieving intelligence. These processes provide for the construction of the next level of the model for information studies, which is presented in Figure 5 [42].

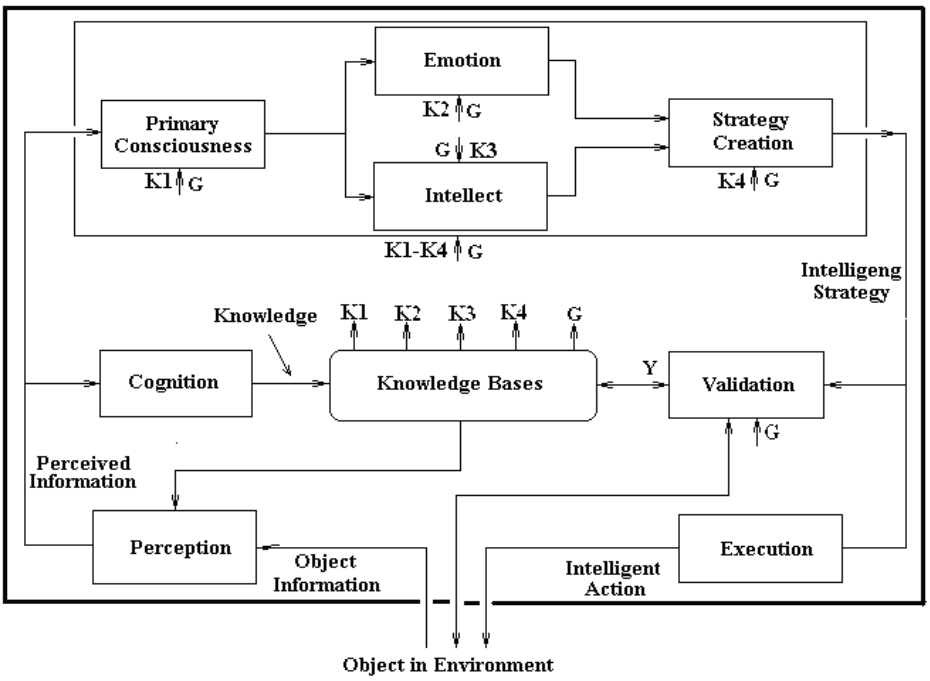



**Figure 5.** A detailed model for processes in information studies.

As we can see, the model in Figure 5 can be treated as an expansion of the model in Figure 2 by specification of the subject and processes in the triad "subject–object interaction". As a result, we have the following processes:

(1) The object information is converted to perceived information by perception mechanisms.
(2) The perceived information is transformed into knowledge by the mind.
(3) The perceived information, complemented by the related knowledge and directed by subject's goals, is successively generates consciousness, emotions, and intellect, which are integrated in the brain and in turn, generate intelligent strategy.

(4) Intelligent strategy is converted to intelligent action via execution mechanisms.
(5) The created intelligent action is finally applied to the object, forming the first cycle the subject–object interaction.

We call this process the *perception-processing-action cycle* because it is repeated in intellectual activity of people.

If the result caused by the intelligent action differs from the goal, the subject receives information about the error and repeats the perception-processing-action cycle. This cycle is also repeated for obtaining more knowledge, optimizing the strategy and getting a better result with a smaller error and/or decreasing consumption of resources.

The model for information studies presented in Figure 3 indicates that information discipline is a large extremely complex system involving diverse human information activities.

It is necessary to remark that there are three levels of perception.

The first level is *reception* of information transmitted from a sender to a receiver. On this level, only information transmission without information processing takes place.

The second level is *perception* of information that comes from a container/source of information to the recipient. On this level, information transmission is followed by information processing.

The third level is *cognition* when the cognizer extracts/obtains information from a container of information. On this level, information extraction involves essential information processing and production of new information.

### *5.3. Ecological Approach in Information Studies*

After elaborating a general model for information studies on the third level considered in Introduction, we develop a *research technological approach* based on this model.

There have been a number of different approaches adopted in information study: structural approach, functional approach, and behavioral approach, to name a few. Each of them has made some progress while also has faced challenges. Which one would be the most appropriate to the information study?

As is stressed above, the focus of the information ecology as methodology is the study of the interrelation between information processing systems within an organization. From the model of information discipline in Figures 2 and 5, it is possible to see that the essence of information processes is the information flow, which converts object information into perceived information proceeding to knowledge, elaborating an intelligent strategy for action and finally, resulting in intelligent actions. This structure forms the basic loop of the flow, which constitutes the lifeline of information processing system. The crucial problem for information processing system is how to construct a mechanism via which the information flow can successfully be realized, controlled and operated.

As a matter of fact, the role that the *structure* and the *function* of an information processing system can play is to serve for, and support to, the mechanism of the system so that the information flow can be realized while the *behavior* of the system is the outcome of the mechanism implementation. The soul of the flights is the principle of air dynamics, not the specific structure of the bird. Similarly, the soul of information processing systems is the mechanism with which the information flow of a system can be produced, instead of the structure, functions, or behavior of the system itself [48].

It is clear from the performed analysis that the approach, which consists of series of information conversions, is the most appropriate approach to information studies. This is the essence of information ecology as a methodology in the information discipline.

### *5.4. Basic Relations and Processes in Information Ecological Systems*

Based on the described guidelines of information ecology as a methodology and the model for the study of information discipline, it is possible to explicate several important categories of interrelations in the information ecological model [49].

Here we analyze interrelations between object information and perceived information, which form the first category of basic interrelations according to information ecology. Our analysis is based on the axiomatic definition of information elaborated in the general theory of information [1].

For a long time, most researchers thought that human sensing organs and technical sensors perform the corresponding conversion of object information into perceived information. Nevertheless, this is not true. For instance, the concept of information defined in Shannon's information theory does not provide a comprehensive representation of perceived information, but reflects only one of its components. Therefore, it is necessary to clarify what is object information and what is perceived information before doing other things.

**Definition 3.** *Object information is information contained in an object.*

It is obvious that every object in reality is producing and presenting its object information all the time. Object information is a kind of pervasive phenomenon and is determined only by the object, and has nothing to do with the subject's factors. This is why that object information is sometimes named as ontological information.

**Definition 4.** *Perceived information is information perceived by the subject, i.e., received, accepted and preprocessed.*

Perceived information is also called *epistemological information*.

As we can see from this definition, the perceived information has three components: syntactic information related to the form of the object, semantic information related to the meaning (or content) of the object, and pragmatic information related to the utility of the object with respect to the subject's goal. For any subject, the three mentioned components form a triad. It is rare in reality, often even impossible, that a subject in general and human subject, in particular, is interested only in the form of an object without taking into account its meaning and utility.

Thus, what are interrelations between the object information and perceived information?

The answer can be found by utilization of the model from Figure 5, which shows how three components of perceived information are produced by human beings and in the components of other intelligent systems.

As we can see from Figure 6, object information is applied to subject and then the subject produces perceived information expressed with its three components in order: syntactic, pragmatic, and semantic information successively. This schema is essential for all kinds of cognitive processes.

More specifically, there are three steps in human beings and other information processing systems for conversion of the object information to perceived information (cf. Figure 4):

(1) Syntactic information production

The syntactic information on the lexical level is first produced by sensing organs information preprocessing and is formed in sensual data. After this production, obtained data, which contain syntactic information, are organized in higher syntactic structures by preprocessing.

(2) Pragmatic information production

(2-1) If the subject has experienced the object before and pragmatic object information has been stored in the subject's memory as {syntactic ($n$), pragmatic ($n$)} where $n$ is the name of the object, then it is very easy to produce pragmatic information via recalling, that is, by using the syntactic information obtained at step (1) in the same way as key words are used for retrieving information from the memory. As long as the "syntactic ($n$)" is matched with the key words, pragmatic ($n$), which is related to syntactic ($n$), is pragmatic object information.

(2-2) If the object is a new one and the subject has never experienced it before having no information in the form {syntactic ($n$), pragmatic ($n$)} stored in memory. Then there is no possibility to produce the pragmatic information through recalling. In this case,

the pragmatic information can be produced via direct testing—making a real experiment to see if the object has any utility related to the subject's goal.

(3) The semantic information production

According to [10], subject's semantic information about the object is the subject's understanding of the object's meaning. So, semantic object information can be produced by the process of abstracting the system {syntactic ($n$), pragmatic ($n$)}, which is established in steps (1) and (2), into the semantic space.

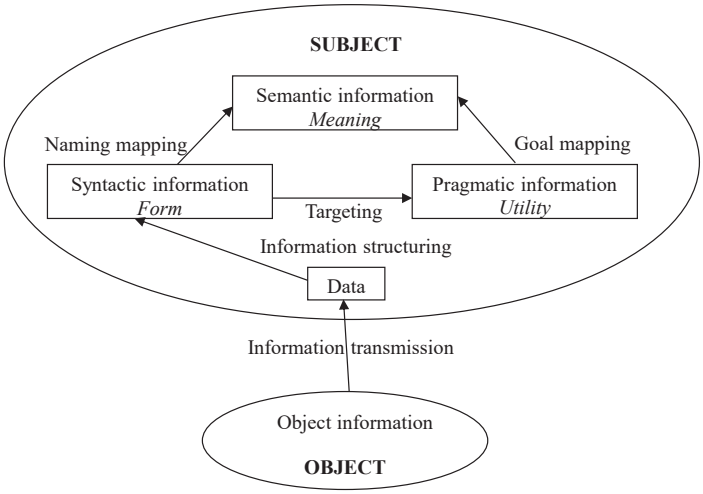



**Figure 6.** Conversion of object information into perceived information in intelligent systems.

As a result, after the steps (1), (2), and (3) were successively performed, all three components of perceived information—syntactic information, pragmatic information, and semantic information—have been produced and available. This shows that interrelations between object information and perceived information are properly described by three steps, which are described above.

It is interesting to note that steps (1) and (2) are easy to understand whereas step (3) is much more complicated. This is caused by the fact that both syntactic information and pragmatic information are specific and thus, can be produced directly using sensing organs and recalling whereas the semantic information is abstract in nature and can only be produced via the abstraction of both syntactic and pragmatic information.

Note also that if syntactic information is denoted by the symbol X, pragmatic information by the symbol Z, and semantic information by the symbol Y, then according to Figure 6, the essence Y can be derived from X and Z, which is possible to represent by the following expression:

$$\mathrm{Y} = \lambda(\mathrm{X}, \mathrm{Z}) \tag{1}$$

The symbol λ in Expression (1) stands for an operator of *abstraction* and *naming*, which are basic operation in the theory of named sets [44]. In fact, Expression (1) can be regarded as a strict definition of semantic information improving understanding that existed before.

In addition, Expression (1) implies that semantic information can represent the union of both syntactic and pragmatic information. Consequently, whenever semantic information is obtained, syntactic and pragmatic information are also obtained. This explains why people are mostly concerned with semantic information and less with syntactic and pragmatic information.

As a matter of fact, there is already large number of works related to and discussing syntactic, semantic and pragmatic information in the context of the research field called "semiotics". Unfortunately, they pay too little attention to the study of relationships between the three types of information in spite of their fundamentality and importance for semiotics.

Information studies have paid more attention to syntactic, semantic and pragmatic information. In information theory, there are developed theories of these types of information (cf., [1]) although the main emphasis is, as a rule, made on one type of information. Information ecology aims at systemic exploration of systems that include syntactic, semantic and pragmatic information as their subsystems.

## 6. Conclusions

We have discussed advantages of information ecology as a methodology for information studies in the context of general ecology. The background, the concept, the progresses achieved, and the significance related to the new methodology have been described.

At the same time, one paper cannot include everything from the scope of information ecology. Other problems are considered elsewhere. Note that having understood the model expressing the interrelation between object information and perceived information in humans in Figure 4, it is essential to investigate whether the interrelation between object information and perceived information can be implemented technically. This will be done in subsequent publications of the authors.

**Acknowledgments:** Some of the ideas and results of this paper were presented at the International Summit on the Studies of Information held in Gothenburg, Sweden, 12–16 June 2017. The authors would like to express sincere thanks to the colleagues for their encouraging comments and discussions.

**Author Contributions:** Contributions of both coauthors are equal.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: New York, NY, USA, 2010.
2. Mazur, M. *Jakosciowa Teoria Informacji*; PAN: Warszawa, Poland, 1970. (In Polish)
3. Shannon, C.E. The Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [CrossRef]
4. Shannon, C.E. *Collected Papers*; Sloane, N.J.A., Wyner, A.D., Eds.; IEEE Press: New York, NY, USA, 1993.
5. Dretske, F.I. *Knowledge and the Flow of Information*; Basil Blackwell: Oxford, UK, 1981.
6. Dretske, F. Précis of Knowledge and the Flow of Information. *Behav. Brain Sci.* **1983**, *6*, 55–63. [CrossRef]
7. Barwise, J.; Seligman, J. *Information Flow: The Logic of Distributed Systems*; Cambridge Tracts in Theoretical Computer Science 44; Cambridge University Press: Cambridge, UK, 1997.
8. Burgin, M. Information Operators in Categorical Information Spaces. *Information* **2010**, *1*, 119–152. [CrossRef]
9. Burgin, M. Weighted E-Spaces and Epistemic Information Operators. *Information* **2014**, *5*, 357–388. [CrossRef]
10. Zhong, Y.X. *Principles of Information Science*; BUPT Press: Beijing, China, 1988. (In Chinese)
11. Zhong, Y.X. The Law of "Information Conversion and Intelligence Creation". In *Information Studies and the Quest for Transdisciplinarity*; World Scientific: New York, NY, USA, 2017.
12. Capurro, R. Towards an Information Ecology. In *Information and Quality*; Wormell, I., Ed.; Taylor Graham: London, UK, 1990; pp. 122–139.
13. Davenport, T.H.; Prusak, L. *Information Ecology*; Oxford University Press: Oxford, UK, 1997.
14. Burgin, M. Principles of General Ecology. *Proceedings* **2017**, *1*, 148. [CrossRef]
15. Willis, A.J. The Ecosystem: An Evolving Concept Viewed Historically. *Funct. Ecol.* **1997**, *11*, 268–271. [CrossRef]
16. Tansley, A.G. The use and abuse of vegetational concepts and terms. *Ecology* **1935**, *16*, 284–307. [CrossRef]
17. Odum, E.P. *Fundamentals of Ecology*; Cengage Learning: Boston, MA, USA, 2004.
18. Weaver, J.E.; Clements, F.E. *Plant Ecology*; McGraw-Hill Book Company: New York, NY, USA, 1938.
19. Young, G.L. Human ecology as an interdisciplinary concept: A critical inquiry. *Adv. Ecol. Res.* **1974**, *8*, 1–105.
20. Shrivastava, P. Knowledge Ecology: Knowledge Ecosystems for Business Education and Training. 1998. Available online: http://www.facstaff.bucknell.edu/shrivast/KnowledgeEcology.html (accessed on 18 May 2017).
21. Malhotra, Y. Knowledge Management for Organizational White Waters: An Ecological Framework. *Knowl. Manag.* **1999**, *2*, 18–21.
22. Pór, G. Nurturing Systemic Wisdom through Knowledge Ecology. *Syst. Think.* **2000**, *11*, 1–5.

23. Bray, D.A. Knowledge Ecosystems: A Theoretical Lens for Organizations Confronting Hyperturbulent Environments. In *Organizational Dynamics of Technology-Based Innovation: Diversifying the Research Agenda*; Springer: Medford, MA, USA, 2007; pp. 457–462.
24. Hai, D.; Farookh, H.; Chang, E. A Human-Centered Semantic Service Platform for the Digital Ecosystems Environment. *World Wide Web* **2010**, *13*, 75–103.
25. Hai, D.; Farookh, H.; Chang, E. A framework for discovering and classifying ubiquitous services in digital health ecosystems. *J. Comput. Syst. Sci.* **2011**, *77*, 687–704.
26. Hai, D.; Farookh, H.; Chang, E. Semantic service matchmaking for Digital Health Ecosystems. *Knowl.-Based Syst.* **2011**, *24*, 761–774.
27. Hai, D.; Farookh, H.; Chang, E. A service concept recommendation system for enhancing the dependability of semantic service matchmakers in the service ecosystem environment. *J. Netw. Comput. Appl.* **2011**, *34*, 619–631.
28. Hai, D.; Farookh, H. Service-requester-centered service selection and ranking model for digital transportation ecosystems. *Computing* **2015**, *97*, 79–102.
29. Corallo, A.; Passiante, G.; Prencipe, A. (Eds.) *The Digital Business Ecosystem*; Edward Elgar Publishing: Northampton, MA, USA, 2007.
30. Nachira, F.; Nicolai, A.; Dini, P.; Le Louarn, M.; Rivera Leon, L. (Eds.) *Digital Business Ecosystems*; European Commission: Brussels, Belgium, 2008.
31. Burgin, M. Techno-Ecology and Problems of Education. In *Intellectual Innovations in Society and Development of Education*; NGU: Novosibirsk, Russia, 1997; pp. 175–178. (In Russian)
32. Damiani, E.; Uden, L.; Trisnawaty, W. The future of E-learning: E-learning ecosystem. In Proceedings of the 2007 Inaugural IEEE-IES Digital EcoSystems and Technologies Conference (DEST '07), Cairns, Australia, 21–23 February 2007.
33. Bateson, G. *Steps to an Ecology of Mind*; Jason Aronson Inc.: St. Albans, UK, 1973.
34. Bookchin, M. *The Ecology of Freedom*; AK Press: Stirling, UK, 2005.
35. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
36. Herbert, N. *Quantum Reality: Beyond the New Physics*; Anchor Books: New York, NY, USA, 1987.
37. Burgin, M. Ideas of Plato in the context of contemporary science and mathematics. *Athens J. Hum. Arts* **2017**, *4*, 161–182.
38. Sneed, J.D. *The Logical Structure of Mathematical Physics*; D. Reidel: Dordrecht, The Netherlands, 1971.
39. Suppe, F. (Ed.) *The Structure of Scientific Theories*; University of Illinois Press: Urbana, IL, USA, 1974.
40. Burgin, M.; Kuznetsov, V. *Introduction to Modern Exact Methodology of Science*; International Science Foundation: Moscow, Russia, 1994. (In Russian)
41. Burgin, M. *Theory of Knowledge*; World Scientific: New York, NY, USA, 2016.
42. Zhong, Y.X. *Principles of Advanced Artificial Intelligence*; Science Press: Beijing, China, 2014. (In Chinese)
43. Zhong, Y.X. Information Ecology. *Proceedings* **2017**, *1*, 200. [CrossRef]
44. Burgin, M. *Theory of Named Sets*; Mathematics Research Developments; Nova Science: New York, NY, USA, 2011.
45. Burgin, M. Bidirectional named sets as structural models of interpersonal communication. *Proceedings* **2017**, *1*, 58. [CrossRef]
46. Burgin, M. Theory of Named Sets as a Foundational Basis for Mathematics. In Proceedings of the International Symposium on Structures in Mathematical Theories (SSMT-90), San Sebastián, Spain, 25–29 September 1990; pp. 417–420.
47. Burgin, M. Named Sets as a Basic Tool in Epistemology. *Epistemologia* **1995**, *18*, 87–110.
48. Zhong, Y.X. Structuralism? Functionalism? Behaviorism? Or Mechanism? Looking for Better Approach to AI. *Int. J. Intell. Comput. Cybern.* **2008**, *1*, 325–336. [CrossRef]
49. Zhong, Y.X. Principles of Information Conversion: Integrative Theory of Information, Knowledge, and Intelligence. *Sci. Bull.* **2013**, *58*, 1300–1306. [CrossRef]


© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

*Proceedings*

# Complexity of Legal Processes and Systems †

**Mark Burgin [1] and Kees (C.N.J.) de Vey Mestdagh [2,*]**

[1] Department of Mathematics, University of California, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu
[2] Foundation for Law & ICT, 9700 Groningen, The Netherlands
* Correspondence: c.n.j.de.vey.mestdagh@law-and-ict.org
† Conference Theoretical Information Studies (TIS), Berkeley, CA, USA, 2–6 June 2019.

