

Communication

Moving towards Smart Cities: A Selection of Middleware for Fog-to-Cloud Services [†]

Hind Bangui ^{1,2,3,*} , Said Rakrak ³, Said Raghay ³ and Barbora Buhnova ^{1,2}

¹ Faculty of Informatics, Masaryk University, 602 00 Brno, Czech Republic; buhnova@fi.muni.cz

² Institute of Computer Science, Masaryk University, 602 00 Brno, Czech Republic

³ Applied Mathematics and Computer Science Laboratory at Cadi Ayyad University, 40000 Marrakech, Morocco; s.rakrak@uca.ma (S.R.); s.raghay@uca.ma (S.R.)

* Correspondence: hind.bangui@mail.muni.cz

[†] The work was supported from ERDF/ESF “CyberSecurity, CyberCrime and Critical Information Infrastructures Center of Excellence” (No. CZ.02.1.01/0.0/0.0/16_019/0000822).

Received: 1 October 2018; Accepted: 8 November 2018; Published: 11 November 2018



Abstract: Smart cities aim at integrating various IoT (Internet of Things) technologies by providing many opportunities for the development, governance, and management of user services. One of the ways to support this idea is to use cloud and edge computing techniques to reduce costs, manage resource consumption, enhance performance, and connect the IoT devices more effectively. However, the selection of services remains a significant research question since there are currently different strategies towards cloud computing, including services for central remote computing (traditional cloud model) as well as distributed local computing (edge computing). In this paper, we offer an integrated view of these two directions and the selection among the edge technologies based on MCDA (Multiple Criteria Decision Analysis) algorithms. To this end, we propose a foglet as a middleware that aims at achieving satisfactory levels of customer services by using fuzzy similarity and TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) to facilitate the rating and selection of services in the fog-to-cloud environment. Then, we describe the selection process with a numerical example, and conclude our work with an outline of future perspectives.

Keywords: fog computing; cloud computing; edge computing; smart services; smart cities; IoT; fuzzy similarity; TOPSIS; MCDA approaches

1. Introduction

Internet of Things (IoT) paradigm is a noticeable emerging concept that has succeeded in becoming an integral part of our daily life by enabling any object around us to produce, connect, and transfer data via network technologies (i.e., 4G). The smart city is one of the concepts that use the ability of IoT technologies to build an immediate bridge between intelligent services and citizens [1]. Consequently, the smart city paradigm is currently gaining enormous popularity by giving birth to new efficient development of societies and industries. To support this approach further and improve the urban citizens quality of living, many critical infrastructures related to IoT data services are being considered to enhance and realize the vision of smart cities, such as smart healthcare [2], smart grid [3], smart transportation [4], management of energy [5], surveillance systems [6], or smart building [7]. Thus, the IoT technologies have supported various domains with specific needs to perform daily tasks, enhancing the quality of living, as well as the governance of smart cities.

Meanwhile, most available IoT services have become data-driven, which allows them to achieve better recommendations and predictions of future trends of events in smart cities, such as the recommendation of the best available car parking lot. For that reason, the IoT technologies need the

incorporation of several techniques to meet the requirements of data, which include seamless real-time access, sharing, storing, processing, and analyzing data anywhere at any time. Cloud computing is one of the technologies that has been proposed to enrich the features of IoT applications by providing unlimited services represented by X as a service [8,9] (i.e., storage as a service) and designing suitable programming models for smart city IoT applications. Consequently, it is possible for IoT data to pave the way for sustainable development in smart cities and their related technologies.

Despite the ability of cloud platforms to provide sophisticated services, many IoT applications prefer to process data and make decisions locally rather than managing and treating IoT data remotely. Further, the access network to cloud platforms requires efficient and stable use of the bandwidth, which is not applicable to IoT-cloud connectivity which might be unreliable or poor in performance (throughput, latency). Consequently, it is beneficial to use the edge services (i.e., fog services) that are closer to IoT devices to improve the availability and scalability of data at the edge of computing [10]. Due to the advantages of edge computing (i.e., mobility support), IoT devices could access the proximate edge resources directly, adapt their request according to location awareness, process their computing tasks, and get results with minimum latency. However, cloud computing with its unlimited resources remains the best central platform to handle IoT tasks while edge computing is targeted to offer distributed local processing with minimized and predictable latency. That pushes us to ask what helps us to determine the best criteria that are driving the decision between a closer or a remote platform for processing IoT data.