Published: 20 May 2020

**Abstract:** Complexity measures allow for reflecting on critical issues and estimating the efficiency of various processes and systems. To better organize the functioning of the legal domain, it is rational to use complexity measures. In essence, any legal process is an information process in the legal domain, involving one or several legal systems. Thus, the goal of this work is the development of theoretical and practical tools for the estimation of complexity of information processes, that is, complexity measures in the legal domain, with the goal to ameliorate functioning of the legal systems.

**Keywords:** legal system; legal process; complexity; measure; law; norm; rule

## 1. Introduction

Legal systems are evolutionary social, i.e., in some sense, natural, behavioral systems, although many like to think that they are just artificial constructions created by people. The latter idea resulted in the amorphous areas of normative ethics and legal studies, although the legal domain needs meta-ethics, descriptive ethics, ethnology and ethology. Having this in mind, if we analyze the essential characteristics of legal systems, we can see that they are the result and product of millions of years of evolution in a changing world. Law consists of formal rules describing expectations concerning a class of human behavior. The origin of these formal rules is an individual or collective process that translates values into individual and general norms, norms into rules and rules into social standards and formal laws (the law). A value is an innate or social need. The argument for the evolutionary character of legal systems is made in [1,2]. The fact that many locally parallel and temporally sequential legal systems have the same characteristics does not prove that they are designed like—for example—cars, but provides evidence that they are the consequence of (have developed on) the basis of the same natural characteristics intrinsic for different societies.

To study legal systems with the goal of improving their functioning, we use complexity theory, because mathematical models of complexity describe efficiency. Indeed, complexity measures allow reflecting critical issues and estimating the efficiency of various processes and systems. Thus, to better organize functioning of the legal domain, it is rational to use complexity measures, which have been constructed in complexity theory and used in a variety of scientific and practical areas. In essence, any legal process is an information process in the legal domain, involving one or several legal systems. Thus, our goal is the development of theoretical and practical tools for the estimation of complexity of information processes, that is, complexity measures, in the legal domain with the goal to ameliorate functioning of the legal domain. Knowledge of dynamic and structural characteristics reflecting complexity of information processes in the legal domain can improve results of these processes, decrease their complexity and better satisfy needs of the society where these processes go on.

To elaborate legal complexity measures, it is necessary to have formal descriptions of legal processes. Here, we use the Logic of Reasonable Inferences (LRI), as the tool for building a formal model of legal knowledge and formalizing descriptions of the legal processes. The LRI was introduced in [3] and utilized for the development of the computer program Argumentator, which was implemented and empirically validated against a multitude of real life legal cases [4]. In [5], a formal model of the factors that determine complexity in the legal domain is presented.

To develop legal complexity measures, we use the theory of direct and dual complexity measures developed in [6,7], as well as inductive complexity introduced and studied in [8]. According to these theories, complexity, at first, is measured for separate processes, and then it is integrated into the complexity of the system in question.

## 2. Complexity of Systems and Processes

The complexity of a system *R* is determined by the resources used by this system in the processes of functioning. Note that these resources can be used by the system itself and/or by another system interacting, e.g., using the system *R*.

Systems use different kinds of resources:

- Natural resources consumed by the system R: time, space, information, energy, power, minerals and so on;
- Social resources consumed by the system R: people involved, their time, efforts, expertise, knowledge and so on;
- Artificial resources consumed by the system R: system time, system space, data, knowledge, memory, system units, system actions and so on.

There are three types of direct complexity measures of a system or a process: static, functional and processual complexity measures.

**Definition 1.** *Static complexity measures depend on the system, in which this process goes, or on the system, which performs (realizes) this process, being estimated by resources given as input for the process.*

Examples of such resources are the system itself, the program of its functioning or the input information.

**Definition 2.** *Functional complexity measures depend both on the input, e.g., algorithm or program, and the output of the process that goes in the system or is realized by it.*

As examples, we can take such measures as the quality of result obtained by the system or the ratio of the size of the input information to the size of the output information.

**Definition 3.** *Processual complexity measures depend on the process itself.*

As examples, we can take such measures as the time of processing some given data or the volume of memory that is demanded by this processing.

In general, there are three classes of complexity measures: *direct*, *dual*, and *mixed* complexity measures. Let us consider some examples of complexities for such systems as algorithms and programs.

*Direct complexity measures:*

- The length of an algorithm or program (static complexity);
- The quality of the result (functional complexity);
- Time of the computation directed by the algorithm or program (functional complexity).

*Dual complexity measures:*

- The length of the shortest algorithm or program needed to obtain the given system (static complexity);
- The best quality of the possible result (functional complexity);
- The least time needed to obtain the given system (functional complexity).

*Mixed complexity measures:*

- Computational complexity, e.g., time, of the shortest program that produces the given system.

It is possible to define complexity of a system (organization) by processes, which either go in this system (organization) or are performed (organized) by this system (organization).

- *Worst case complexity* of a system (organization) is the top measure of the processes, which either go in this system (organization) or are performed (organized) by this system (organization);
- *Average case complexity* of a system (organization) is the average measure of the processes, which either go in this system (organization) or are performed (organized) by this system (organization);
- *Best case complexity* of a system (organization) is the least measure of the processes, which either go in this system (organization) or are performed (organized) by this system (organization).

Having a direct complexity measure $\mu$, we define its dual complexity measure $\mu(R)$ of a system $R$ as the minimal value $\mu(A)$ of the system, program or process that allows obtaining, e.g., construction or computation, of $R$.

For instance, the algorithmic (Kolmogorov) complexity is a paradigmatic example of a dual complexity measure. The algorithmic (Kolmogorov) complexity of a string of symbols $R$ is the minimal length of a program (algorithm) that allows computation (recognition) of $R$ [9,10]. In general, algorithmic (Kolmogorov) complexity measures have been defined and explored in the axiomatic setting. This allows application of these measures in a variety of domains (such as medicine, biology, neurophysiology, physics, economics, hardware and software engineering) in general, and the legal domain in particular.

There are also temporally related complexity measures:

- The *estimated complexity* of a system or process reflects opinions about the complexity of this system or process;
- The *projected complexity* of a system or process reflects intentions or requirements to the complexity of this system or process;
- The *actualized complexity* of a system or process is the complexity of this system or process measured a posteriori.

Analyzing social processes in general and legal processes in particular, we see that almost any process $P$ is based on the process *org P* of preparation and organization of $P$. In addition, there is the process *act P* of putting into action the results of $P$.

For instance, taking such a process as court trial, we see that its preparation includes investigation while putting its decision into action includes law enforcement.

As a result, we have three temporal legal complexity measures:

- The *temporal organizational complexity* of a legal process is time used for organization of the legal process;
- The *temporal operational complexity* of a legal process is time used for conducting the legal process coming to a definite decision;
- The *temporal execution complexity* of a legal process is time used for putting the legal decision into action.

In turn there are three extent legal complexity measures:

- The *extent organizational complexity* of a legal process is the number of items used for organization of the legal process;
- The *extent operational complexity* of a legal process is the number of items used for conduction the legal process coming to a definite decision;

- The *extent execution complexity* of a legal process is the number of items used for putting the legal decision into action.

Both temporal and extent complexity measures are direct. Dual complexity measures are obtained by application of minimization operation to direct complexity measures.

## 3. Complexity in the Legal Domain

Items utilized in legal processes, which we will call legal articles, can be:

- instances of material evidence, i.e., weapon or fingerprints;
- participants of the process, e.g., attorneys, judges, witnesses;
- those who prepare (organize) the process, e.g., detectives, attorneys, judges;
- testimonies, depositions, speeches of participants during the process, negotiations, subpoenas, requests, interrogatories, affidavits, transcripts, objections, statements;
- legal rules and laws.

Rules for conducting legal processes, e.g., laws of the country, are basic items (legal articles) utilized in legal processes. This delineates an important class of legal complexity measures, namely, of extent operational complexities—imperative complexity measures.

The imperative complexity of a legal process is the number of legal rules used for conduction the legal process coming to a definite decision.

There are three modalities of legal rules (laws):

- Obligatory rules (laws) describe what is required to do in a legal process;
- Discretionary rules (laws) describe what is permitted (possible) to do in a legal process;
- Veto rules (laws) describe what is prohibited to do in a legal process.

Now let us look at dual complexity measures of legal processes.

- The obligatory (requisite) organizational algorithmic complexity of a legal process is the minimal number of items (legal articles) necessary (sufficient) to organize the legal process;
- The obligatory (requisite) operational algorithmic complexity of a legal process is the minimal number of items (legal articles) necessary (sufficient) to conduct the legal process coming to a definite decision;
- The obligatory (requisite) execution algorithmic complexity of a legal process is the minimal number of items (legal articles) necessary (sufficient) for putting the legal decision into action.

It is possible to consider only some type legal articles, e.g., rules or laws, estimating the complexity of a legal process by the (minimal) number of used rules or laws. However, for the sides involved in a legal process, it is important not only to organize, conduct and put into action a legal process, but to achieve definite results in this process. These assumptions bring us to new types of algorithmic complexity.

Let us suppose that there are sides (participants) $A_1, \dots, A_n$ in a legal process $P$.

- The $A_k$-winning organizational algorithmic complexity of a legal process is the minimal number of items (legal articles) necessary (sufficient) to organize the legal process successful for $A_k$.
- The $A_k$-winning operational algorithmic complexity of a legal process is the minimal number of items (legal articles) necessary (sufficient) to conduct the legal process successful for $A_k$.

Further research will focus first of all on the legal articles distinguished in the formal model of the factors that determine complexity in the legal domain [5].

## 4. Conclusions

We have elaborated a system approach to the estimation of complexity of information processes in the legal domain aimed at upgrading and modernizing the functioning of the legal domain based on information technology.

An interesting problem for future research is construction of legal complexity measures based on Shannon's entropy [11] and Fisher's statistical information measures [12], as there are intrinsic relations between complexity measures introduced and studied in this paper, and the above mentioned information measures [13,14]

Communication is an important component of diverse legal processes. Consequently, it would be important to construct and study communication legal complexity, taking into account communication complexity of algorithmic processes [15].

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. de Vey Mestdagh, C.N.J.; Hoepman, J.H. Inconsistent Knowledge as a Natural Phenomenon: The Ranking of Reasonable Inferences as a Computational Approach to Naturally Inconsistent (Legal) Theories. In *Information and Computation, Essays on Scientific and Philosophical Understanding of Foundations of Information and Computation*; Dodig-Crnkovic, G., Burgin, M., Eds.; World Scientific: Hackensack, NJ, USA; pp. 439–476.
2. de Vey Mestdagh, C.N.J. *What Is Law? Paper for the Workshop Governance Meets Law, 23–24 June 2011*; University of Groningen: Het Kasteel, Groningen, The Netherlands, 2011.
3. de Vey Mestdagh, C.N.J.; Verwaard, W.; Hoepman, J.H. *The Logic of Reasonable Inferences. Legal Knowledge Based Systems, Model-Based Legal Reasoning*; Breuker, J.A., de Mulder, R.V., Hage, J.C., Eds.; Vermande: Lelystad, The Netherlands, 1991; pp. 60–76.
4. de Vey Mestdagh, C.N.J. Legal Expert Systems. Experts or Expedients? The Law in the Information Society. In Proceedings of the Fifth International Conference of the Italian National Research Council, Florence, Italy, 2–5 December 1998; Ciampi, C., Marinai, E., Eds.; Istituto per la documentazione giuridica del Centro Nazionaledelle Richerche: Firenze, Italy, 1998; p. 8.
5. de Vey Mestdagh, C.N.J. A Model of Complexity for the Legal Domain. In Proceedings of the IS4SI 2017 Summit Digitalisation for a Sustainable Society, Gothenburg, Sweden, 12–16 June 2017; Volume 1, p. 192.
6. Burgin, M.S. Generalized Kolmogorov Complexity and other Dual Complexity Measures. *Cybernetics* **1990**, *26*, 481–490.
7. Burgin, M. Algorithmic Complexity of Computational Problems. *Int. J. Comput. Inf. Technol.* **2010**, *2*, 149–187.
8. Burgin, M. Algorithmic Complexity of Recursive and Inductive Algorithms. *Theor. Comput. Sci.* **2004**, *317*, 31–60.
9. Kolmogorov, A.N. Three approaches to the definition of the quantity of information. *Probl. Inf. Transm.* **1965**, *1*, 3–11.
10. Chaitin, G.J. On the Length of Programs for Computing Finite Binary Sequences. *J. Assoc. Comput. Mach.* **1966**, *13*, 547–569.
11. Shannon, C.E. The Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423.
12. Fisher, R.A. *Statistical Methods and Scientific Inference*; Oliver and Boyd: Edinburgh, UK, 1956.
13. Li, M.; Vitanyi, P. *An Introduction to Kolmogorov Complexity and Its Applications*; Springer: New York, NY, USA, 1997.
14. Burgin, M. Inductive Complexity and Shannon Entropy. In *Information and Complexity*; World Scientific: New York, NY, USA; London, UK; Singapore, 2016; pp. 16–32.
15. Hromkovic, J. *Communication Complexity and Parallel Computing*; Springer: New York, NY, USA, 1997.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

Proceeding Paper

# Monitoring Thermal Conditions and Finding Sources of Overheating †

**Arkadiy Dantsker [1,*] and Mark Burgin [2,*]**

[1] Detex Analytics, Seattle, WA 98034, USA
[2] Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA
* Correspondence: amdantsker@cs.gmail.com (A.D.); markburg@cs.ucla.edu (M.B.)
† Presented at the Conference on Theoretical and Foundational Problems in Information Studies, IS4SI Summit 2021, Online, 12–19 September 2021.

**Abstract:** This research defines a method for detecting the source of thermal heating in a cargo container by monitoring surface temperature and through reverse engineering of the thermal process. A simplified heat direct model in 3D space was formulated and the inverse model wa olved using the theory of hypernumbers.

**Keywords:** information; temperature; heat; source; hypernumber

**Citation:** Dantsker, A.; Burgin, M. Monitoring Thermal Conditions and Finding Sources of Overheating. *Proceedings* **2021**, *81*, 38. https://doi.org/10.3390/proceedings2022081038

Academic Editor: Marcin Schroeder

Published: 19 September 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Cargo fires and explosions often happen due to self-heating [1]. Self-heating occurs when an exothermic (heat-producing) chemical or biochemical reaction starts within a body of cargo. Monitoring thermal conditions in a container with cargo is important for detecting the source of overheating. It is a challenging problem due to the complexity of three-dimensional heat exchange models, which provide information about the behavior of the cargo [2,3]. Finding the source of overheating will allows us to predict dangerous thermal situations and events. This task is critical for multiple situations such as air cargo transport safety, protecting pharmaceutical products from spoiling, and preventing fire. The list of overheating causes includes:

a. Calcium hypochlorite and other solids oxidation. The self-decomposition of such solids can evolve self-heating process. This can lead to "thermal runaway".
b. Biomass heating due to the rotting process, in which methane concentration is produced. "Anaerobic" rotting can produce dangerous concentrations of methane and lead to explosion.
c. The decomposition of fertilizers in the bulk with the evolution of heat.
d. Lithium battery heat release due to natural discharge.
e. Liquid monomers polymerization which evolves heat. Self-heating normally occurs in localized hot spots within a bulk cargo and identifying events by temperature measurement is a challenging problem.

The described solution monitors container temperatures at defined points, solving the ill-posed problem of re-engineering temperature transfer in a 3D space with insulated boundary conditions. The heat exchange in the package of the batteries could be approximated with a composed heat rate coefficient. The direct model of heat exchange with pointed heat source in a semi-infinite body is developed in [2]. The model can be extended for bounded 3D spaces. Such an extension is given below. The location of the source can be found by resolving the ill-posted problem using the minimum least square criterion and finding a solution to the non-linear system of the equations with the hypernumber method [4–7]. In this work, we focus on detecting the pointed sources of overheating in homogeneous media or such that with some approximation can be assumed as homogeneous.

Our model of inverse engineering for locating the source of heating and predicting transient temperature distribution inside the container has several computational advantages:

- It is a simplified direct model for calculating the heat propagation from a pointed source in a three-dimensional space.
- The inverse model uses a hypernumber recursive analytical method, which is much faster in comparison with the utilized numerical methods. Thermodynamic parameters, which in most cases are not known, or the values, which cannot be estimated precisely, are calculated based on inverse re-engineering definitions.
- The hypernumber method guarantees the convergence of the process.
- Due to the relative simplicity of the algorithm, the computation can be implemented using inexpensive controllers such as, for example, Atmega 2560. The time of calculation using embedded *C* would be in a range of a tenth of milliseconds. This approach can be generalized to the cases when the heat exchange is not the same in all directions.

## 2. Direct Model of the Heat Transfer in a Cargo Container

The three-dimensional heat transfer model from the pointed source is derived from the one-dimensional heat transfer model [2];

$$\begin{aligned} dT(x,y,z,t) = {} & \frac{dq}{\gamma C(4\pi a\pi(t-t_0))^{\frac{3}{2}}} e^{-\frac{(x-x_s)^2+(y-y_s)^2+(z-z_s)^2}{4a(t-t_0)}} + \\ & \frac{dq}{\gamma C(4\pi a\pi(t-t_0))^{3/2}} e^{-\frac{(x-x_s)^2+(y-y_s)^2+(z+(h-z_s))^2}{4a(t-t_0)}} + \\ & \frac{dq}{\gamma C(4\pi a\pi(t-t_0))^{3/2}} e^{-\frac{(x-x_s)^2+(y+(b-y_s))^2+(z-\ z_s)^2}{4a(t-t_0)}} + \\ & \frac{dq}{\gamma C(4\pi a\pi(t-t_0))^{3/2}} e^{-\frac{(x+(a-x_s))^2+(y-y_s)^2+(z+z_s)^2}{4a(t-t_0)}} \end{aligned} \tag{1}$$

where $a$—diffusivity, $\gamma$—density of the heat transfer medium, $dq$—delta thermal heat, $x_s$, $y_s$, $z_s$—coordinates of the heat source.

Considering the steady rate of the heat, the temperature at time t can be found as:

$$\begin{aligned} T(x,y,z,t) = {} & \int_0^t \frac{pd\tau}{\gamma C(4\pi a\pi(\tau-t_0))^{3/2}} \Big(e^{-\frac{(x-x_s)^2+(y-y_s)^2+(z-\ z_s)^2}{4a(\tau-t_0)}} + \\ & e^{-\frac{(x-x_s)^2+(y-y_s)^2+(z+(h-z_s))^2}{4a(\tau-t_0)}} + \\ & e^{-\frac{(x-x_s)^2+(y+(b-y_s))^2+(z-\ z_s)^2}{4a(\tau-t_0)}} + e^{-\frac{(x+(a-x_s))^2+(y-y_s)^2+(z+z_s)^2}{4a(\tau-t_0)}} \Big) \end{aligned} \tag{2}$$

## 3. Reverse Engineering of the Heat Source with the Theory of Hypernumbers

The source of heating $A(x_s,\ y_s\ ,\ z_s)$ can be found with the Least Square algorithm

$$A(x_s,\ y_s\ ,\ z_s) : U = min \sum_{i=0}^{m=amoint\ of\ sensors} \sum_{j=0}^{n} \left(T_{i,j}^a - T_{i,j}^m\right)^2 \tag{3}$$

where $T_{i,j}^a$—is the analytically defined temperature at the time moment with index $j$ at temperature sensor $j$ location, $T_{i,j}^m$—is the monitoring temperature at the time moment with index $j$ and sensor $m$.

$$F_1 = \frac{\partial U}{\partial x_s} = 2\sum_{0}^{m}\sum_{j=0}^{n}\left(T_{i,j}^a - T_{i,j}^m\right)\frac{\partial T_{i,j}^a}{\partial x_s} = 0 \tag{4}$$

$$F_2 = \frac{U}{\partial y_s} = 2\sum_{0}^{m}\sum_{j=0}^{n}\left(T_{i,j}^{a} - T_{i,j}^{m}\right)\frac{\partial T_{i,j}^{a}}{\partial y_s} = 0 \tag{5}$$

$$F_3 = \frac{\partial U}{\partial z_s} = 2\sum_{0}^{m}\sum_{j=0}^{n}\left(T_{i,j}^{a} - T_{i,j}^{m}\right)\frac{\partial T_{i,j}^{a}}{\partial z_s} = 0 \tag{6}$$

$$F_4 = \frac{\partial U}{\partial p} = 2\sum_{0}^{m}\sum_{j=0}^{n}\left(T_{i,j}^{a} - T_{i,j}^{m}\right)\frac{\partial T_{i,j}^{a}}{\partial p} = 0 \tag{7}$$

$$F_5 = \frac{\partial U}{\partial a} = 2\sum_{0}^{m}\sum_{j=0}^{n}\left(T_{i,j}^{a} - T_{i,j}^{m}\right)\frac{\partial T_{i,j}^{a}}{\partial a} = 0 \tag{8}$$

The $(x_s)_m$, $(y_s)_m$, $(z_s)_m$ , $\gamma_m$, $a_m$ are solutions defined with hypernumbers in Equations (9)–(12)

$$(x_s)_m = H_n((x_s)_m)_{m\in\omega} \tag{9}$$

$$(y_s)_m = H_n((y_s)_m)_{m\in\omega} \tag{10}$$

$$(z_s)_m = H_n((z_s)_m)_{m\in\omega} \tag{11}$$

$$\gamma_m = H_n(\gamma_m)_{m\in\omega} \tag{12}$$

$$a_m = H_n(a_m)_{m\in\omega} \tag{13}$$

The equations for the hypernumber deviations $\{\delta(x_s)_m, \delta(y_s)_m, \delta(z_s)_m, \delta\gamma_m, \delta a_m\}$ are defined below

$$\begin{pmatrix} \delta F_{1,m+1} \\ \delta F_{2,m+1} \\ \delta F_{3,m+1} \\ \delta F_{4,m+1} \\ \delta F_{5,m+1} \end{pmatrix} = \begin{pmatrix} d_{11} & d_{12} & d_{13} & d_{14} & d_{15} \\ d_{21} & d_{22} & d_{23} & d_{24} & d_{25} \\ d_{31} & d_{32} & d_{33} & d_{34} & d_{35} \\ d_{41} & d_{42} & d_{43} & d_{44} & d_{45} \\ d_{51} & d_{52} & d_{53} & d_{54} & d_{55} \end{pmatrix} \times \begin{pmatrix} \delta(x_s)_m \\ \delta(y_s)_m \\ \delta(z_s)_m \\ \delta\gamma_m \\ \delta a_m \end{pmatrix} = \begin{pmatrix} (1-\theta)F_{1,m} \\ (1-\theta)F_{2,m} \\ (1-\theta)F_{3,m} \\ (1-\theta)F_{4,m} \\ (1-\theta)F_{5,m} \end{pmatrix} \tag{14}$$

The coefficients $d_{i,j}$ are calculated with Equations (15)–(19)

$$d_{i,1} = \frac{\partial F_i}{\partial x_s} \tag{15}$$

$$d_{i,1} = \frac{\partial F_i}{\partial y_s} \tag{16}$$

$$d_{i,1} = \frac{\partial F_i}{\partial z_s} \tag{17}$$

$$d_{i,1} = \frac{\partial F_i}{\partial x_s} \tag{18}$$

$$d_{i,1} = \frac{\partial F_i}{\partial x_s} \tag{19}$$

Using Equations (3)–(19) was written software that shows the accurate results of solving the inverse problem with artificial simulation of the heat source.

## 4. Conclusions

The described method for detecting thermal threats in a cargo container would allow essentially improving the safety of cargo transportation.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Sanders, N. Fire. Available online: https://www.swedishclub.com/media_upload/files/Loss%20Prevention/Fire/TSC%20Fire%20Guide%20%28web%29.pdf#page=14 (accessed on 9 May 2019).
2. Raval, A.; Solanki, R.; Yadav, R. A simplified heat transfer model for predicting temperature change inside food package kept in cold room. *J. Food Sci. Technol.* **2011**, *50*, 257–265. [CrossRef]
3. Seitz, S.; Franke, J.; Giebish, G. Determination of the Self-ignition Behaviour of Bulk Materials from Heat Storage Tests below Atmospheric Pressure. *Chem. Eng. Trans.* **2016**, *48*, 583–588.
4. Burgin, M. Nonlinear partial differential equations in extrafunctions. *Integr. Math. Theory Appl.* **2010**, *2*, 17–50.
5. Burgin, M.; Dantsker, A.M. Lithium Battery Temperature Prediction. *Integr. Math. Theory Appl.* **2014**, *3*, 319–331.
6. Burgin, M.; Dantsker, A. *Real-Time Inverse Modeling of Control Systems in Hypernumbers*; Nova Science Publishers: New York, NY, USA, 2015.
7. Burgin, M.; Dantsker, A.M. *A Method of Solving of Operator Equations of Mechanics by means of the Theory of Hypernumbers*; Notices of the Academy of Sciences of Ukraine: Kyiv, Ukraine, 1995; Volume 8, pp. 27–30.

proceedings

MDPI

Proceeding Paper

# Named Sets as an Efficient Tool for Modeling Data Relationships in Database Models †

**Mark Burgin [1,*] and H. Paul Zellweger [2]**

[1] Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA
[2] ArborWay Labs, Rochester, MN 55901, USA; pz@arborwaylabs.com
* Correspondence: markburg@cs.ucla.edu
† Presented at the Conference on Theoretical and Foundational Problems in Information Studies, IS4SI Summit 2021, Online, 12–19 September 2021.

**Abstract:** An important problem for databases is the unification of utilized data structures and amplification of operation tools. Here, after a brief overview and analysis of database models, we demonstrate that all considered data can be reduced to systems of named sets allowing representation of the described database models as special cases of the named-set database model, which provides efficient operations for data mining, information extraction, and database management.

**Keywords:** database; data; information; data structure; database model; graph; system; network; relation

**Citation:** Burgin, M.; Zellweger, H.P. Named Sets as an Efficient Tool for Modeling Data Relationships in Database Models. *Proceedings* **2022**, *81*, 51. https://doi.org/10.3390/proceedings2022081051

Academic Editor: Rao Mikkilineni

Published: 16 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Data are representations and containers (carriers) of information. Data andtheir relationships form data structures. That is why modeling data relationships in the context of data and knowledge structures is critically important for organization and optimization of information processes in databases and beyond.

Data have different structures and both data processing in general and data mining in particular depends on these structures. For instance, the well-known computer scientist and mathematician Yuri Gurevich concluded his lecture [1] on the advancement of theoretical computer science with the statement that to be useful for database technology, computational models have to work with structures and not with strings of symbols. The most popular data structures include Boolean values, characters, integers, fixed-precision number values, floating-point number values, arrays, records, lists, streams, sets, multisets, stacks, queues, and graphs, just to mention the most important of them. Here, in addition to these conventional data structures, we consider named sets and chains of named sets as the fundamental data structures for modeling data relationships in database models.

## 2. Database Models

As Angles and Gutierrez write, from a database point of view, the conceptual tools that make up a database model should at least address data structuring, description, maintenance, and a way to retrieve or query the data [2]. These principles imply that a database model consists of three components: a system of utilized data structure types with their logical and operational organization; a system of operators and inference rules; and a system of integrity rules [3]. The logical structure of a database includes the relationships and constraints that determine how data can be stored and accessed. As a rule, database models mainly pay attention to utilized data structures, which are represented by a database model. Let us consider the main database models used for storing and preserving information.

The *hierarchical database model* is the oldest being developed by IBM for information management system (IMS) [4]. In it, data are organized in the tree structure.

The *network database model* represents data with records and sets. Records contain fields, which may be organized hierarchically, while sets define one-to-many relationships

between records. This model is an expansion of the hierarchical model allowing many-to-many relationships in a tree-like structure.

The *flat*(or *table*) *database model* consists of a single, two-dimensional array of data elements, where all members of a given column are assumed to be similar values and all members of a row are assumed to be related to one another having the same type. The flat model predates the relational model.

The *relational database model* was introduced by Codd [5] and highlights the concept of abstraction levels by introducing the idea of separation between physical and logical levels. It is based on the notions of sets and relations. Due to its ease of use, it gained wide popularity among business applications.

The *multivalue database model* is an extension of the relational model. In it, a field/attribute can have several values at the same time.

The *semantic database model* represents objects and their relations in a natural and clear way, providing users with tools to correctly reflect the desired domain semantics. The *entity-relationship model* is an example of semantic database models [6].

The *resource space database model* (RSM) is a non-relational data model based on multi-dimensional classification [7].

The *object-orienteddatabase model* is based on the object-oriented paradigm representing data as a collection of objects that are organized into classes and assigned complex values [7].

The *graphdatabase model* represents objects and their relations in the form of a graph overcoming the limitations imposed by traditional database models with respect to capturing the inherent graph structure of data appearing in applications such as hypertext or geographic information systems, where the interconnectivity of data is an important aspect [2].

The *semistructureddatabase model* exemplifies data with a flexible structure, for example, documents or Web pages. Semistructured data are neither raw nor strictly typed, as in the conventional database systems [8].

The *XML* (eXtensible Markup Language) *database model* focuses on information with tree-like structure [9].

The *named-set database model* represents information in the form of systems of named sets such as named set chains [10].

## 3. Named Sets in Database Modeling and Data Representation

Here we show that all of these database models can be treated as special cases of the named-set database model since all utilized data structures are either named sets or systems of named sets.

So, the question is as follows: Why are named sets really essential, and what is so specific about them?

First, it is proved that any mathematical structure is a named set or is built of named sets and thus, the named set is the most fundamental structure in all mathematics [11]. For instance, functions, relations, variables, graphs, multigraphs, and morphisms (arrows) in categories are special cases of named sets. Ordinary sets are also specific named sets, namely they are singlenamed sets since all elements in a set with the name, say Q, have the common name "an element of the set Q" [11].

Second, we see that named sets are vital for representation of data, knowledge, and information as well as all cognitive processes and communication [12]. Taking any book on databases, we see many examples of named sets (cf., for example, [13]).

Third, it is proved that the named set (also called fundamental triad) is the most basic structure in nature [12]. As a consequence, named sets have become ubiquitous in modeling natural systems.

Let us consider the basic definition.

**Definition 1.** *(a) A basic named set, also called a basic fundamental triad, is a triad* $\mathbf{X} = (X, f, N)$ *with the following visual (graphic) representation:*

$$X \xrightarrow{f} N$$

*(b) A bidirectional named set, also called a bidirectional fundamental triad, is a triad* $\mathbf{X} = (X, f, Z)$ *with the following visual (graphic) representation:*

$$X \xleftrightarrow{f} N$$

In this triad $\mathbf{X} = (X, f, N)$, the components $X$ and $N$ are two objects and $f$ (in case a) is a correspondence (e.g., a binary relation) from $X$ to $N$ and (in case b) is a correspondence (e.g., a binary relation) between $X$ and $Z$, which goes in two directions. With respect to $\mathbf{X}$, the object $X$ is called the *support* of $\mathbf{X}$ and denoted $\mathrm{S}(\mathbf{X})$, the object $N$ is called the *component of names* (*reflector*) or *set of names* of $\mathbf{X}$ and denoted $\mathrm{N}(\mathbf{X})$, and the object $f$ is called the *namingcorrespondence* (*reflection*) of $\mathbf{X}$ and denoted $\mathrm{r}(\mathbf{X})$. It means that $\mathbf{X} = (\mathrm{S}(\mathbf{X}), \mathrm{r}(\mathbf{X}), \mathrm{N}(\mathbf{X}))$. Note that in $\mathbf{X}$, components $X$ and $N$ are not automatically sets, while $f$ is not necessarily a mapping or a function even if $X$ and $N$ are sets. For instance, $X$ and $N$ are sets of words and $f$ is an algorithm.

The standard example is a basic named set (basic fundamental triad), in which $X$ consists of people, $N$ consists of their names, and $f$ is the correspondence between people and their names.

Let us analyze the considered database models in the context of named sets.

Hierarchical data are organized as tree structures, which are chains of named sets starting with the root of the tree and ending with its leaves [14]. Consequently, the hierarchical database model is a special case of the named-set database model.

Any network in general and a network of records, in particular, have the structure of a graph. A graph consists of vertices (nodes) and edges connecting some vertices. If $V$ is the set of all vertices and $E$ is the set of all edges in graph $G$, then this graph is a named set $(V, E, V)$. Consequently, the network database model is a special case of the named-set database model. Note that as records contain fields and fields are named sets with values as their names, any network of records is a nested named set [15].

Any two-dimensional arrays in general and two-dimensional arrays of data elements, in particular, are nested named sets [15]. Consequently, the flat database model is a special case of the named-set database model.

Relations are special cases of set-theoretical named sets [11]. Consequently, the relational and multivalue database models are special cases of the named-set database model.

Objects (entities) with relations form a named set [11]. Consequently, the semantic database model is a special case of the named-set database model.

Any classification is a set-theoretical named set [11]. Consequently, the resource space database model is a special case of the named-set database model.

In the object-oriented approach, data are formed as objects and each object has a name, set of attributes, and behaviors. Thus, the support of an object consists of its name which is connected to its attributes and behaviors. This is the structure of a named set. In addition, any set of objects is the named set, which is the union of the named sets of individual objects [11]. Consequently, the object-oriented database model is a special case of the named-set database model.

As it was demonstrated that any graph is a named set, the graph database model is a special case of the named-set database model.

Finally, as any structure is built from named sets, the semistructured and XML database models are special cases of the named-set database model. For XML data, this was demonstrated in [16].

## 4. Conclusions

An important peculiarity of utilization of named sets in databases is that algorithms in general, and software systems in particular, for operation with data are also specific named sets and systems of named sets. Namely, they are algorithmic named sets and their systems (i.e., such named sets in which the relation $f$ is an algorithm or a program) [11,17].

An important advantage of the named-set database model is not only structural unification but also operational affluence. Indeed, manipulation with data demands utilization of various operations and, in the case of using named sets for data representation, a variety of operations such as mappings of different kinds, union, intersection, difference, renaming, naming, interpreting, and reinterpreting, and their properties are provided by the theory of named sets [11].

Operating with data in the named-set database model involves structural recursion capturing the system's repeating patterns. As a result, an important direction for future research is exploration of structural recursion in the context of named sets and its application to the problems of data search, as well as to database development and management.

Nesting is an important phenomenon in many areas in general and the database technology in particular. Nested structures are efficiently modeled by nested named sets [15]. Thus, one more interesting direction for future research is to study nested named sets and operations in their domain with the goal of their employment in database operation and control.

**Author Contributions:** M.B. and H.P.Z. contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Gurevich, Y. The Church-Turing Thesis: Story and Recent Progress. 2009. Available online: https://youtu.be/7XfA5EhH7Bc (accessed on 21 January 2022).
2. Angles, R.; Gutierrez, C. Survey of Graph Database Models, ACM Computing Surveys. *ACM Comput. Surv.* **2008**, *40*, 1. [CrossRef]
3. Codd, E.F. Data Models in Database Management. In Proceedings of the 1980 Workshop on Data Abstraction, Databases, and Conceptual Modeling, Pingree Park, CO, USA, 23–26 June 1980; ACM Press: New York, NY, USA, 1980; pp. 112–114.
4. Tsichritzis, D.C.; Lochovsky, F.H. Hierarchical data-base management: A survey. *ACM Comput. Surv.* **1976**, *8*, 105–123. [CrossRef]
5. Codd, E.F. A relational model of data for large shared data banks. *Commun. ACM* **1970**, *13*, 377–387. [CrossRef]
6. Zhuge, H. The Web Resource Space Model. In *Web Information Systems Engineering and Internet Technologies Book Series*; Springer: Berlin/Heidelberg, Germany, 2008; Volume 4, p. 2.
7. Kim, W. Object-oriented databases: Definition and research directions. *IEEE Trans. Knowl. Data Eng.* **1990**, *2*, 327–341. [CrossRef]
8. Buneman, P. Semistructured Data. In Proceedings of the 16th Symposium on Principles of Database Systems (PODS), Tucson, AZ, USA, 12–14 May 1997; ACM Press: New York, NY, USA, 1997; pp. 117–121.
9. Bray, T.; Paoli, J.; Sperberg-Mcqueen, C.M. Extensible Markup Language (XML) 1.0, W3C Recommendation 10. 1998. Available online: http://www.w3.org/TR/1998/REC-xml-19980210 (accessed on 11 November 2021).
10. Burgin, M. Structural Organization of Temporal Databases. In Proceedings of the International Conference on Software Engineering and Data Engineering (SEDE-2008), Las Vegas, NV, USA, 14–17 July 2008; ISCA: Los Angeles, CA, USA, 2008; pp. 68–73.
11. Burgin, M. *Theory of Named Sets*; Mathematics Research Developments; Nova Science: New York, NY, USA, 2011.
12. Burgin, M. *Theory of Knowledge*; World Scientific: New York, NY, USA; London, UK; Singapore, 2016.
13. Date, C.J. *An Introduction to Database Systems*; Addison Wesley: Boston, MA, USA; San Francisco, CA, USA; New York, NY, USA, 2004.
14. Burgin, M.; Zellweger, P. A Unified Approach to Data Representation. In Proceedings of the 2005 International Conference on Foundations of Computer Science, Las Vegas, NV, USA, 27–30 June 2005; CSREA Press: Las Vegas, NV, USA, 2005; pp. 3–9.
15. Burgin, M.; Zellweger, P. Nested Named Sets in Information Retrieval. In *Advances in Data Science and Information Engineering*; Springer: New York, NY, USA, 2021; pp. 451–467.