In this paper, we address this challenge with the help of multi-criteria decision analysis (MCDA) [11,12] applied to the context of the edge/cloud paradigm. To this end, we first provide a summary of the most common technologies related to edge computing. Notably, we focus on the coordinated fog-to-cloud platforms since this kind of collaborative system integrates the required preferences for making the IoT devices self-contained and autonomous in real time. Then, we review the utilization of the MCDA methods in edge computing, since the MCDA approaches are very well known as a collection of techniques that offer comparison and comprehension of complex problems, enriching them with different forms of information. Next, we propose a middleware architecture to choose an adequate service that is recommended for processing IoT data. Following that, we describe step by step the application of the fuzzy similarity and TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) methods to deal with the selection process. Finally, we conclude our work by identifying future perspectives.

The remainder of this article is organized as follows. Section 2 gives a brief overview of the current technologies and projects in Edge of computing. Next, it describes the application of MCDA in Edge environment. In Section 3, we describe the proposed architecture and its features. Then, in Section 4, we move to explain the steps followed to select one service among others by applying fuzzy similarity based on the TOPSIS method with a numerical example. In Section 5, we discuss the experience of comparing the ranking results of different popular fuzzy similarity techniques. Finally, we conclude the article with future perspectives in Section 6.

2. Related Work

The objective of this section is to present an overview of the edge technologies and utilization of MCDA at the edge of computing.

2.1. Edge Technologies

Advances in edge computing have recently facilitated the growth of IoT applications as well as different scenarios in research and business sectors, including smart cities, smart home, environment monitoring, and others. Thus, in this subsection, we describe the benefits of deploying edge technologies as extenders and collaborators for cloud platforms.

2.1.1. Edge as an Extender of Cloud Services

Currently, there are specific platforms that support the rapid growth of the IoT paradigm and provide a rich menu of requirements for developing edge applications, such as OpenFog [13], Google's Go language [14], among others. Consequently, there are many real implementations of edge technologies, such as the development of wearable cognitive assistance spanning on Google Glass at the edge [15], where the computation-intensive tasks of the wearers are offloaded to nearby cloudlets to achieve tight end-to-end latency constraint. Nippon Telegraph and Telephone Corporation (NTT) [16,17] is one of the examples of edge projects that aims at designing, implementing and validating new management edge and cloud computing systems. One of the vital applications at edge computing is the use of the vast number of vehicles in cities as a fog platform to deliver various services in traffic support and analytics [18–20]. The realization of this idea of vehicular fogs surrounding IoT applications involves multiple technologies, such as vehicular technology, artificial intelligence, big data analytics, or wireless communication technology. For the orchestration of IoT devices, the MQTT (Message Queuing Telemetry Transport) broker [21] has been pre-installed at the Fog computing node based on SDN (Software-Defined Networking) for improved delivery performance of IoT applications. Likewise, in [14], the fog nodes have been exploited to manage the various smart objects directly in complex IoT scenarios. In [22], fogging has been used to automate website performance optimization at edge servers. This novel idea illustrates the importance of edge location by providing dynamic, customizable optimization based on local network and client device conditions. Further, catching technique was implemented within the fog servers to speed up the content delivery process to end-users. Similarly, the solution proposed in [23] considered fog servers as the main functional provider of computing and storage services. Another solution for reducing the waste of IoT resources was discussed in [24], where a Fog model based on previous QoE (Quality of Experience) records was proposed to solve this issue by managing the IoT resources and improving their QoS (Quality of Service) at fog. Equally, in [25], an architecture based on OpenStack platform, called WiCloud, has been developed as a mobile edge computing platform for enhancing location-awareness, managing inter-mobile-edge communication and data acquisition for innovative network data service on the smart campus.

Despite the investigation of the advantages of edge technologies for adapting and obtaining the most efficient results according to IoT users' requirements, the running of the requested services in isolated and limited edge resources is considered a big challenge to face during the advancement of edge technologies. For that reason, several works have tried to combine edge technologies with unlimited cloud resources to allow the end-users to benefit from both paradigms.