16. Nocedal, A.S.; Gerrikagoitia Arrien, J.K.; Burgin, M. A mathematical model for managing XML data. *Int. J. Metadata Semant. Ontol. (IJMSO)* **2011**, *6*, 56–73. [CrossRef]
17. Zellweger, H.P. Tree Visualizations in Structured Data Recursively Defined by the Aleph Data Relation. In Proceedings of the 20th International Conference Information Visualisation, Lisbon, Portugal, 19–22 July 2016; Volume 4, pp. 21–26.

proceedings

MDPI

Proceeding Paper

# Symbiotic Information Processing and Technological Progress †

**Mark Burgin [1,*] and Rao Mikkilineni [2]**

[1] Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA
[2] School of Business, Golden Gate University, San Francisco, CA 94105, USA; rmikkilineni@ggu.edu
* Correspondence: markburg@cs.ucla.edu
† Presented at Philosophy and Computing Conference, IS4SI Summit 2021, online, 12–19 September 2021.

**Abstract:** The goal of the paper is the introduction and exploration of new types of information processing. Starting with the typology developed in such an important class of information processing as computation, we extend this typology by analyzing information representations used in computational processes and delineating novel forms of information representations. While the traditional approach deals only with two dimensions of information processing—symbolic and sub-symbolic, our analysis explicated one more dimension—super-symbolic information processing. Information processing in biological systems is both symbolic and sub-symbolic, having the form of genes and neural networks. Nevertheless, in their evolution, biological systems have advanced their abilities one step further by developing super-symbolic information processing and evolving symbiotic information processing that performs information processing on the combined knowledge in the brain from both symbolic, subsymbolic, and super-symbolic information processing to derive higher order autopoietic and cognitive behaviors. Performing all forms of information processing, biological systems achieve much higher cognitive and intelligence level. That is why here we also consider a new type of computing automata called structural machines with the goal of transferring these advantageous features of biological systems to the existing information processing technology.

**Keywords:** information; computing; process; symbolic; subsymbolic; super-symbolic; structure; superstructure; structural machine

**Citation:** Burgin, M.; Mikkilineni, R. Symbiotic Information Processing and Technological Progress. *Proceedings* **2022**, *81*, 84. https://doi.org/10.3390/proceedings2022081084

Academic Editor: Peter Boltuc

Published: 28 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Information processing has many forms, types and categories, which are differentiated according to specific characteristics of information processing. Infware, that is, objects that are carriers and representations of the processed information, form an imperative constituent of information processing in general and computation in particular. Based on the computational infware, traditionally two pure forms of conventional computations are taken into consideration—symbolic computations and sub-symbolic computations [1]. With respect to infware, both are pure types of computations, while existing and new combined or amalgamated types and forms of computations are studied later.

Here we study these and more advanced types in the scope of the general information processing, assuming that computation does not encompass all types and forms of information processing.

*Symbolic information processing* is performed with data having the form of explicit symbolic systems, such as systems of numbers, icons, or letters, and the information processing system operates with individual symbols. Here, symbols are defined as linguistic objects, such as digits 0 and 1 or letters $a$ and $b$, understanding that these linguistic symbols are only names of symbols in a more general philosophical sense [2].

*Sub-symbolic information processing* is performed as transformations of data described by non-linguistic objects such as specific signs, signals, or geometric relationships. Note that sub-symbolic information processing can be modeled by symbolic information processing.

In both cases, the transformed entities are represented by the states of the information processing system and by the states of its elements. Thus, to portray and study information processing systems and their functioning, researchers use various mathematical models of information processing, such as recursive functions or Turing machines, which work with separate symbols.

Sub-symbolic (intuitive) information processing means that the information processing system uses elementary operations with concealed semantics. This kind of information processing is performed by such theoretical models as artificial neural networks and cellular automata. Sub-symbolic (intuitive) information processing allows the elimination of explicit algorithms/programs and using instead optimization processes, which improve functioning of the system by upgrading its implicit algorithms and programs.

Sub-symbolic (intuitive) information processing is realized by the neural ensembles in the brain. Researchers of cognition conjecture that the object formation can function as the transition from a stream of massively parallel sub-symbolic micro-functional events to symbol-type serial processing through sub-symbolic integration [3]. Sub-symbolic (intuitive) information processing is a model of functioning of the emotional and effective systems of the human brain [4].

Symbolic (rational) information processing means that the information processing system uses elementary operations with elementary objects having individual semantics. This kind of information processing is performed by such theoretical models as Turing machines, inductive Turing machines, and vector machines [5]. Symbolic (rational) information processing is a model of functioning of the left hemisphere of the human brain. The advantage of the symbolic (rational) information processing is the explicit form of the algorithms and programs that control and direct the functioning of the system.

## 2. A New Dimension of Information Processing

Symbolic and sub-symbolic information processing form two dimensions in the space of information processes. At the same time, the general theory of structures and brain neurophysiology point to one more pure type of information processing. Indeed, if there is sub-symbolic information processing, then it must be *super-symbolic information processing*, in which superstructures are transformed.

In information technology, supercomputers are computers that have essentially better characteristics of information processing, and usually it is the higher speed of computing. In a similar way, superstructures are structures that have essentially higher complexity.

Researchers try to model functioning of the brain using artificial neural networks. It is possible to compare this to a situation when, using only functioning of biological cells, biologists would try to explain the multifaceted functioning of the human organism with its higher functions.

Super-symbolic (transcendent) computing is a model of functioning of the right hemisphere of the brain. The processing of images by operation with holistic shapes is an example of super-symbolic computing. The advantage of the super-symbolic (transcendent) computing is its ability to operate big formal and informal systems of data and knowledge. Implementation of super-symbolic computing is the solution to the problem of big data and information overflow.

Symbolic structures are composed from symbols in a simple way, that is, these structures have low structural complexity. Symbols, words, texts as linear composition of words, and sets are symbolic structures.

According to the general theory of structures, there is a hierarchy of structures composed of different orders of structures [2].

Symbolic superstructures are composed from symbols and symbolic structures. Intricate hypertexts, multicomponent images, and structures of higher order are symbolic superstructures.

## 3. The Three-Dimensional Picture of Information Processing

The combination of pure types produces mixed types of information processing. The first step in this direction gives us *hybrid information processing*, which comprises both symbolic and sub-symbolic information processing being a two-fold type of information processing and encompassing hybrid computations [1]. Hybrid information processing allows combining advantages of both symbolic and sub-symbolic information processing.

Conventional models of computation perform either symbolic information processing, e.g., finite automata, Turing machines, inductive Turing machines or random-access machines (RAM), or sub-symbolic information processing, e.g., neural networks or cellular automata [5]. New models, such as neural Turing machines [6] or structural machines with symbolic and sub-symbolic processors [7], carry out hybrid information processing. A *neural Turing machine* is a recurrent neural network with a network controller connected to external memory resources. As a result, it combines the sub-symbolic computation of neural networks with symbolic computation of Turing machines.

Super-symbolic (intuitive) information processing adds one more dimension to the general schema. Synthesizing it with symbolic (rational) information processing and sub-symbolic (intuitive) information processing in one model, we come to *symbiotic information processing*. Structural machines with flexible types of processors can accomplish symbiotic information processing. Symbiotic information processing allows combining advantages of all three pure types of information processing representing the entire type of information processing.

As a result, we have:

**Three pure types:**

- Sub-symbolic information processing;
- Symbolic information processing;
- Super-symbolic information processing.

**Three twofold types:**

- *Hybrid information processing* combines symbolic and sub-symbolic information processing.
- *Blended information processing* combines sub-symbolic and super-symbolic information processing.
- *Fused information processing* combines symbolic and super-symbolic information processing.

**One entire type of information processing:**

- *Symbiotic information* processing combines all three pure types of information processing.

## 4. Structural Machines as a Tool for Symbiotic Information Processing

Structural machines provide means for symbiotic information processing when they possess processors of different types [7].

A structural machine M works with structures of a given type and has three components:

**The *control device*$C_M$ regulates the state of the machine *M***

The *processor* $P_M$ performs transformation of the processed structures and its actions (operations) depend on the state of the machine *M* and the state of the processed structures.

The *functional space* $Sp_M$ consists of three components:

- The *input space* $In_M$, which contains the input structure.
- The *output space* $Out_M$, which contains the output structure.
- The *processing space* $PS_M$, in which the input structure(s) is transformed into the output structure(s).

We assume that all structures—the input structure, the output structure, and the processed structures—are of the same type.

Computation of a structural machine *M* determines the *trajectory of computation*, which is a tree in a general case and a sequence when the computation is deterministic, and is performed by a single processor unit.

## 5. Modeling and Implementing Autopoietic and Cognitive Behaviors with Structural Machines

All living organisms are autopoietic and cognitive. Autopoiesis refers to a system with well-defined identity, which is capable of reproducing and maintaining itself. Cognition, on the other hand, is the process of information acquisition. A living organism is a unique autonomous system made up of components and relationships changing over time without changing the unity of the system. The genome contains the knowledge that is required to build the components using physical and chemical processes and physical resources. Information processing structures in the form of genes and neurons provide the means for building, operating, and managing the stability of the system while interacting with the external world where the interactions are often, non-deterministic in nature and are subject to large fluctuations. Our understanding of how theses information processing structures operate comes from the analysis of the genome, experiments in neuroscience, and the studies of cognitive behaviors in living organisms.

In the language of the General Theory of Information (GTI) [4], a genome encapsulates "knowledge structures" [7] coded in the form of DNA and executed using the biological "structural machines" [8–10] in the mode of genes and neurons, which utilize physical and chemical processes (dealing with conversion of matter and energy). The information accumulated through biological evolution is encoded into knowledge to create the genome, which contains the knowledge network defining the function, structure, and the autopoietic and cognitive processes to build and evolve the system while managing both deterministic and non-deterministic fluctuations in the interactions among the internal components or their interactions with the environment.

This knowledge about the genome and the GTI leads us to the Burgin–Mikkilineni [11] thesis that allows designing a new class of digital automata by infusing autopoietic and cognitive behaviors. Being a form and component of autopoietic and cognitive information processing systems, the digital genome is a collection of "knowledge structures" [8] coded in the executable form to be processed with technical "structural machines" [9] implemented using digital genes (in the form of symbolic computing algorithms) and digital neurons (in the form of sub-symbolic neural network algorithms), both of which use stored program control implementation of Turing machines and more advanced automata. Figure 1 shows super-symbolic computing in the shape of both biological and digital computing machines. DNA uses together symbolic computing with the sequences of nucleotides A, T, C, G, and subsymbolic computing with neural networks. The digital genome uses symbolic computing with algorithms operating on sequences of symbols 0 and 1, and sub-symbolic computing using neural network algorithms. The super-symbolic computing in biology is performed by the neocortex managing both the networks of genes and neurons. The autopoietic behavior is controlled by the genes and cognitive behaviors are controlled by the nervous system with 4E cognition. The digital genome proposed here uses the structural machine as shown in Figure 1 to create a super-symbolic overlay with knowledge structures managing both symbolic and sub-symbolic computations using current state of the art.

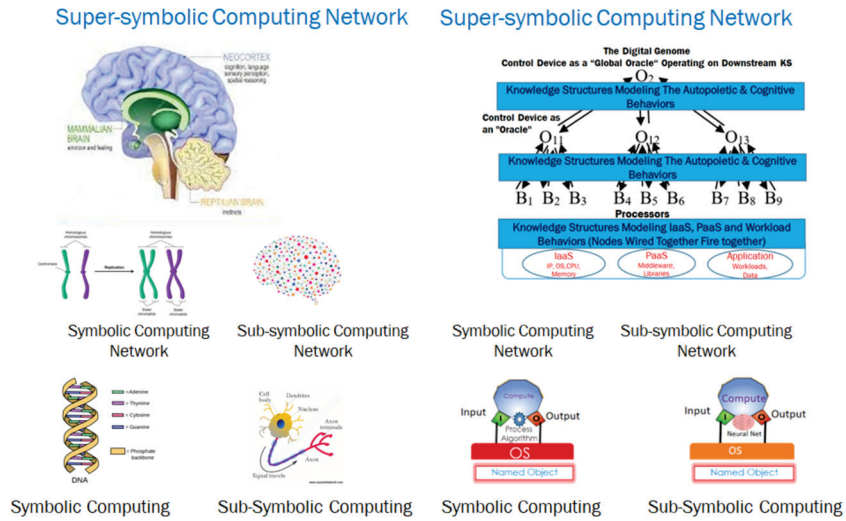



**Figure 1.** Super-symbolic computing in biological and digital computing.

## 6. Conclusions

We have introduced and discussed a new pure type of information processing called super-symbolic information processing; three twofold types of information processing called hybrid, blended, and fused information processing; and one entire type of computations called symbiotic information processing. Symbiotic information processing combines advantages of subsymbolic, symbolic, and super-symbolic information processing aimed at the advancement of the current state of the art in information technology with higher order autopoietic and cognitive behaviors, as well as at the modeling of information processing in the mind. The structural and functional analysis of these forms of information processing is the further goal of this research.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data sharing is not applicable to this article.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M.; Dodig-Crnkovic, G. A Taxonomy of Computation and Information Architecture. In Proceedings of the 2015 European Conference on Software Architecture Workshops, Dubrovnik, Croatia, 7–11 September 2015; pp. 1–8.
2. Burgin, M. *Structural Reality*; Nova Science Publishers: New York, NY, USA, 2012.
3. Clark, A. *Microcognition: Philosophy, Cognitive Science, and Parallel Distributed Processing*; MIT Press: Cambridge, MA, USA, 1989.
4. Burgin, M. *Theory of Information*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
5. Burgin, M. *Super-Recursive Algorithms*; Springer: New York, NY, USA; Heidelberg/Berlin, Germany, 2005.
6. Graves, A.; Wayne, G.; Danihelka, I. Neural Turing Machines. *arXiv* **2014**, arXiv:1410.5401.
7. Burgin, M.; Adamatzky, A. Structural machines and slime mold computation. *Int. J. Gen. Syst.* **2017**, *45*, 201–224. [CrossRef]
8. Burgin, M. Triadic Automata and Machines as Information Transformers. *Information* **2020**, *11*, 102. [CrossRef]
9. Burgin, M.; Mikkilineni, R.; Phalke, V. Autopoietic Computing Systems and Triadic Automata: The Theory and Practice. *Adv. Comput. Commun.* **2020**, *1*, 16–35. [CrossRef]
10. Burgin, M.; Mikkilineni, R. From Data Processing to Knowledge Processing: Working with Operational Schemas by Autopoietic Machines. *Big Data Cogn. Comput.* **2021**, *5*, 13. [CrossRef]
11. Burgin, M.; Mikkilineni, R. On the Autopoietic and Cognitive Behavior, EasyChair Preprint no. 6261, Version 2. 2021. Available online: https://easychair.org/publications/preprint/tkjk (accessed on 17 February 2022).

proceedings

Proceeding Paper

# Pollution Information Acquisition and Discovering Pollution Sources †

**Mark Burgin [1,*] and Arkadiy Dantsker [2,*]**

[1] Department of Computer Science, UCLA, Los Angeles, CA 90095, USA
[2] Detex Analytics, Kirkland, WA 98034, USA
* Correspondence: markburg@cs.ucla.edu (M.B.); amdantsker@gmail.com (A.D.)
† Presented at the Conference on Theoretical and Foundational Problems in Information Studies, IS4SI Summit 2021, online, 12–19 September 2021.

**Abstract:** This paper describes analytical methods used for the elaboration of the fast algorithm for detecting unknown sources of contamination release into water and/or air network systems. In addition, an analysis of existing theoretical approaches to computing pollutants released into water parameters is given and the necessity for a new methodology to solve multiple problems related to the real-time source identification is demonstrated.

**Keywords:** information; pollution; source; unknown; hypernumber; liquid; contaminant

**Citation:** Burgin, M.; Dantsker, A. Pollution Information Acquisition and Discovering Pollution Sources. *Proceedings* **2022**, *81*, 74. https://doi.org/10.3390/proceedings2022081074

Academic Editor: Rao Mikilineni

Published: 24 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

The contaminant leakage into liquid or gas systems is a frequent event. Multiple publications disclose information related to the release of contaminants such as bacteria, heavy metals, PFAS (per- and polyfluoroalkyl substances) and many others from unknown sources. The bacterial contaminant release is a major cause of the beach closure. Per ACS publication, nearly all (93%) of the Los Angeles and Orange County advisories and closures were caused by unknown sources of fecal indicator bacteria (FIB). The PFAS contamination is another critical problem [1]. According to the Nordic Co-operation published socioeconomic analysis of environmental and human health impacts from exposure to PFAS estimates that EEA member countries incur EUR 52 to 84 billion in PFAS-related health costs annually. Similar to the bacterial contamination, there is a significant number of cases when the sources of PFAS contamination are unknown.

Finding a fast algorithm for detecting unknown sources of pollution is critical to solving many problems such as:

1. *Detecting the source of contamination in potable water systems.*
2. *Identifying the smoke source in ventilation systems.*
3. *Locating the source* of the explosive in air systems.
4. Detecting contamination in water streams while monitoring multiple streams from a remote location.

The approaches for detecting sources of pollution are covered in multiple publications [2–5].

One of these methods, which is discussed in publications [4,5], defines pollutant transport with a stochastic direct model and set $S$ of pollutant source parameters by solving equations of the inverse engineering mode State of the Art Report on Mathematical Methods for Groundwater Pollution Source Identification. The set $S$ of computed parameters for this ill-posted problem are provided below.

$$S = \{\text{Distance from sensor}, \text{ Rate of the discharge}, \text{ Coefficient of the dispersion}\}$$

The stochastic modeling allows us to extend the obtained solution for complex network systems. However, the numerical solution of the inverse model cannot be applied for real-time water security tasks and many other real-time processes.

Here, we describe a new direction for obtaining analytical solutions for the inverse model of the pollutant propagation using the theory of hypernumbers [6,7], which is a new field in functional analysis developed for rigorous and consistent operation with singular dynamics when integrals and series tend to infinity. Real hypernumbers form a natural extension of real numbers in the same way as real numbers extend rational numbers by topological constructions. In a similar way, complex hypernumbers are built from complex numbers.

In addition to the utilization in solving diverse differential, integral and functional equations, hypernumbers are efficiently used for the further development of the theory of the Feynman path integral and probability theory.

## 2. Defining Pollutant Source Location and Release Rate with the Theory of Hypernumbers

The stochastic approach leads to a simple integral formula for a pointed source. Furthermore, we can receive an analytical solution with this approach for any distributed source or for a set of pointed sources.

$$c = \int_0^t \frac{\mu e^{-\frac{(x-v\tau)^2}{2D\tau}}}{\sqrt{2\pi D\tau}} d\tau \tag{1}$$

where $c$—contaminant concentration; $x$—distance from the contamination source to the sensor; $D$—dispersion; $v$—contaminant transport velocity; $(\tau,\ t)$—time; and $\mu$ contaminant discharge rate.

$$U = \sum_{i=1}^{n} \left( c_i - \int_0^{t_i} \frac{\mu e^{-\frac{(x-v\tau)^2}{2D\tau}}}{\sqrt{2\pi D\tau}} d\tau \right)^2 \tag{2}$$

where $U$ is the operator defining distance between monitoring and calculated contaminant's concentration and $c_i$ is the concentration at time $t_i$.

The $U$ minimization requirement must satisfy conditions (3)–(5).

$$F_1 = \frac{\partial U}{\partial x} = 2\sum_{i=1}^{n} \left( c_i - \int_0^{t_i} \frac{\mu e^{-\frac{(x-v\tau)^2}{2D\tau}}}{\sqrt{2\pi D\tau}} d\tau \right) \int_0^{t_i} \frac{\mu e^{-\frac{(x-v\tau)^2}{2D\tau}} (x - v\tau)}{\sqrt{2\pi D\tau}\, D\tau} d\tau = 0 \tag{3}$$

$$F_2 = \frac{\partial U}{\partial D} = 2\sum_{i=1}^{n} \left( c_i - \int_0^{t_i} \frac{\mu e^{-\frac{(x-v\tau)^2}{2D\tau}}}{\sqrt{2\pi D\tau}} d\tau \right) \int_0^{t_i} \frac{-\mu e^{-\frac{(x-v\tau)^2}{2D\tau}} (x - v\tau)^2}{2\sqrt{2\pi D\tau}(D\tau)^2} d\tau = 0 \tag{4}$$

This expression is simplified by the following replacement:

$$I_1 = \int_0^{t_i} \frac{\mu e^{-\frac{(x-v\tau)^2}{2D\tau}}}{\sqrt{2\pi D\tau}} d\tau,\ I_2 = \int_0^{t_i} \frac{\mu e^{-\frac{(x-v\tau)^2}{2D\tau}} (x - v\tau)}{\sqrt{2\pi D\tau}\, D},\ I_3 = \int_0^{t_i} \frac{-\mu e^{-\frac{(x-v\tau)^2}{2D\tau}} (x - v\tau)^2}{2\sqrt{2\pi D\tau}(D\tau)^2} d\tau \tag{5}$$

Replacing (3) and (4) with the expressions (3)–(5),we receive Equations (6), (8) and (9).

$$\frac{\partial U}{\partial \mu} = 2\sum_{i=1}^{n} (c_i - \mu I_1) I_1 = 2\sum_{i=1}^{n} c_i I_1 - \mu \sum_{i=1}^{n} I_1^2 = 0 \tag{6}$$

$$\mu = \frac{\sum_{i=1}^{n} c_i I_1}{\sum_{i=1}^{n} I_1^2} \tag{7}$$

$$\frac{\partial U}{\partial x} = 2\sum_{i=1}^{n}\left(c_i - \frac{\sum_{i=1}^{n} c_i I_1}{\sum_{i=1}^{n} I_1^2} I_1\right) I_2 \tag{8}$$

$$\frac{\partial U}{\partial D} = 2\sum_{i=1}^{n}\left(c_i - \frac{\sum_{i=1}^{n} c_i I_1}{\sum_{i=1}^{n} I_1^2} I_1\right) I_3 \tag{9}$$

Following the methods from the theory of hypernumbers for solving non-linear operator equations [8,9], we determine the solutions for the source distance and dispersion at (10)–(13).

$$x_m = H_n(x_m)_{m\in\omega} \tag{10}$$

$$D_m = H_n(D_m)_{m\in\omega} \tag{11}$$

$$x_{m+1} = x_m + \delta(x)_m \tag{12}$$

$$D_{m+1} = D_m + \delta(D)_m \tag{13}$$

The deviations of hypernumbers $x_m$ and $D_m$ are computed using Formulas (14) and (15).

$$\delta F_{1,m+1} = \delta(x)_m d_{11} + \delta(D)_m d_{12} = (1-\theta)F_{1,m} \tag{14}$$

$$\delta F_{2,m+1} = \delta(x)_m d_{21} + \delta(D)_m d_{22} = (1-\theta)F_{2,m} \tag{15}$$

The formula for defining $d_{11}$ is given below.

$$d_{11} = \frac{\partial^2 U}{\partial x^2}\, 2\sum_{i=1}^{n}\left(c_i - \frac{\sum_{i=1}^{n} c_i I_1}{\sum_{i=1}^{n} I_1^2} I_1\right)\frac{dI_2}{dx} 2\sum_{i=1}^{n}\left(\frac{d\left(\frac{\sum_{i=1}^{n} c_i I_1}{\sum_{i=1}^{n} I_1^2}\right)}{dx} I_1 + \frac{\sum_{i=1}^{n} c_i I_1}{\sum_{i=1}^{n} I_1^2}\frac{dI_1}{dx}\right) I_2 \tag{16}$$

A similar computational approach is utilized to find the rest of the coefficients. Using Equations (1)–(16), we created a software for computing the pollutant discharge parameter.

## 3. Conclusions

The described algorithms are used for the development of the Pollution Positioning System (PPS), which detects location of pollution sources with the great precision, as well as for building the Contamination Positioning System (CPS), which efficiently and accurately finds critically contaminated places and regions.

Besides, the defined solution for locating the source of the pollution discharge into the liquid or gas network system will allow us to extend the applicability of the suggested approach for a variety of critical problems related to the security and technological process optimization. It is possible to use this approach to solve several other important problems such as locating the place of the liquid and gas leakage in underground pipe network systems or locating the source of the pollution with pollutant decay during its transport. The latter problem would require a new mathematical description of the stochastic contaminant transport model.

In addition, the suggested approach opens ways for utilization of real extrafunctions for solving even more serious ecological and technological problems.

**Author Contributions:** M.B. and A.D. contribute equally. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References


1. Benotti, M.J.; Fernandez, L.A.; Peaslee, G.F.; Douglas, G.S.; Uhler, A.D.; Emsbo-Mattingly, S. A forensic approach for distinguishing PFAS materials. *Environ. Forensics* **2020**, *21*, 319–333. [CrossRef]
2. Atmadja, J.; Amvrossios, C.; Bagtzoglout, A.C. State of the Art Report on Mathematical Methods for Groundwater Pollution Source Identification. *Environ. Forensics* **2001**, *2*, 205–214. [CrossRef]
3. Dantsker, A.; Johnson, B.; Titov, V. Inverse Problem of Toxic Waste Spreading. In Proceedings of the Workshop on Advanced Technologies in Real Time Monitoring and Mod for Drinking Water Safety and Security, Newark, NJ, USA, 27–28 June 2002.
4. Dantsker, A.; Davis, J.; Manvelyan, O.; Solyanik, V.; Titov, V. *Water System Network Algebra*; Wit Press: Southampton, UK, 2003; Volume 8, pp. 255–264.
5. Yan, X.; Zhu, Z.; Li, T. Pollution source localization in an urban water supply network based on dynamic water demand. *Environ. Sci. Pollut. Res.* **2019**, *26*, 17901–17910. [CrossRef] [PubMed]
6. Burgin, M. Nonlinear Partial Differential Equations in Extrafunctions. *Integr.: Math. Theory Appl.* **2010**, *2*, 17–50.
7. Burgin, M. *Hypernumbers and Extrafunctions: Extending the Classical Calculus*; Springer: New York, NY, USA, 2012.
8. Burgin, M.; Dantsker, A.M. A method of solving operator equations by means of the theory of hypernumbers. *Not. Nat. Acad. Sci. Ukr.* **1995**, *8*, 27–30. (In Russian)
9. Burgin, M.; Dantsker, A.M. *Real-Time Inverse Modeling of Control Systems Using Hypernumbers, in Functional Analysis and Probability*; Nova Science Publishers: New York, NY, USA, 2015; pp. 439–456.

proceedings

*Editorial*

# Symposium Theoretical Information Studies †

**Mark Burgin**

Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA; markburg@cs.ucla.edu

† Presented at the IS4SI 2017 Summit DIGITALISATION FOR A SUSTAINABLE SOCIETY, Gothenburg, Sweden, 12–16 June 2017.

Published: 3 August 2017

---

Information has always been important and sometimes vital for people but now it has become the most valuable asset and the strongest moving force in the contemporary society. Natural science is a courageous human endeavor in getting information about the universe. Social science is an indispensable human enterprise in getting information about society. Information is the bread and butter of mass media. Everything in the human organism is controlled by flows of information. Information processing and communication systems, such as computers, the Internet, and cell phones, has become the core of the modern society. Thus, it is natural that our time is called the information age.

Information studies constitute the central cognitive part of the information facet of the contemporary society, which form the intertwined system of the following parts:

- Information studies
- Information technology
- Information activity

Any utilization of information technology is information activity. At the same time, there are also other kinds of information activity such as writing books or music, communication with or without technological devices, teaching and researching.

It is practical to divide the whole area of information studies into three components:

- Theoretical information studies
- Experimental information studies
- Practical information studies

At the same time, intellectual information studies also have three branches:

- Information theory
- Philosophy, methodology and logic of information and information studies
- Foundations of information studies

Practical information studies include application of information theory but also contain empirical research in the area of information, communication and information processing systems.

Experimental information studies are aimed at discovery of regularities of information functioning and structure, as well as at experimental validation of theoretical results.

Information theory constitutes the basic component of theoretical information studies but there are also other components. One of them is application of information studies to the theoretical areas of science and humanities, e.g., application to theoretical physics or to theoretical computer science.

Therefore, the areas of interest to the Symposium included (but were not confined to):

- Information theory
- Application of information theory to theoretical physics
- Application of information theory to theoretical biology

- Application of information theory to theoretical computer science
- Application of information theory to the theory of complexity
- Application of information theory to general systems theory
- Application of information theory to mathematics
- Application of information theory to theoretical linguistics
- Applications of information theory in economics
- Application of information theory to pedagogy
- Applications of information theory in sociology
- Applications of information theory in psychology
- Applications of information theory in anthropology
- Application of information theory to semiotics

People begin to understand that information is everywhere and if we can know anything about ourselves, other people and the world in which we live, this is entirely due our ability to obtain and process information. In essence, any science intrinsically depends on information acquisition and procession. For instance, the book Physics from Fisher Information [1] persuasively demonstrates how it is possible to deduce basic physical laws from definite regularities in the world of information. The book *The Touchstone of Life: Molecular Information, Cell Communication, and the Foundation of Life* [2] colorfully describes how all biological processes are rooted in and based on information.

In his proclamation It from Bit, the famous physicist John Wheeler makes even a stronger claim stating that everything in nature comes from information [3]. Moreover, it is discovered that information is the basic essence and the driving force not only in the physical universe but also in mental and structural worlds encompassing the entire reality as its unifying factor.

This situation clearly demonstrates that traditional scientific cognition (in physics, chemistry, biology, etc.) needs the concept of information and methods of information science for its development and proliferation.

Theory of information became popular with the contribution of Claude Shannon to communication technology [3]. Since that time, researchers have developed many other directions in information theory. The most notable of them are algorithmic information theory [4], semantic information theory [5] and economic information theory [6]. The most fundamental achievement in this area is creation of the general theory of information, which gives a flexible, efficient and all-encompassing definition of information comprising all other information theories in a unified system [7]. As a result, the general theory of information becomes a unified theory of information, which has been one of the most aspiring goals in the area of information studies [8].

There are also essential achievements in philosophy of information [9] and in foundations of information science [10]. New areas of information studies emerged such as evolutionary information theory [11] and information ecology [12,13]. As always, new discoveries open new problems and bring new challenges for the intellectual inquiry, scientific exploration of reality, social organization and technological innovations [14–17].

The Symposium Theoretical Information Studies was a part of the International Society for the Study of Information (is4si) 2017 Summit at Gothenburg, Sweden held at Chalmers Institute of Technology 12–16 June 2017 with the overarching theme: DIGITALISATION FOR A SUSTAINABLE SOCIETY: Embodied, Embedded, Networked, Empowered through Information, Computation & Cognition.

The goal of the Symposium was to bring together academics and researchers providing beneficial conditions for presenting and discussing recent achievements and problems of information theory and its applications to theoretical issues of science and humanities. More than 30 researchers from 15 countries and 5 continents participated in the Symposium presenting their discoveries in the area of information and discussing new achievements and critical problems in information science. The Summit in general and the Symposium in particular provided beneficial conditions for interaction of academics and other researchers with representative of information and communication technology.

The contributions that follow this note are short versions of papers presented at the Symposium. We hope they will grow into extensive studies published in the other venues representing the Summit.

## References


1. Frieden, R.B. *Physics from Fisher Information*; Cambridge University Press: Cambridge, UK, 1998
2. Loewenstein, W.R. *The Touchstone of Life: Molecular Information, Cell Communication, and the Foundation of Life*; Oxford University Press: Oxford, UK; New York, NY, USA, 1999.
3. Shannon, C.E. The Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423.
4. Chaitin, G.J. Algorithmic information theory. *IBM J. Res. Dev.* **1977**, *21*, 350–359.
5. Bar-Hillel, Y.; Carnap, R. Semantic Information. *Br. J. Philosophical Sci.* **1958**, *4*, 147–157.
6. Arrow, K.J.; Marschak, J.; Harris, T. Economics of information and organization: Optimal inventory policy. *Econometrica* **1951**, *19*, 250–272.
7. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
8. Hofkirchner, W. (Ed.) The Quest for a Unified Theory of Information. In Proceedings of the Second International Conference on the Foundations of Information Science, Zurich, Switzerland, 10–12 March 1999.
9. Dodig-Crnkovic, G. Shifting the Paradigm of Philosophy of Science: Philosophy of Information and a New Renaissance. *Minds Mach.* **2003**, *13*, 521–536.
10. Marijuán, P.C. Foundations of Information Science: Selected papers from FIS 2002. *Entropy* **2003**, *5*, 214–219.
11. Burgin, M. Evolutionary Information Theory. *Information* **2013**, *4*, 224–268.
12. Capurro, R. *Towards an Information Ecology, in Information and Quality*; Taylor Graham: London, UK, 1990; pp. 122–139, 288.
13. Zhong, Y.X. The Law of "Information Conversion and Intelligence Creation". In *Information Studies and the Quest for Transdisciplinarity*; World Scientific: New York, NY, USA; London, UK; Singapore, 2017; pp. 165–190.
14. Calude, C.S. Algorithmic information theory: Open problems. *J. Univ. Comput. Sci.* **1996**, *2*, 439–441.
15. Raber, D. *The Problem of Information*: *An Introduction to Information Science*; Scarecrow: Lanham, MD, USA, 2003.
16. Burgin, M. Information: Problems, Paradoxes, and Solutions. *TripleC* **2003**, *1*, 53–70.
17. Adriaans, P. Open Problems in the Study of Information and Computation. The Stanford Encyclopedia of Philosophy (Fall 2013 Edition). Zalta, E.N., Ed. Available online: https://plato.stanford.edu/archives/fall2013/entries/information/supplement.html (accessed on 7 July 2017).


© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

Editorial

# IS4SI SUMMIT Berkeley 2019 †

**Marcin J. Schroeder [1], GordanaDodig-Crnkovic [2] and Mark Burgin [3,*]**

[1] Global Learning Center, Tohoku University, Sendai, Miyagi Prefecture, 980-8576, Japan; schroeder.marcin.e4@tohoku.ac.jp
[2] Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg 40482, Sweden; gordana.dodig-crnkovic@chalmers.se
[3] Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA; mburgin@math.ucla.edu
* Correspondence: mburgin@math.ucla.edu

† 2019 Summit of the International Society for the Study of Information, Berkeley, CA, USA, 2–6 June 2019.

Published: 12 May 2020

---

The leading theme of the 2019 Summit of the International Society for the Study of Information, held 2–6 June 2019 at the University of California at Berkeley, was the question "Where is the I in AI, and the meaning of Information?" This question addresses one of the central issues not only for scientific research and philosophical reflection, but also for technological, economic and social practice.

There is no doubt that the progress in designing computer based technological devices known as Artificial Intelligence (AI) systems has transformed the life and work of almost the entire human population. Moreover, this transformation continues and accelerates. At the same time, there is no one commonly accepted definition of AI. This is a reflection of the fact that there is no commonly accepted definition of intelligence in general. Moreover, there is no agreement about human intelligence, and psychologists and cognitive scientists frequently opt for multiple intelligences because, in essence, intelligence is a contextual trait. Thus, an important goal of researchers is to more definitely qualify the natural intelligence of people and maybe other living beings, making it possible to understand whether artificial intelligence, i.e., the intelligence of a machine, is the same, and how it is possible to estimate this intelligence.

In addition, there are also ethical issues in this area. AI is based on technology, and we can already see the great benefits of accelerating technological transformation, but also some of its negative, destructive or even dangerous consequences. In order to make the best use of already existing technologies and to avoid their negative consequences, as well as to assure the development of AI for good, we have to understand well their mechanisms and their interactions with humanity and natural environment. This is a surprisingly difficult task, especially for the situation when machines will be able to achieve superintelligence, i.e., higher intelligence than the intelligence of people.

The second part of the question, which was the leading theme of the Summit in Berkeley, was about the meaning of information. This reflects the actuality that intelligence is based on information and information processes, while meaning is a pivotal characteristic of information. The current situation is characterized by the common objection to the ascription of intelligence to currently existing technological devices due to the doubt that these artefacts are capable of associating meaning with formally encoded information. Of course, this brings us back to the centuries of philosophical discourse on the mind–body relationship, which is still an open question and an objective of research and philosophical reflection which, in the case of AI, is transformed into the problem of the mind–machine relationship. The contemporary achievements of science, mathematics and philosophy position these problems in the context of the Existential Triad of the world, which consists of three worlds: physical, mental and structural worlds.

New achievements in the research in this area, reflecting different approaches to the described problems, were presented in the talks of the participants of the 2019 Summit, which contained many interesting ideas and appealing results.