2.1.2. Coordinated Fog-to-Cloud Resources

The extension of cloud services is not the only operation at the edge computing. There is also the Fog-to-cloud (F2C) concept [26], which has been merged into a collaborative model to facilitate an optimal match between cloud and fog resources. With the inception of the F2C, many works have discussed the advantages of putting together cloud and fog resources. For example, in [27], a new workload allocation solution proposed highlighting how the cooperation of F2C could enhance the performance of cloud platforms regarding reducing communication latency. Similarly, in [28], a second fog layer was proposed to enhance service allocation and communication of combined fog-cloud infrastructures. The mF2C [29] is another example of the innovative edge projects that aim to build a global stack of cloud–fog resources to optimize services execution in real-time. Furthermore, interesting work in [30] introduced a Fog-to-Fog communication algorithm to reduce the overall end-to-end delay for F2C by allowing fogs to communicate with each other. The authors highlighted how the edge technologies could play two roles at the same time, which are: middle layers and collaborators for cloud computing.

As one can see, many existing edge solutions are seeking to enhance the processing of IoT data, since they are close to the original sources. Also, they focus on improving the quality of edge

services by collaborating with centralized cloud platforms and exploiting the advancement of network technologies. However, the advantages of edge computing (i.e., low latency) are not the only criteria to pick edge services as the best solutions. Notably, in the fog-to-cloud environment, the IoT clients have the choice to consume services from clouds and edge servers. Yet, this requires a deep understanding of the selection service problem to determine which ones perform the best.

2.2. MCDA in Edge Computing

The MCDA concept [11,12] is a collection of various algorithms that support the selection of the best solution among several alternatives to ensure specific application requirements. Thus, in this subsection, we conduct a review of literature that specifically targets the advantages of applying MCDA algorithms and concepts to edge computing. To this end, we used specific keywords characterizing edge computing and MCDA algorithms, such as MCDA, multiple criteria decision analysis, decision analysis, fog computing, fogging, fog-to-cloud, cloudlet, vehicular ad hoc network based on edge, fog-to-fog. We limited the search to the up-to-date papers over the last three years, which is from 2015 to 2018. We then synchronized the paper selection by the commonly agreed papers, which resulted in 24 references in total. After collecting the search results, each paper underwent a relevance check, during which its relevance to edge technologies was verified. The results are summarized in Table 1, which gives a global overview of edge technologies based on MCDA algorithms.

Table 1. Applied multiple criteria decision analysis methods in edge technologies.

Year	Paper	Main Contribution	MCDA Method Used
2018	[31]	Propose a new multi-dimensional hierarchical performance evaluation model for edge platforms.	AHP (Analytic Hierarchy Process)
2018	[32]	Propose an adaptive nature inspired smartFog.	NSGA-II multi-objective
2018	[33]	Make decision whether to execute tasks locally or offload them to fog servers.	Fuzzy-logic
2018	[34]	Propose container migration algorithms to support mobility tasks.	Markov Decision Process
2015	[35]	Propose a dynamic Service Migration in Mobile Edge-Clouds.	Markov Decision Process
2018	[36]	Propose a Lightweight Auto-Scaling Mechanism for Fog Computing by integrating hypervisor technique with container virtualization.	Fuzzy-logic
2018	[37]	Propose a hybrid model for clinical decision support system in fog–cloud computing.	Neural network
2018	[38]	Propose two semi-Markov decision process-based coordinated virtual machine allocation methods to balance the tradeoff in cloud-fog platforms.	Markov decision process
2018	[39]	Optimize the scheduling of Fog computing job based on bees swarm.	Fuzzy theory
2018	[40]	Propose a framework for data processing in fog computing.	fuzzy
2017	[41]	Evaluate the trustworthiness of the fog servers.	Fuzzy
2017	[42]	Propose a linguistic framework based on fogging for knowledge inference in wearable devices.	Fuzzy
2017	[43]	Propose a secure trust model based in vehicular ad hoc networks.	Fuzzy logic
2015	[44]	Deploy and evaluate an extended CMA (Context-Aware Mobile Approach) hub-service in fogging.	Fuzzy logic and neural networks
2017	[45]	Describe a distributive and central traceable stream mechanism for traceability process in fogging.	Fuzzy
2016	[46]	Propose a dynamic load balancing mechanism in fog computing.	Graph repartitioning
2016	[47]	Propose a framework based on fogging for managing resources in Vehicular Ad Hoc.	Fuzzy logic
2016	[48]	Propose an adaptive edge router for monitoring and controlling the network quality-of-service changes within heterogeneous networks.	Fuzzy weighted queueing
2015	[49]	Propose a framework for selecting an adaptive mobile cloud service from clouds or cloudlets.	AHP and TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution)
2015	[50]	Select a candidate among cloud, cloudlet or mobile devices to save energy consumption and increase the service availability for user (the distances of the fuzzy positive-ideal solution and fuzzy negative-ideal solution are used to rank the alternatives).	TOPSIS and Fuzzy
2018	[51]	Propose an online offloading framework for mobile edge computation offloading.	Binary decision