The 2019 Summit was organized as an assembly of parallel conferences and workshops, which included on-site and off-site presentations. In this issue, we publish selected proceedings of the talks delivered at the following conferences:

- Conference Morphological, Natural, Analog and Other Unconventional Forms of Computing for Cognition and Intelligence (MORCOM) organized by Gordana Dodig-Crnkovic and Marcin J. Schroeder
- The Fourth International Conference on Philosophy of Information (POI) organized by Joseph Brenner and Kun Wu
- Conference Theoretical Information Studies (TIS) organized by Mark Burgin
- Workshop on Habits and Rituals (HAR) organized by Raffaela Giovagnoli and Gianfranco Basti

**Conflicts of Interest:** The authors declare no conflict of interest.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

*Editorial*

# Conference Theoretical Information Studies Berkeley 2019 †

**Mark Burgin**

Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095, USA; mburgin@math.ucla.edu

† Conference Theoretical Information Studies, Berkeley, California, USA, 2–6 June 2019.

Published: 13 May 2020

**Abstract:** This paper has a two-fold goal. In the first part, the area of theoretical information studies is described. In second part, contributions to the conference "Theoretical Information Studies" (Berkeley 2019) are characterized.

**Keywords:** information; general theory; special theory; artificial intelligence; intelligence; mathematics; physics; computer science

---

## 1. Information Studies as a Field of Research and Domain of Knowledge

Information studies encompass the following basic fields:

- Information science,
- Information philosophy,
- Information methodology, and
- Information logic.

There are also fields directly associated with information studies:

– Computer science is associated with information studies because it studies information processing by technical devices.
– Linguistics is associated with information studies it studies information processing by natural and artificial languages.
– Semiotics is associated with information studies it studies symbolic information processing.
– Psychology is associated with information studies it studies information processing by people.
– Pedagogy is associated with information studies it studies information transmission and knowledge acquisition.
– Artificial intelligence is associated with information studies because it studies information processing by artificial systems.

There are also fields which are conceptually associated with information studies, including:

– Information-oriented and information-based physical theories (cf. for example, [1–9]).
– Information-oriented and information-based biological theories (cf. for example, [10–18]).
– Information-oriented and information-based economic theories (cf. for example, [19–26]).
– Information-oriented and information-based sociological theories.
– Information-oriented and information-based anthropological theories.

Information science as any natural science consists of three components:

- Theoretical information science,
- Experimental information science, and
- Applied information science.

Information theory constitutes the basic part of theoretical information science, which also includes subtheoretical fragments, i.e., research ingredients, which are situated on a lower level than the level of theory. Although many think that information theory is Shannon's statistical information theory [27], which was originally called communication theory, now there are many information theories: statistical information theory [27], algorithmic information theory [28], semantic information theory [29], pragmatic information theory [11], economic information theory [30], qualitative information theory [31], and the general theory of information [32, 33]. There is a possibility that the elaboration of a unified theory of information would eliminate the necessity of all other information theories. However, this is not true. First, a unified theory of information has already been created and it is called the general theory of information [32]. Second, the role of this general theory is not the elimination of other special information theories but the unification of the scientific field called theoretical information studies in general and theoretical information science in particular. Other information theories become subtheories of the general theory of information, but they are necessary because they bring forth the more exact representation and study of information in different specific fields, provide more adequate means for solving specific problems, and allow a better understanding of the role of information in different spheres of the world.

Examples from mathematics and physics show that the existence of a general theory coexists with the active functioning of its subtheories. For instance, in mathematics the theory of groups contains subtheories—such as the theory of Abelian groups, the theory of nilpotent groups, and the theory of ordered groups—which are actively growing. In physics, Newton's dynamics coexists with Kepler's theory of planetary movement, which is a subtheory of Newton's dynamics.

The goal of experimental information science is to study information using physical and mental experiments. It is necessary to acknowledge that experimental information science is not sufficiently developed. Some experiments are conducted only by statistical information theory and algorithmic information theory. For instance, Shannon devised an experiment aimed at determining the amount of statistical information (entropy) in a given letter in English.

It is possible to classify applied or practical information science by the domain of its application. This gives us three classes:

Theoretical applications, i.e., applications of information science to the theoretical areas of science and humanities, such as physics, biology, or economics.

Philosophical applications, i.e., applications of information science to philosophical teachings or methodological systems.

Practical applications, i.e., applications of information science in practical areas such as engineering, linguistics, or cryptography.

These considerations allow us to discern the following areas of theoretical applications:

- Application of information theory to theoretical physics.
- Application of information theory to theoretical computer science.
- Application of information theory to the theory of complexity.
- Application of information theory to mathematics.
- Application of information theory to theoretical linguistics.
- Applications of information theory in economics.
- Application of information theory to pedagogy.
- Applications of information theory in sociology.
- Applications of information theory in anthropology.
- Application of information theory to semiotics.

## 2. Presentations at the Conference

The conference of Theoretical Information Studies was organized as a part of the IS4SI Summit 2019 in Berkeley, California, USA. It included on-site and off-site presentations. The goal of the conference was to bring together academics and researchers, providing beneficial conditions for presenting and discussing recent achievements and problems of information theory and its applications in theoretical issues in science and humanities. The participants came from nine countries and four continents.

Here, we publish the proceedings of the selected presentations from the conference of Theoretical Information Studies with innovative results and ideas, some of which are at the forefront of theoretical information science.

In their presentation, Gianfranco Basti, Antonio Capolupo, and Giuseppe Vitiello explored semantic information in quantum mechanics and quantum field theory.

In his presentation, Paul Benioff described how local mathematics and number scaling provide information about physical and geometric systems. His work is aimed at the unification of physics and mathematics.

In his presentation, Mark Burgin explicated and discussed two classes of information—potential and impact information. This approach is based on the inherent relations between physics and information theory.

In his presentation, Jaime F. Cárdenas-García presented his work with Timothy Ireland on analyzing and further developing Bateson's approach to information theory.

In their presentation, Gordana Dodig-Crnkovic and Mark Burgin analyzed the presentation of information in recent books, delineating the emergent academic field of the study of information.

In his presentation, Wolfgang Hofkirchner explored interrelations between people and artificial systems such as computers and computer networks.

In another presentation, Wolfgang Hofkirchner discussed the origin of system thinking.

In their presentation, Stefan Leijnen and Fjodor van Veen presented the taxonomy of neural networks, which have become an important and very popular tool in information processing with the advance of deep learning.

In his presentation, Vladimir Lerner analyzed the natural origin of information and the natural encoding of information.

In his presentation, Rafal Maciag described and analyzed the ontological basis of knowledge in the theory of discursive space and its consequences.

In his presentation, Marcin Schroeder described and scrutinized a variety of important theoretical tools for the study of information, such as equivalence and cryptomorphism.

In his presentation, Paul Zellweger described the Branching Data Model, which he developed for data management and information visualization using the theory of named sets.

In his presentation, Yixin Zhong analyzed concepts of information in the context of artificial intelligence research, specifying two types of information—ontological and epistemological information.

In his presentation, Rao Mikkilineni described the advancement of computation beyond the Church–Turing Thesis boundaries, which was based on structural machines, digital genes, and digital neurons being developed in the context of future AI.

In her presentation, Annette Grathoff explored the structural and kinetic components of physical information from an evolutionary perspective.

In their presentation, Mark Burgin and Kees De Vey Mestdagh discussed their research on the complexity of the information components of legal systems and processes.

In their presentation, Mark Burgin and Gordana Dodig-Crnković introduced and explored the novel scheme of modeling computing devices by information operators.

In his presentation, Jose Monserrat Neto described and analyzed the birth and evolution of symbolic information.

In another presentation, Jose Monserrat Neto considered the diversification of symbolic systems.

In his presentation, Mark Burgin outlined the problem-oriented foundations of intelligence in the context of superintelligence, suggesting a mathematical approach to the formalization of intelligence.

**Conflicts of Interest:** The author declares no conflict of interest.

**Funding:** This research received no external funding.

## References

1. Frieden, R.B. *Physics from Fisher Information*; Cambridge University Press, Cambridge, UK, 1998.
2. Harmuth, H.F. *Information Theory Applied to Space-Time Physics*, World Scientific, Singapore, 1992.
3. Chiribella, G.; D'Ariano, G.M.; Perinotti, P. Informational derivation of quantum theory. *Phys. Rev. A* **2011**, *84*, 012311.
4. Chiribella, G.; D'Ariano, G.M.; Perinotti, P. Quantum theory, namely the pure and reversible theory of information. *Entropy* **2012**, *14*, 1877–1893.
5. D'Ariano, G.M. Physics without physics: The power of information-theoretical principles. *Int. J. Theory Phys.* **2017**, *56*, 97–128.
6. Fuchs, C. Quantum mechanics as quantum information (and only a little more). *arXiv* **2002**, arXiv:quant-ph/0205039v1.
7. Goyal, P. Information physics—Towards a new conception of physical reality. *Information* **2012**, *3*, 567–594.
8. Lee, J.-W. Quantum mechanics emerges from information theory applied to causal horizons. *Found. Phys.* **2011**, *41*, 744–753.
9. Clifton, R.; Bub, J.; Halvorson, H. Characterizing quantum theory in terms of information-theoretic constraints. *Found. Phys*. **2003**, *33*, 1561–1591.
10. Pattee, H.H. The Physics of Autonomous Biological Information. *Biol. Theory*: *Integr. Dev. Evol. Cogn.* **2006**, *1*, 224–226.
11. Weinberger, E.D. A theory of pragmatic information and its application to the quasi-species model of biological evolution. *Biosystems* **2002**, *66*, 105–119.
12. Loewenstein, W.R. *The Touchstone of Life: Molecular Information, Cell Communication, and the Foundation of Life*; Oxford University Press: Oxford, UK; New York, NY, USA, 1999
13. Barbieri, M. *The Organic Codes*: *An Introduction to Semantic Biology*; Cambridge University Press: Cambridge, UK, 2003.
14. Burgin, M.; Simon, I. *Information, Energy, and Evolution*, electronic ed.; Preprint in Biology 2359, Cogprints; 2001. Available online: http://cogprints.ecs.soton.ac.uk (accessed on 12 May 2020).
15. Lerner, V.S. Information Macrodynamic Approach for Modeling in Biology and Medicine. *J. Biol. Syst.* **1997**, *5*, 215–264.
16. Lerner, V.S. Macrodynamic cooperative complexity of biosystems. *J. Biol. Syst.* **2006**, *14*, 131–68.
17. Lerner, V.S.; Talyanker, M.I. Informational Geometry for Biology and Environmental Applications. In Proceedings of the International Conference on Environmental Modeling and Simulation, San Diego, CA, USA, 10-14 January 2000; pp. 79–84.
18. Reinagel, P. Information theory in the brain. *Curr. Biol.* **2000**, *10*, 542–544.
19. Arrow, K.J. *Collected Papers of Kenneth J. Arrow*; Volume 4: The Economics of Information; Harvard University Pres: Cambridge, MA, USA, 1984.
20. Boisot, M.H. *Knowledge Assets: Securing Competitive Advantage in the Information Economy*; Oxford University Press: Oxford, UK, 1998.
21. Laffont, J.J. *The Economics of Uncertainty and Information*; Bonin, J.P., Bonin, H., Translator; MIT Press: Cambridge, MA, USA, 1989.
22. Lerner, V.S. An elementary information macrodynamic model of market economic system. *J. Inf. Sci.* **2006**, *176*, 3556–3590.
23. Marschak, J. Towards an economic theory of organization and information. In *Decision Processes*; New York, NY, USA, John Wiley, 1954; pp. 187–220.
24. Marschak, J. Economics of information systems. *J. Am. Stat. Assoc.* **1971**, *66*, 192–219.
25. Marschak, J. Economic information, decision, and prediction, Selected essays. Vol. II, Economics of information and organization. In *Theory and Decision Library*; D. Reidel Publishing Co.: Dordrecht, The Netherlands; Boston, MA, USA, 1980.

26. Stigler, G. J. The economics of information. *J. Polit. Econ.* **1961**, *69*, 213–225.
27. Shannon, C.E. The Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423
28. Chaitin, G.J. Algorithmic information theory. *IBM J. Res. Dev.* **1977**, *21*, 350–359.
29. Bar-Hillel, Y.; Carnap, R. Semantic Information. *Br. J. Philos. Sci.* **1958**, *4*, 147–157.
30. Arrow, K.J.; Marschak, J.; Harris, T. Economics of information and organization: Optimal inventory policy. *Econometrica* **1951**, *19*, 250–272.
31. Mazur, M. *Jakosciowa Teoria Informacji*; PAN: Warszawa, Poland, 1970. (In Polish)
32. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010.
33. Burgin, M. The General Theory of Information as a Unifying Factor for Information Studies: The noble eight-fold path. *Proceedings* **2017**, *1*, 164. doi:10.3390/IS4SI-2017-04044.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

MDPI

*Proceedings*

# Recent Books Delineating the Emergent Academic Filed of the Study of Information †

**Gordana Dodig-Crnkovic [1,*] and Mark Burgin [2]**

[1] Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg 40482, Sweden; gordana.dodig-crnkovic@chalmers.se
[2] Department of Computer Science, University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA; markburg@cs.ucla.edu
* Correspondence: gordana.dodig-crnkovic@chalmers.se
† Conference Theoretical Information Studies (TIS), Berkeley, CA, USA, 2–6 June 2019.

Published: 11 May 2020

**Abstract:** One of the imperative tasks of theoretical information studies is exploration of methods and techniques of presentation of information. In this article, we study the presentation of information in the following three books describing the results of the collaboration of researchers with the goal of defining the emergent filed of the Study of Information: *Philosophy and Methodology of Information*: *The Study of Information in a Transdisciplinary Perspective* (2019), *Theoretical Information Studies*: *Information in the World* (2020) and *Information and Computation* (2011), published by World Scientific Publishing Co. Series in Information Studies.

**Keywords:** study of information; information; computation; info-computation; theoretical information studies; methodology of information; transdisciplinarity

## 1. Introduction

An important problem in theoretical information studies is exploration of methods and techniques of presentation of information in general and specific kinds of information such as information about information studies in particular [1]. Here, we consider presentation of information about information, information theory, philosophy and methodology in three recently published books. We analyze the content of these books demonstrating how they help to delineate the emergent filed of the Study of Information and to form the foundations of this field.

## 2. Philosophy and Methodology of Information. The Study of Information in the Transdisciplinary Perspective

Books *Philosophy and Methodology of Information* [2] and *Theoretical Information Studies* [3] constitute a duology published to lay out the foundation for the research field of the study of information. Both books are based on the selected materials from the summit of the International Society for the study of information held in Gothenburg in 2017 (http://is4si-2017.org). They represent the basic domains of the study of information—philosophy of information, methodology of information and information theory. The first book contains a collection of the best philosophical and methodological contributions from the Gothenburg summit, with the number of invited contributions of leading researchers in the study of information. The book presents works aimed at establishing philosophical and methodological aspects of a new interdisciplinary/transdisciplinary field of study, with new ways of knowing and novel models of explanation, based on the transdisciplinary approach and data-information-knowledge ecology. It provides perspectives on information as a phenomenon that connects together, and enables new non-reductionist unifications of phenomena usually studied

in isolation within particular sciences, humanities, technologies and arts, and other information fields. Specialized academic fields currently divide human knowledge and experience into isolated silos. In contrast to this, the study of information creates a holistic approach to the reality defined by the omnipresence and pervasiveness of information in all areas and domains. Founded on the research results of the domain-knowledge, the Study of Information seeks a new, networked, ecological approach to enable an inclusive view, from which future research projects transcending disciplinary boundaries will derive, building on the understanding of information structures and processes underlying particular domains.

This volume emphasizes fundamental dependence of the study of information in general, and the philosophy of information, in particular, on the methodology and logic of information as essential parts of the study of information. It includes works offering new logical perspectives and exploring transconsistent logic for model-based reasoning, arguments for the necessity of logical pluralism, while addressing the logic of transdisciplinarity. In the book, connections of the philosophy of information with sciences and technologies, as well as their societal and ethical aspects within the study of information are explicated. In addition, the authors also make connections with its methodological and logical foundations that are in the process of continuous evolution.

The book consists of four parts: (1) Philosophy of Information, (2) Methodology of Information, (3) Philosophy of the Study of Information, and (4) Methodology of Information Studies.

## 3. Theoretical Information Studies. Information in the World

This second volume [3] in the duology presents the research on establishing theoretical facets of the new interdisciplinary/transdisciplinary field of the study of information, with innovative ways of knowing and advanced models of explanation, established on the forward-thinking general theory of information, data-information-knowledge ecology, reality stratification via levels of abstraction/organization, and information plus computation taxonomies. It provides theoretical perspectives on information and its dynamics including computation and providing a new unification mechanism for phenomena within fields founded on information and computation. Based on the research results of the domain-knowledge, the study of information and computation seeks a new, networked approach to enable an inclusive view, from which future research projects transcending disciplinary boundaries will derive, starting from the understanding of information structures and its processes underlying particular domains. The study of information and computation involves a recursive process of analysis and synthesis through continuous learning—about the natural as well as synthetic/created/cultural worlds, and about their actors—living beings and artificial agents. Theoretical information studies form an essential intrinsic component of the study of information.

Covering the most important domains of theoretical information studies, the book consists of four parts: (1) Foundations of Information, (2) Information as a Natural Phenomenon, (3) Cognition and Intelligence in Natural and Artificial Systems, and (4) Social, Cultural, Ethical and Legal Aspects of Information.

## 4. Information and Computation. Essays on Scientific and Philosophical Understanding of Foundations of Information and Computation

Chronologically, this book [4] preceded the previous two. Thematically, it provides deep-diving into the tandem of fundamental complementary concepts of information and computation, where computation is defined as information dynamics or information process. Information is a basic structure of the world, while computation is a process of the dynamic change of information.

On the one hand, transparent relations between information and computation are explicated in algorithmic information theory [5]. On the other hand, computation is a kind of information processing representing important issues of information dynamics.

Despite increasing importance and omnipresence of information, especially proliferated through Information and Communication Technology (ICT), scientists have not yet come to the common view on the phenomena of information and computation. There are hundreds of approaches

and dozens of theories but still the concepts of information and computation are evasive and vague, although adequate knowledge about their nature is necessary for our understanding of the world we live in. Understanding the place and role of information in the world is an important philosophical and methodological problem. Some authors relate information only to human communication in society. Others also include information structures and processes on the level of individual human beings. In contrast to this, present authors as well as a number of other researchers working on the naturalization of information, argue that information is everywhere in nature. As a result, researchers study information and its processes (computation) at different levels of the structure of the world. Our approach implies considering the structure of information processes as basic natural, social and technological phenomena. It involves an exploration of relations between the concept of information and basic concepts used to reason about the world, such as matter/energy, mind and knowledge. In this framework, information is related to everything and everything is related to information. For a cognitive agent, epistemologically, energy comes as a kind of information, because for an agent, all physical characteristics are information about the physical world. Theoretically, this is explained in the general theory of information [1]. This understanding is also supported by physicists. For instance, according to Wheeler, exploration of nature demonstrates that every physical quantity derives its meaning from information [6]. Von Weizsäcker developed the fundamental approach to physics based on information, the Ur-theory, with the basic idea that physicists learn about nature through observations, experiments and measurements that enable information extraction from physical objects and processes, as they are entirely characterized by the information that can be acquired from them [7].