Table 1. Cont.

Year	Paper	Main Contribution	MCDA Method Used
2015	[52]	Propose a middleware for resource allocation and offloading management of neighboring mobile devices.	TOPSIS
2017	[53]	Redirect users' requests dynamically from host cloudlet to other cloudlets based on the latest network and server status.	Neuro-Fuzzy
2016	[54]	Propose a hybrid approach based on parameters of the mobile device and offloading servers for mobile code offloading.	Markov decision process

2.3. Discussion

After reviewing the selected papers, we found that the recent literature has focused on using edge technologies as a new concept that provides localized computing resources for reducing the latency, power consumption of IoT devices, and time of data processing at the edge network. Furthermore, among the selected works, we found that the usability of MCDA algorithms is still in an early stage at edge computing, such as using fuzzy for making decision of whether to execute tasks locally or offload them to fog servers [33], or for monitoring and controlling the network quality-of-service changes within heterogeneous networks at edge computing [48]. Another work proposed a framework based on AHP (Analytic Hierarchy Process) and TOPSIS algorithms for selecting an adaptive mobile cloud service from clouds or cloudlets [49]. Similarly, in [50], the authors used the distances of the fuzzy positive-ideal solution and fuzzy negative-ideal solution to rank the alternatives and select a candidate among clouds, cloudlets or mobile devices to save energy consumption and increase the availability of services for the end-users. However, in the era of edge computing, many systematic challenges should be considered such as what is the definition of the best practice for choosing sophisticate services? Is it to select services from fog-to-cloud platforms, cloud servers, or only fog nodes?

Indeed, edge computing comes to resolve the deficiencies of cloud services by realizing ubiquitous computing in real-time. Similarly, cloud services enhance the functionalities of edge platforms by collaborating with them. In other words, the concept of fog computing enables us to run the IoT tasks in a secluded environment from the rest of the network, which could cause a failure of fogging to satisfy the requirement of large processing capacities of IoT services that can only be served by the cloud infrastructure. On the other hand, IoT sensors sensitive to latency must resort to fog computing in terms of the communication network. Thus, the combination of cloud and fog services will reduce their defects, since they are highly complementary to each other. As a result, this collaborative model could define new challenges, which pushes us to rethink how to use the combined resources across networks as well as where to select an optimal service among cloud or fog platforms. Notably, the growth of edge/cloud services imposes the need for comparison of their functionalities (i.e., performance) before starting the selection process. For solving this issue, middleware is required to enhance the communication between this collaborative model and IoT applications. In this case, the middleware could interact with centralized or decentralized platforms and choose which one of them could offer services on a smaller scale in real-time. Yet, the middleware must be adapted to manage the selection of the centralized or decentralized services. Also, it should guarantee the integrity and availability of suitable services for the IoT end-clients. Consequently, we identified the following research questions: What are the most critical factors that are necessary for developing anticipatory middleware of users' requests? and how could the middleware ensure the scalability, reliability, and elasticity of services selected from edge or cloud platforms?

3. Preliminary Foglet-Middleware Design

In this section, we describe the preliminary architecture and design of our proposed solution, which is a middleware that allows choosing one service from diverse candidates or alternatives belonging to diverse fog and cloud platforms.