*Information and Computation* provides a cutting-edge view of the world-class authorities in fields where information and computation play a central role. It sketches the contours of the future landscape for the development of our understanding of information and computation, their mutual relationship and the role in cognition, informatics, biology, artificial intelligence, and information technology.

## 5. Conclusions

In this article, we explored information presentation in three books dedicated to the study of information: duology *Philosophy and Methodology of Information* [2] and *Theoretical Information Studies*: *Information in the World* [3] together with the book *Information and Computation* [4]. They are addressing the study of contemporary dominant and conceptually fundamental dual phenomena of information and computation in a new theoretical, philosophical and methodological framework of knowledge ecology and the general theory of information. These books reflect the work of world's leading researchers within the field, indicating important unsolved problems and opening new directions for the future development.

**Funding:** The research of the first author is funded by Swedish Research Council grant.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Burgin, M. *Theory of Information*: *Fundamentality*, *Diversity and Unification*; World Scientific: New York, NY, USA; London, UK; Singapore, 2010
2. Dodig-Crnkovic, G.; Burgin, M. *Philosophy and Methodology of Information. The Study of Information in the Transdisciplinary Perspective*; World Scientific: Singapore, 2019. Available online: https://worldscientific.com/worldscibooks/10.1142/11190 (accessed on 6 May 2020).
3. Burgin, M.; Dodig-Crnkovic, G. *Theoretical Information Studies. Information in the World*; World Scientific: Singapore, 2020. Available online: https://worldscientific.com/worldscibooks/10.1142/11191 (accessed on 6 May 2020).
4. Dodig-Crnkovic, G.; Burgin, M. *Information and Computation*; World Scientific: Singapore, 2011. Available online: https://worldscientific.com/worldscibooks/10.1142/7637 (accessed on 6 May 2020).
5. Chaitin, G.J. Algorithmic information theory. *IBM J. Res. Dev.* **1977**, *21*, 350–359

6. Wheeler, J.A. Information, Physics, Quantum: The Search for Links. In *Complexity, Entropy, and the Physics of Information*; Zurek, W., Ed.; Addison-Wesley: Redwood City, CA, USA, 1990; pp. 3–28
7. von Weizsäcker, C.F. *Aufbau der Physik*; Hanser, Munich, Germany, 1985; English translation: *The Structure of Physics*; Springer: Berlin/Heidelberg, Germany; New York, NY, USA, 2006.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

proceedings

Conference Report

# 2021 Summit of the International Society for the Study of Information (IS4SI) †

**Marcin J. Schroeder [1,*] and Mark Burgin [2]**

[1] Global Learning Center, Institute for Excellence in Higher Education, Tohoku University, Sendai 980-8576, Japan
[2] Department of Computer Science, University of California, 520 Portola Plaza, Los Angeles, CA 90095, USA; markburg@cs.ucla.edu
* Correspondence: mjs@gl.aiu.ac.jp
† Summary of 2021 Summit of the International Society for the Study of Information, online, 12–19 September 2021.

**Abstract:** Continuing the tradition of S4SI Summits, the 2021 Summit had the form of ten federated conferences focusing on more specific aspects of the study of information. Its title, "Information Study for the Benefit of Humanity: Learning from the Past and Building the New Normal," referred to the present great challenge faced by the global community—the pandemic of a new coronavirus that itself can be considered a large scale informational phenomenon. However, the pandemic was not the only global challenge discussed at the conferences contributing to the summit. Another great global challenge discussed at the summit was related to the issues arising with the rapidly increasing role of information technology in all domains of human life and work, and the dangers of their abuse or uncontrolled drift in directions that threaten human wellbeing. These concerns pointed at the need for a better understanding of information as a central concept of science, technology, and culture, another main theme of the summit.

**Keywords:** information; computing; research forum; artificial and natural intelligence; theory; foundations; digital humanism

**Citation:** Schroeder, M.J.; Burgin, M. 2021 Summit of the International Society for the Study of Information (IS4SI). *Proceedings* **2022**, *81*, 1. https://doi.org/10.3390/proceedings2022081001

Academic Editor: Carlos Filipe Da Silva Portela

Published: 17 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

The 2021 Summit of the International Society for the Study of Information (IS4SI) was the fourth biannual congregation of the members of the society together with other researchers, philosophers, and practitioners active in all domains of the study of information.

Continuing the tradition established at the earlier S4SI Summits in Vienna (2015), Gothenburg (2017), and Berkeley (2019), the 2021 Summit had the form of ten federated conferences focusing on more specific aspects of the study of information. Its title, "Information Study for the Benefit of Humanity: Learning from the Past and Building the New Normal," referred to the present great challenge faced by the global community—the pandemic of a new coronavirus that itself can be considered a large scale informational phenomenon involving biological, social, political manifestations of information and their dynamics. However, the pandemic was not the only global challenge discussed at the conferences contributing to the summit.

Another great global challenge discussed at the summit was related to the issues arising with the rapidly increasing role of information technology in all domains of human life and work. Although there are unquestionable, great benefits of Artificial Intelligence, digital means of communication, automation, and other technological developments in information processing, storage, and retrieval, there are equally great and formidable dangers of their intentional abuse or uncontrolled drift in directions that threaten human wellbeing. These concerns pointed at the need for a better understanding of information as a central concept of science, technology, and culture.

The search for an appropriate methodology for the study of information, providing us with the tools for such understanding, was the complementing theme of the summit. Not all contributions to the summit addressed directly the concept of information. The summit had, as one of its objectives, fostering mutual understanding between diverse fields of study, those in which information already has its central role and those where the presence of information is acknowledged but at present less recognized.

Due to the limited size, the summary of the summit is mainly about the plenary contributions consisting of plenary discussions and 76 plenary lectures, talks, and presentations suggested for all participants of the summit. The plenary sessions were scheduled without alternative tracks to allow for the attendance of all participants. The ten contributing conferences had their more extensive programs of presentations and discussions in parallel sessions with a more specialized focus, leaving the choice of attendance to the preferences of participants. The information about these parallel sessions can be found on the website of the Summit (IS4SI 2021) and/or on the individual websites of the contributing conferences.

## 2. President's Theme Statement for the 2021 IS4SI Summit (Marcin J. Schroeder)

The 2021 IS4SI Summit happened in the shadow of the global crisis of the new coronavirus pandemic. Dozens of millions of people have already been infected, some of them suffered but managed to recover but more than a million perished. Millions lost their sources of income, jobs, businesses. The word "crisis" comes from the Greek meaning discernment, judgment, which describes well our present experience. There are now many examples of authentic heroism manifested by people who risk their health or lives bringing medical care to those infected with the virus and there are many examples of the unscrupulous exploitation of human suffering for political or economic gains.

Recently, there is hope that new vaccines will soon bring the spread of the virus under control and that life will return to normal. In this desire to return to normal and to put the dramatic experience behind us there is a hidden danger of forgetting about what we can learn from it. Hegel, in the Introduction to his Philosophy of History, pessimistically claimed "What experience and history teach is this—that people and governments never learned anything from history, or acted on principles deduced from it." Now, almost two centuries later, this assessment of the short span of memory among politicians and in the general public is equally accurate, but this should not be an excuse for the academic community to forget and not to learn from our experience. Going back to the "old normal" would be a defeat of reason and betrayal of those who suffered and those who sacrificed themselves in the fight with the disease.

It is important not only not to forget and to learn, but what to learn. If the lesson we were given was only about the lack of responsibility of political and economic leaders, their lack of imagination, about the deficiency of their morals, or about the frequently irrational reactions of the general public, then the pain brought by the pandemic would have been in vain. We have multiple examples of those in the past centuries or even millennia. Only someone naïve can expect that next time will be different. Searching for someone to blame for errors cannot replace searching for sources of errors. We can see that a lot of damage and suffering was a result of ignorance fueled by antiscientific and anti-intellectual attitudes, frequent both among political and economic leaders and within broad masses of the human population. It would be counterproductive to be satisfied with this diagnosis, in particular in the latter case. Are we, members of the academic community, without blame? It is our responsibility to promote knowledge and understanding of the world through education. Can we be satisfied with what we offer our societies? Do we have a comprehensive, consistent, scientific, or more generally intellectual view of reality that we could all understand and share among ourselves, and which we can present to the global society?

This does not mean that we have to seek consensus regarding the answers to specific research questions. Progress in science and philosophy has always depended on competing ideas, but productive competition requires common foundations, common rules, and mu-

tual understanding. Therefore, we can disagree about answers, but we have to understand each other's questions. We should be able to present this common foundation to the general public, which does not have any reason to trust in our advice when confronted with the lack of agreement and mutual understanding among scientists and intellectuals, in stark contrast to the extreme self-confidence of those who unite in the fight for political and economic power.

There is a desperate need for building a common foundation for science, philosophy, and humanities, for the construction of a comprehensive view of reality integrating the results of diverse methods of inquiry, to restore the position of knowledge and reason as guides for the global society. IS4SI has a special role to play in this task. The concept of information, as probably no other concept in modern thought, is not only present and fundamental in virtually all academic and technological disciplines, but it is familiar (admittedly without a very deep understanding) to the majority of the human population. Its universality qualifies it as the best tool to build an authentic transdisciplinary vision of reality and, at the same time, its familiarity with the general public makes it a good tool for the propagation of this vision.

The 2021 Summit consisted of several component conferences grouping experts in specific forms of inquiry of information, at the same time including plenary activities bringing together all participants. These conferences provided an opportunity to present to fellow researchers the most recent developments within their foci of expertise and the plenary events of the summit had the role of a forum for the exchange of views coming from multiple perspectives for seeking mutual understanding and cooperation.

## 3. Organizing Committee of the Summit

The Organizing Committee was chaired by Marcin J. SCHROEDER (IS4SI President; Professor at Global Learning Center of IEHE, Tohoku University at Sendai, Japan) and it consisted of the chairs of the organizing committees of contributing conferences and the members with special responsibilities (e.g., information technology support):

- Syed Mustafa ALI (School of Computing and Communications, The Open University, Milton Keynes, UK),
- Charalambos ALIFIERIS (Independent Scholar, Greece),
- Peter BOLTUC (University of Illinois, Springfield, MO, USA & Warsaw School of Economics, Warsaw, Poland),
- Joseph BRENNER (Independent Scholar, Switzerland),
- Mark BURGIN (University of California, Los Angeles, CA, USA),
- Zhicheng CHEN (IS4SI Vice-President for Special Inteest Groups, China),
- Gordana DODIG-CRNKOVIC (Chalmers University of Technology and the University of Gothenburg, Gothenburg, Sweden),
- Raffaela GIOVAGNOLI (Faculty of Philosophy, Pontifical Lateran University, Vatican),
- Annette GRATHOFF (IS4SI Secretary General, Vienna, Austria),
- Masami HAGIYA (Department of Information Science, Graduate School of Information Science and Technology University of Tokyo, Tokyo, Japan),
- Wolfgang HOFKIRCHNER (The Institute for a Global Sustainable Information Society, Vienna, Austria),
- David J. KELLEY (AGI Laboratory, Seattle, WA, USA),
- Hans-Jörg KREOWSKI (University of Bremen, Bremen, Germany),
- Robert LOWE (Department of Applied IT, University of Gothenburg, Gothenburg, Sweden),
- Pedro MARIJUÁN (Independent Scholar, Zaragoza, Spain),
- Dénes NAGY (President, International Society for the Interdisciplinary Study of Symmetry, Budapest, Hungary),
- Yasuhiro SUZUKI (Graduate School of Informatics, Nagoya University at Nagoya, Nagoya, Japan)
- Vera VIANA (Faculty of Architecture, The University of Porto, Porto, Portugal),

- Kun WU (Department of Philosophy, School of Humanities and Social Sciences, Xi'an Jiaotong University, Xi'an, China),
- Yixin ZHONG (Beijing University of Posts and Telecommunications, Beijing, China).

Contributing conferences had their own organizing committees, with membership listed on the website of the Summit (IS4SI2021).

The summit did not have a program committee separate from program committees for contributing conferences, which, in some cases, were identical with organizing committees. Chair of the Summit Organizing Committee coordinated the work of program committees for contributing conferences regarding the choice and design of contributions to the plenary program. The schedule of plenary events was designed by the chair of the summit in cooperation with chairs and co-chairs of contributing conferences. This was a complex task for plenary events, which were intended for all participants of the summit as they were in almost all possible time zones of the world. There were last minute changes in the schedule due to unpredictable circumstances (e.g., health emergencies) that required fast decisions. For this reason, the flexible one person decision making on the schedule was preferable.

## 4. Contributing Conferences

The titles of contributing conferences (together with abbreviations used in the materials of the summit) were as follows:

- **Theoretical and Foundational Problems in Information Studies (TFP)**
- **Information in Biologically Inspired Computing Architectures (BICA)**
- **Digital Humanism (Dighum)**
- **Symmetry, Structure, and Information (SIS)**
- **Morphological Computing of Cognition and Intelligence (MORCOM)**
- **Habits & Rituals (H&R)**
- **13th International Workshop on Natural Computing (IWNC)**
- **Philosophy and Computing (APC)**
- **5th International Conference on Philosophy of Information (ICPI)**
- **Global Forum for Artificial Intelligence (GFAI)**

Some of these conferences belonged to the tradition of the IS4SI Summits, some others were new contributions from organizations with interests related to information.

A more detailed presentation of the topics and programs of contributing conferences and extended abstracts of papers presented at their sessions can be found on the website of the summit (IS4SI 2021) and the websites created by their organizers:

Dighum: IS4SI 2021 Digital Humanism workshop programmed—GSIS (accessed on 16 March 2022),

TFP: Theoretical and Foundational Problems (TFP) in Information Studies (tfpis.com) (accessed on 16 March 2022),

IWNC: https://www.natural-computing.com/#iwnc-13/ (accessed on 16 March 2022),

SIS web page: Schedule for the Conference "Symmetry, Structure and Information" in the IS4SI 2021 Summit—The International Society for the Interdisciplinary Study of Symmetry Schedule for the Conference "Symmetry, Structure and Information" in the IS4SI 2021 Summit—The International Society for the Interdisciplinary Study of Symmetry (symmetry-us.com) (Schedule for the Conference "Symmetry, Structure and Information" in the IS4SI 2021 Summit—The International Society for the Interdisciplinary Study of Symmetry (symmetry-us.com) (accessed on 16 March 2022).

All participants were invited to submit their papers to the Special Issue of Proceedings (MDPI). The choice of the venue for reviewing and publishing materials presented at the summit belongs to the organizing committees of contributing conferences and the present volume of proceedings includes only papers selected in this process. In essence, the review process had two stages—at first, the materials submitted for the presentation at the conference were reviewed and selected for presentation and then presented papers were reviewed and selected for publication.

Several videos with recordings of the presentations from the plenary sessions have been already edited and published on YouTube. Editing and publishing of other recordings is an ongoing project of IS4SI (IS4SI 2021 accessed on 16 March 2022).

**5. Sessions of Plenary Program**

The following list shows the titles of presentations at plenary sessions gathering all the participants of the summit and names of authors in the chronological order of their presentations. The titles of presentations at nonplenary, parallel sessions of contributing conferences are reported in the editorials in *Proceedings* for contributing conferences presenting submitted and accepted papers. The readers will find, in Proceedings, a large collection of papers derived from the work presented at the summit.

Presentations and discussions of the plenary program of the summit, listed below, addressed a very wide range of topics related to information studies from the aspects addressed by natural sciences (in particular biology and physics), computer science (including technological, philosophical and practical matters), logic, and mathematics, to humanities, social sciences, and philosophy. The intention of the program design was to saturate plenary sessions with topics providing a broad view on information and its role in our understanding of reality and, on the other hand, in creating a better future for humanity.

1. *Opening Panel Discussion "What is the SI in IS4SI?"* moderated by Marcin J. Schroeder (Tohoku University, Sendai, Japan)
2. *Autopoietic machines: Going beyond the half-brained AI and Church-Turing Thesis* presented by Rao Mikkilineni (Ageno School of Business, Golden Gate University, San Francisco, CA, USA)
3. *Research in the area of Neosentience, Biomimetics, and Insight Engine 2.0* by Bill Seaman (Computational Media, Arts and Cultures; Emergence Lab, Duke University, Durham, NC, USA)
4. *Mind, Nature, and Artificial Magic* by Rossella Lupacchini (University of Bologna, Bologna, Italy)
5. *Non-Diophantine arithmetics as a tool for formalizing information about nature and technology* by Michele Caprio, Andrea Aveni, and Sayan Mukherjee (Duke University, Durham, NC, USA)
6. *Ontological information—information as a physical phenomenon* by Roman Krzanowski (The Pontifical University of John Paul II, Krakow, Poland)
7. *Materialization and Idealization of Information* by Mark Burgin (University of California, Los Angeles, CA, USA)
8. *Paradigm Shift, an Urgent Issue for the Studies of Information Discipline* by Yixin Zhong (Beijing University of Posts and Telecommunications, Beijing, China)
9. *Structural Analysis of Information: Search for Methodology*, by Marcin J. Schroeder (Global Learning Center, IEHE, Tohoku University, Sendai, Japan)
10. *Quality of information* by Krassimir Markov (ITHEA, Sofia, Bulgaria)
11. *A QFT Approach to Data Streaming in Natural and Artificial Neural Networks* by Gianfranco Basti (Faculty of Philosophy, Pontifical Lateran University, Vatican City) and Giuseppe Vitiello (Department of Physics "E. R. Caianiello", University of Salerno, Fisciano (Salerno), Italy)
12. *Arithmetic loophole in Bell's theorem: Overlooked threat to entangled-state quantum cryptography* by Marek Czachor (Institute of Physics and Computer Science, Gdańsk University of Technology, Gdańsk, Poland)
13. *Advanced NLP procedures as premises for the reconstruction of the idea of knowledge* by Rafal Maciag (Institute of Information Studies, Jagiellonian University, Krakow, Poland)
14. *Toward a Unified Model of Cognitive Functions* by Pei Weng (Temple University, Philadelphia, PA, USA)
15. *A Nested Hierarchy of Analyses: From Understanding Computing as a Great Scientific Domain, through Mapping AI and Cognitive Modeling and Architectures, to Developing a*

*Common Model of Cognition* by Paul Rosenbloom (USC Institute for Creative Technologies, University of Southern California, Playa Vista, CA, USA)

16. *The Development and Role of Symmetry in Ancient Scripts* by Peter Z. Revesz (Department of Computer Science and Engineering, University of Nebraska-Lincoln, Lincoln, NE, USA)
17. *Symmetry and Information: An odd couple (?)* by Dénes Nagy (President, International Society for the Interdisciplinary Study of Symmetry, Budapest, Hungary)
18. *Antinomies of Symmetry and Information* by Marcin J. Schroeder (Tohoku University, Sendai, Japan)
19. *SIS Conference Panel Discussion* moderated by Dénes Nagy & Marcin J. Schroeder
20. *Digital Humanism* by Julian Nida-Rümelin (Munich University, Munich, Germany)
21. *Humanism Revisited* by Rainer E. Zimmermann(Institute for Design Science Munich Germany/Clare Hall, Cambridge University, Cambridge, UK)
22. *The Indeterminacy of Computation: Slutz, Shagrir, and the mind* by B. Jack Copeland (University of Canterbury, Christchurch, New Zealand)
23. *Falling Up: The Paradox of Biological Complexity* by Terrence W. Deacon (University of California Berkeley, Berkeley, CA, USA)
24. *Almost disjoint union of Boolean algebras appeared in Punch Line* by Yukio Pegio Gunji (Department of Intermedia Art and Science, School of Fundamental Science and Technology, Waseda University, Tokyo, Japan)
25. *Why do not hatching alligator eggs ever produce chicks?* by Aaron Sloman (School of Computer Science, University of Birmingham, Birmingham, UK)
26. *Morphogenesis as a model for computation and basal cognition* by Michael Levin (Tufts Center for Regenerative and Developmental Biology, Tufts University, Medford, MA, USA)
27. *Cross-Embodied Cognitive Morphologies: Decentralizing Cognitive Computation Across Variable-Exchangeable, Distributed, or Updated Morphologies* by Jordi Vallverdú (Universitat Autònoma de Barcelona, Catalonia, Spain)
28. *Designing Physical Reservoir Computers* by Susan Stepney (University of York, York, UK)
29. *The Aims of AI: Artificial and Intelligent* by Vincent C. Müller (TU/e & University of Leeds, Turing Institute, Leeds, UK)
30. *Cognition through Organic Computerized Bodies. The Eco-Cognitive Perspective* by Lorenzo Magnani (University of Pavia, Pavia, Italy)
31. *Digital Consciousness and the Business of Sensing, Modeling, Analyzing, Predicting, and Taking Action* by Rao Mikkilineni (Ageno School of Business, Golden Gate University, San Francisco, CA, USA)
32. *On Leveraging Topological Features of Memristor Networks for Maximum Computing Capacity* by Ignacio Del Amo and Zoran Konkoli (Chalmers University of Technology, Gothenburg, Sweden)
33. *Habits and Rituals as Stabilized Affordances and Pregnances: A Semiophysical Perspective* by Lorenzo Magnani (University of Pavia, Pavia, Italy)
34. *A neurocomputational model of relative value processing: Habit modulation through differential outcome expectations* by Robert Lowe (Department of Applied IT, University of Gothenburg, Gothenburg, Sweden)
35. *Capability and habit* by Matthias Kramm (Wageningen University & Research, Wageningen, The Netherland)
36. *Collective Intentionality and the Transformation of Meaning During the Contemporary Rituals of Birth* by Anna M. Hennessey (Visiting Scholar, Berkeley Center for the Study of Religion, University of California, Berkeley, CA, USA)
37. *Habitual Behavior: from I-intentionality to We- intentionality* by Raffaela Giovagnoli (Faculty of Philosophy, Pontifical Lateran University, Vatican)
38. *Machines computing and learning?* by Genaro J. Mart'ınez (Artificial Life Robotics Laboratory, Escuela Superiorde C'omputo, InstitutoPolit´ecnico Nacional, M´exico & Unconventional Computing Lab, University of the West of England, Bristol, UK)

39. *Computing with slime mould, plants, liquid marbles, and fungi* by Andy Adamatzky (Unconventional Computing Lab, UWE, Bristol, UK)
40. *IWNC Panel Discussion* moderated by Marcin J. Schroeder
41. *Exploring open-ended intelligence using patternist philosophy* by Ben Goertzel (Singularty Net Foundation & OpenCog Foundation)
42. *The Artificial Sentience Behind Artificial Inventors* by Stephen Thaler (Imagination Engines Inc.)
43. *Potential Impacts of Various Inventorship Requirements* by Kate Gaudry (Kilpatrick Townsend & Stockton LLP)
44. *Panel Commentary* by Peter Boltuc (University of Illinois, Springfield, USA & Warsaw School of Economics, Warsaw, Poland)
45. *On Two Different Kinds of Computational Indeterminacy* by Oron Shagrir, Philippos Papayannopoulos, and Nir Fresco (Hebrew University of Jerusalem, Jerusalem, Israel)
46. *Cognitive neurorobotic self in the shared world* by Jun Tani (Cognitive Neurorobotics Research Unit, Okinawa Institute of Science and Technology (OIST), Okinawa, Japan)
47. *The Future of Anthroposociogenesis—Panhumanism, Anthroporelational Humanism and Digital Humanism* by Wolfgang Hofkirchner (The Institute for a Global Sustainable Information Society, Vienna, Austria)
48. *The Philosophy—Science Interaction in Innovative Studies* by Yixin Zhong (Beijing University of Posts and Telecommunications, Beijing, China)
49. *Information and the Ontic-Epistemic Cut* by Joseph Brenner (Independent Scholar, Switzerland)
50. *A Chase for God in the Human Exploration of Knowledge* by Kun Wu, Kaiyan Da, Tianqi Wu (Department of Philosophy, School of Humanities and Social Sciences, Xi'an Jiaotong University, Xi'an, China)
51. *The Second Quantum Revolution and its Philosophical Meaning* by Hongfang L. (School of Marxism University of Chinese Academy of Sciences)
52. *Information and Disinformation with their Boundaries and Interfaces* by Gordana Dodig-Crnkovic (Chalmers University of Technology and the University of Gothenburg, Gothenburg, Sweden)
53. *A Quantum Manifestation of Information* by Tian'en Wang (Shanghai University, Shanghai, China)
54. *Computation and Eco-Cognitive Openness-Locked Strategies, Unlocked Strategies, and the Dissipative Brain* by Lorenzo Magnani (University of Pavia, Pavia, Italy)
55. *In what sense should we talk about the perception of other minds?* by Duoyi Fei (China University of Political Science and Law, China)
56. *An a Priori Theory of Meaning* by Marcus Abundis (Bön Informatics, Aarau, Switzerland)
57. *Some Problems of Quantum Hermeneutics* by Guolin Wu (Institute for Advanced Study in Science, Technology and Philosophy, South China University of Science and Engineering, Guangzhou, China)
58. *The fast-changing paradigm of war calls for great wisdom of peace* by Lanbo Kang (Department of Political Science, Engineering University, Xi'an, China)
59. *Technologies, ICTs and Ambiguity* by Tomáš Sigmund (Prague University of Economics and Business, Czech Republic)
60. *The Data Turn of Scientific Cognition and the Research Program of Philosophy of Data* by Xinrong Huang (Research Center of Management Philosophy, Jiangxi University of Finance and Economics Nanchang, China)
61. *Testimony and Social Evidence in the Covid Era* by Raffaela Giovagnoli (Faculty of Philosophy, Pontifical Lateran University, Vatican)
62. *Developments of research on the Nature of Life from the Information Theory of Individuality* by Dongping Fan, Wangjun Zhang (South China Normal University, Institute for Science, Technology and Society, Center for Systems Science and Systems Management Research Guangzhou, Guangdong, China)
63. *On Information Interaction between the Hierarchy of the Material System* by Zhikang Wang (Sun Yat-sen University, China)

64. *Informational Aesthetics and the Digital Exploration of Renaissance Art* by John Douglas Holgate (St. George Hospital, Sydney, Australia)
65. *Practice, Challenges and Countermeasures of Accelerating the Development of new Generation of Artificial Intelligence in Xinjiang* by Hong Chen (Party School of Xinjiang Uygur Autonomous Region, Committee of the Communist Party of China, Xinjiang, Urumqi, China)
66. *A Basic Problem in the Philosophy of Information Science: Redundant Modal Possible World Semantics* by Xiaolong Wan (Research Center "Philosophy, Logic, and History of Science and Technology", University of Electronic Science and Technology of China, China)
67. *Paradigm Revolution Creates the General Theory of AI* by Yixin Zhong(Beijing University of Posts and Telecommunications, Beijing, China)
68. *Two Approaches to Artificial Intelligence* by Mark Burgin (University of California, Los Angeles, CA, USA)
69. *Intelligence Science Drives Innovation* by Zhongzhi Shi (Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China)
70. *On the Essential Difference Between the Intelligence Body and the Program Body* by HeHuacan (Northwestern Polytechnical University, China) and He Zhitao (Beijing University of Aeronautics and Astronautics, Beijing, China)
71. *Human body networks mechanisms of the Covid-19 symptoms* by Pin SUN, Rong LIU, Shui GUAN, Jun-Xiu GAO, and Chang-Kai SUN (Dalian University of Technology, Dalian, China)
72. *The Development and Characterization of A New Generic Wearable Single Channel Ear-EEG Recording Platform* by Rong Liu (School of Biomedical Engineering, Department of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian, China)
73. *Brain Imitating Method for Social Computing—Illumination of Brain Information Processing System* by Liqun Han (College of Artificial Intelligence, Beijing Technology and Business University, Beijing, China)
74. *Research and Prospects of Artificial Intelligence in Traditional Chinese Medicine* by Zixin Shu, Ting Jia, Haoyu Tian, Dengying Yan, Yuxia Yang, and Xuezhong Zhou (Institute of Medical Intelligence, School of Computer and Information Technology, Jiaotong University, Beijing, China)
75. *A Framework of "Quantitative $\oplus$ Fixed Image $\Rightarrow$ Qualitative" induced by contradiction generation and Meta Synthetic Wisdom Engineering* by Jiali Feng (Information Engineering College, Shanghai Maritime University, Shanghai, China)
76. *Paradox, Logic, and Property of Infinity* by Jincheng Zhang (101 College Entrance Examination Continuation School of Guangde County, China)
77. *A Call for Paradigm Shift in Information Discipline* by Zhong Yixin (Beijing University of Posts and Telecommunications, Beijing, China)

**Funding:** The work on the summary received no external funding.

**Acknowledgments:** The authors would like to express their gratitude to the participants of the summit for their presentations, to the members of the Organizing Committee of the Summit, and to the organizers of contributing conferences for their work on many aspects of the implementation of the Summit. Special thanks to David Kelly and his colleagues in AGI Laboratory for the technical support for the summit and for editing and publishing several video recordings from the plenary sessions.

**Conflicts of Interest:** The authors declare no conflict of interest.

proceedings

*Editorial*

# Conference Theoretical and Foundational Problems in Information Studies †

**Mark Burgin**

Department of Computer Science, University of California, Los Angeles, CA 90095, USA; markburg@cs.ucla.edu

† Presented at the Conference on Theoretical and Foundational Problems in Information Studies, IS4SI Summit 2021, online, 12–19 September 2021.

**Abstract:** This paper has a two-fold goal. In the first part, the area of theoretical and foundational information studies is delineated. In the second part, the general overview of the conference "Theoretical and Foundational Problems in Information Studies" (TFP) is given.

**Keywords:** information; general theory; special theory; quantum information; artificial intelligence; knowledge; meaning; information processing; mathematics; physics; computer science; ontology

**Citation:** Burgin, M. Conference Theoretical and Foundational Problems in Information Studies. *Proceedings* **2022**, *81*, 32. https://doi.org/10.3390/proceedings2022081032

Published: 11 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Theoretical Information Studies and Foundations of Information

Information theory constitutes the basic component of theoretical information studies; however, there are also other components, including the applications of information studies to the theoretical areas of science and humanities, e.g., in relation to theoretical physics or theoretical computer science. Another component consists of information-oriented and information-based theories in different areas, e.g., genetics, which treats hereditary information or theoretical computer science, and studies information processing.

Therefore, theoretical information studies include the following areas.

- Information theory;
- Theoretical computer science as it studies information processing by technical devices;
- Theoretical linguistics as it studies information processing by natural and artificial languages;
- Semiotics as it studies symbolic information processing;
- Psychology as it studies information processing by people;
- Pedagogy as it studies information transmission and knowledge acquisition;
- Application of information theory to theoretical physics (information and the Universe);
- Application of information theory to theoretical computer science (information, computation and networking);
- Application of information theory to the theory of complexity (information and complexity);
- Application of information theory to mathematics (information and structure);
- Application of information theory to theoretical linguistics (information and language);
- Applications of information theory in ecology (information and environment);
- Application of information theory to decision making (information and decision-making);
- Applications of information theory in economics and finance (information and economy);
- Application of information theory to pedagogy (information and education);
- Applications of information theory in sociology (information and society);
- Applications of information theory in psychology (information and creativity);
- Applications of information theory in biology and medicine (information and the organism);
- Applications of information theory in anthropology (information and human beings);

- Application of information theory to semiotics (information and symbols);
- Application of information theory to epistemology (information and cognition);
- Application of information theory to esthetics (information and beauty);
- Application of information theory to ethics (information and moral);
- Application of information theory to law (information and law);
- Application of information theory in humanities;
- Information-oriented and information-based physical theories;
- Information-oriented and information-based psychological theories;
- Information-oriented and information-based biological theories;
- Information-oriented and information-based theories of decision making;
- Information-oriented and information-based economic theories;
- Information-oriented and information-based sociological theories;
- Information-oriented and information-based ecological theories;
- Information-oriented and information-based epistemology;
- Information-oriented and information-based esthetics (information and beauty);
- Information-oriented and information-based ethics (information and moral);
- Information-oriented and information-based anthropological theories.

Theoretical information studies also have diverse applications to a variety of practical problems.

The foundations of information and information studies include the following areas:

- Mathematical foundations of information and information studies;
- Logical foundations of information and information studies;
- Philosophical foundations of information and information studies;
- Methodological foundations of information and information studies;
- Linguistic foundations of information and information studies;
- Physical foundations of information and information studies;
- Biological foundations of information and information studies;
- Psychological foundations of information and information studies;
- Sociological foundations of information and information studies;
- Technological foundations of information and information studies;
- Ecological foundations of information and information studies;
- Anthropological foundations of information and information studies.

This shows that the goal of the foundations of information and information studies is to provide sound reliable and flexible foundations for information studies and their applications in general, as well as for theoretical information studies in particular.

## 2. The Conference

The conference, titled "Theoretical and Foundational Problems in Information Studies" (TFP), had participants from six continents, i.e., from all inhabited continents. They included top information science experts who had contributed the most to this exciting area as well as theoreticians from other disciplines who had used information theory in their studies. Coming from 33 countries, the participants delivered many interesting talks with new innovative results and prolific insights.

The conference had seven special sessions:

Special Session ***New Directions in Information Processing***
Session organizer: **Rao Mikkilineni**
Special Session ***Neosentience, Biomimetics, and the Insight Engine 2.0***
Session organizer: **Bill Seaman**
Special Session ***Information in Natural Sciences***
Session organizer: **Annette Grathoff**
Special Session ***Information in Social Sciences***
Session organizer: **Peter Carr**
Special Session ***Information, Knowledge, and Meaning***

Session organizer: **Rafal Maciag**
Special Session ***Information Phenomenon***
Session organizer: **Krassimir Markov**
Special Session ***Information in Practical Problems***
Session organizer: **Arkadiy Dantsker**

Here, only a very brief overview of these special sessions is given, while more detailed expositions can be found in the editorials of the session organizers.

The Special Session *New Directions in Information Processing* united presentations on the role of information theory in the development of information technology. Researchers suggested and considered various innovative machines, methods, and ideas in the area of information processing, such as the design of autopoietic machines, quantum computing, data streaming in natural and artificial neural networks, the principles of knowledge and database creation, the temporal theory of the brain, processing information by molecules, the evolution of the extended mind, consciousness, machines, and ethics.

The main goal of the Special Session *Neosentience, Biomimetics, and the Insight Engine 2.0* was to present diverse ideas and research results related to an intelligent autonomous learning robotic system via transdisciplinary information processes and information exchanges, enabling neosentience to arise via the system's functionality in the form of The Insight Engine 2.0. The researchers discussed the problems of 3D visualization, the embodied, embedded, enactive and extended approaches to human cognition, $n$-dimensional combinatoric bio-algorithm development, mindful awareness, second-order cybernetics, neuroscience and computational neuroscience, neuroscience and the arts, computational creativity, artificial general intelligence, AI and ethics; robotics and situated knowledge production, the history of AI, computational intuition, autonomous adaptive intelligence, and attention schema theory.

The presenters in the Special Session *Information in Natural Sciences* discussed the role of information in natural sciences and the essence of information in nature considering quantum information, negative probability, impact of arithmetic on physics and cryptography, topological quantum theory, biomathics, and nonlinear phenomena in physics and biology.

The presenters in the Special Session *Information in Social Sciences* talked about the role and essence of information in society considering synergy in scientific collaboration, the problem of fake news, pricing theory as an information tool in finance, information aspects of aboriginal ontology, self-learning as an important cognitive process in social environment, the role of information in fighting the COVID pandemic, processing legal information, and problems of human values related to information functioning in contemporary society.

The Special Session *Information, Knowledge, and Meaning* dealt with problems of the theory of meaning, conceptual spaces, NLP procedures, vagueness, and errors. In addition, it included an interesting group of presentations that discussed how numbers and arithmetic convey information about the multitude of things reflecting on the new advancements in the domain of numbers and arithmetic. In particular, problems of non-Diophantine arithmetics and their applications in science and technology were discussed.

The Special Session *Information Phenomenon* was dedicated to the foundational problems in information studies. The presentations in this session treated problems in such areas as information definition, information quality, natural typology of information, the paradigm of the information discipline, hilomorphic theory, structural analysis, probability in the context of information, as well as the phenomenology and ontology of information.

Finally, the Special Session *Information in Practical Problems* was dedicated to the applications of information theory and innovative mathematical theories to imperative practical problems, such as pollution of water resources, monitoring and safety management of thermal processes, the recovery of blurred images, and ways to provide more renewable energy.

## 3. Conclusions

The conference "Theoretical and Foundational Problems in Information Studies" (TFP) brought together experts from different disciplines, allowing them to present their pioneering results, hold fruitful discussions on various topics, and formulate innovative directions for future research.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The author would like to thank the members of the Organizing and Program committees of the FTP conference, and contributors to the FTP conference at IS4SI summit 2021 on excellent collaboration and MDPI for publication of the conference materials.

**Conflicts of Interest:** The author declares no conflict of interest.

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
Tel. +41 61 683 77 34
Fax +41 61 302 89 18
www.mdpi.com

MDPI Books Editorial Office
E-mail: books@mdpi.com
www.mdpi.com/books

We dedicate this book to the memory of our colleague, Professor Mark Burgin, who recently passed away. The book comprises a reprint of Professor Burgin's collected articles originally published for the publisher MDPI. Mark Burgin was a distinguished mathematician, computer scientist, philosopher, and information theoretician who made significant contributions to various fields, including information and computation, foundations of mathematics, philosophy of information and computation, theory of knowledge, and logic. His innovative ideas and research had a profound impact on many areas of study, providing novel insights and advancing our understanding of complex information-based phenomena. His interdisciplinary approach, which bridged the gaps between different fields, led to new connections, approaches, and applications that significantly influenced the development of these fields. With this book, we aim to honor the memory of Mark Burgin and celebrate his remarkable career. We hope that this book will serve as a testament to his contributions and as a source of inspiration for future research in the fields to which he has contributed so significantly.

www.mdpi.com

MDPI