3.1. General Description of the Proposed Foglet

In this section, we propose a foglet as a middleware that aims at performing the interaction between the customers and fog-to-cloud providers (Figure 1). When a user delegates a task to fog-to-cloud platforms, it sends a request to the foglet, where, the handler module ensures the communication between the end-client and the foglet by intercepting a request and sending back the results. Then, the request is processed by the selection service manager that is responsible for the management of data in this architecture. In fact, this module will first check the available services that belong to centralized and decentralized platforms. To do that, it sends a notification to the service repository that contains a list of available QoSs related to the edge and cloud services. Then, a list of edge-cloud candidates is selected according to the user's request and comes back to the selection service manager. Following this, the menu of selected services is sent to the module of rank services for evaluation of the quality of services according to defined criteria. At that point, the Fuzzy-TOPSIS MCDA methods will be applied to choose one alternative (or service) as an optimal solution. To do that, the MCDA methods will evaluate the alternatives in terms of criteria. Then, they will select a more appropriate service according to the request, which will be redirected by the scheduler to a selected fog/cloud service. After sending back the results, the scheduler, which ensures the communication between the foglet and services, will intercept the response and redirect it to the handler module that will return it to the end-user.

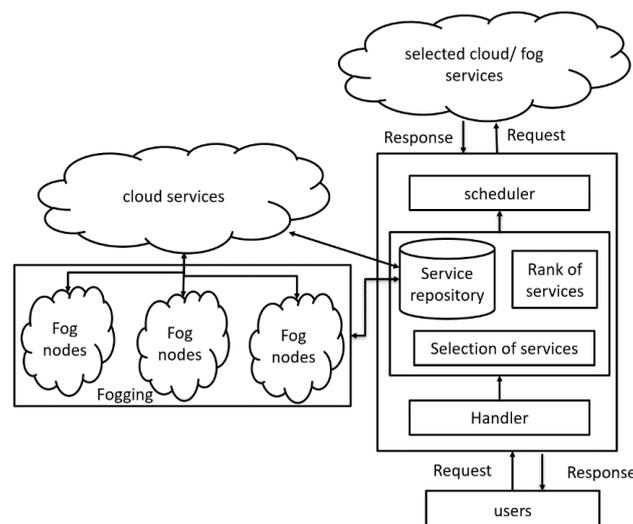


Figure 1. Foglet architecture.

3.2. Selection of Criteria

IoT technologies are becoming part of our everyday lives. For that reason, many works [55] have focused on improving connectivity and communication among numerous IoT devices as well as making their systems self-contained and autonomous by collecting, generating, and extracting accurate and timely information. To enhance further this vision, the benefits of the fog and cloud services are used effectively and efficiently for ensuring the sustainability of IoT services. However, the selection of the adequate services is a crucial task since it is based on a set of quantitative and qualitative parameters belonging to the characteristics of the fog-to-cloud platforms. This coordinated paradigm adopts two models, which are: the decentralized model that offers quick, even real-time, interaction and location-based services to closer smart objects, and the centralized model that is perfect at batch processing and provides the unlimited proper services necessary for the treatment of IoT data (Figure 2). Therefore, the choice of criteria for establishing preference relations between fog and cloud services depends on understanding the nature of the collaborative model.

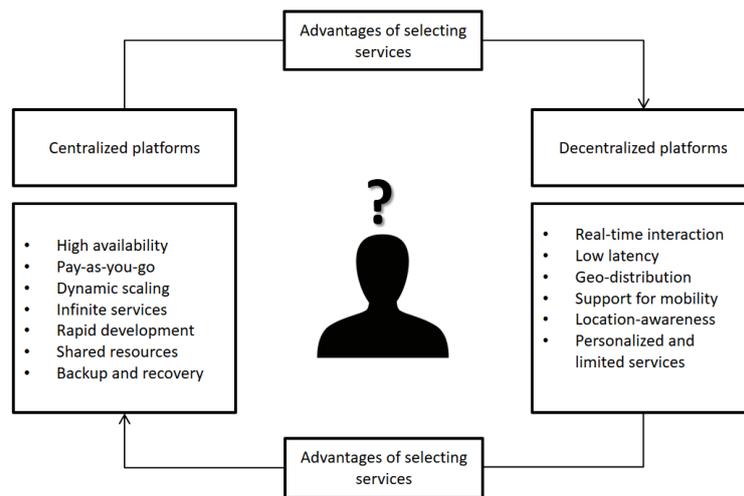


Figure 2. Advantages of selecting decentralized and centralized services.

In this paper, we identify and analyze the criteria that have a direct influence on the selection phase and express clearly the most important characteristics of fog and cloud platforms. We consider ten criteria (Table 2), which are: usability, scalability, adaptability, performance, proximity, mobility, cost, network, availability, and security to be considered for this problem of fog/cloud service selection.

Table 2. Criteria.

Criteria	Description
Availability	It refers to the guarantee of the level of availability in the edge/cloud services.
Usability	It describes the use of fog/cloud services to support users’ requirements.
Scalability	It is the capacity of fog/cloud services to fit a given problem and use resources effectively.
Adaptability	It is the ability of fog/cloud services to adjust according to users’ requests.
Performance	It defines the quality of service provided by the fog/cloud providers.
Proximity	It describes the closeness relationship between the users and fog/cloud services by using geographic information to satisfy different users’ demands at different locations.
Mobility	It is the ability to provide suitable fog/cloud services to support timely the high mobility of applications.
Cost	It represents the cost of fog/cloud resources.
Network	It refers to different network types that have a large impact on the communication between users and services.
Security	It is the ability of fog/cloud providers to keep the high confidence of privacy and data safety that could be attained by the fog/cloud services.

3.3. Ranking Fog/Cloud Services

On having the architecture of the proposed solution and selection criteria, the next step is to devise the ranking that will be used within the Rank of services component of the foglet architecture (Figure 1). This section is devoted to the description of the process of rating the alternatives, while Section 4 guides the reader through a specific computation of the ranking tables, together with the validation of the results.

Based on Table 1 and our previous study [11] in which we described how the most certain and uncertain MCDA methods enhance the offloading process in mobile cloud computing as well as the selection of cloud services in general, we found that the TOPSIS is the most proposed ranking approach in the literature, since it is one of the famous fundamental multi-criteria decision-making ranking approaches in MCDA. The main features of TOPSIS [56] are chosen as the alternatives that simultaneously have the shortest distance from the ideal solution and the farthest distance from the

anti-ideal solution. Also, among the selected works, we notice that the fuzzy method is prevalently used in mobile cloud offloading since this method is characterized by using linguistic variables to describe fuzzy terms that are then mapped to numerical variables. Moreover, it deals effectively with uncertain and imprecise information to solve real-world problems in different domains including cloud computing. Furthermore, the success of fuzzy and TOPSIS methods is measured by solving several MCDA problems, such as management of research and development projects as a portfolio of investments [57]. Accordingly, in this paper, we focus on using the benefits of these MCDA methods to solve the selection of services in the fog-to-cloud environment. Note that for understanding the space and time complexity of the solution, existing works on MCDAs can be consulted [12,56,58].

To do that, our computational procedure of the ranking consists of two parts. Part 1 applies fuzzy similarity [58] to deal with the imprecise or vague nature of linguistic assessment through a trapezoidal fuzzy number (Figure 3). Then, we calculate the fuzzy weight of each criterion identified for the decision problem. Part 2 applies the fuzzy weight of each criterion for the TOPSIS based model to rate and select the best candidate based on the score of each alternative (Figure 4). In the next section, the computation is demonstrated on a numerical example via MATLAB (see Appendix A).

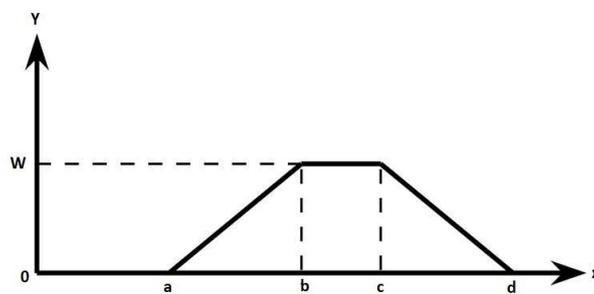


Figure 3. A generalized trapezoidal fuzzy.

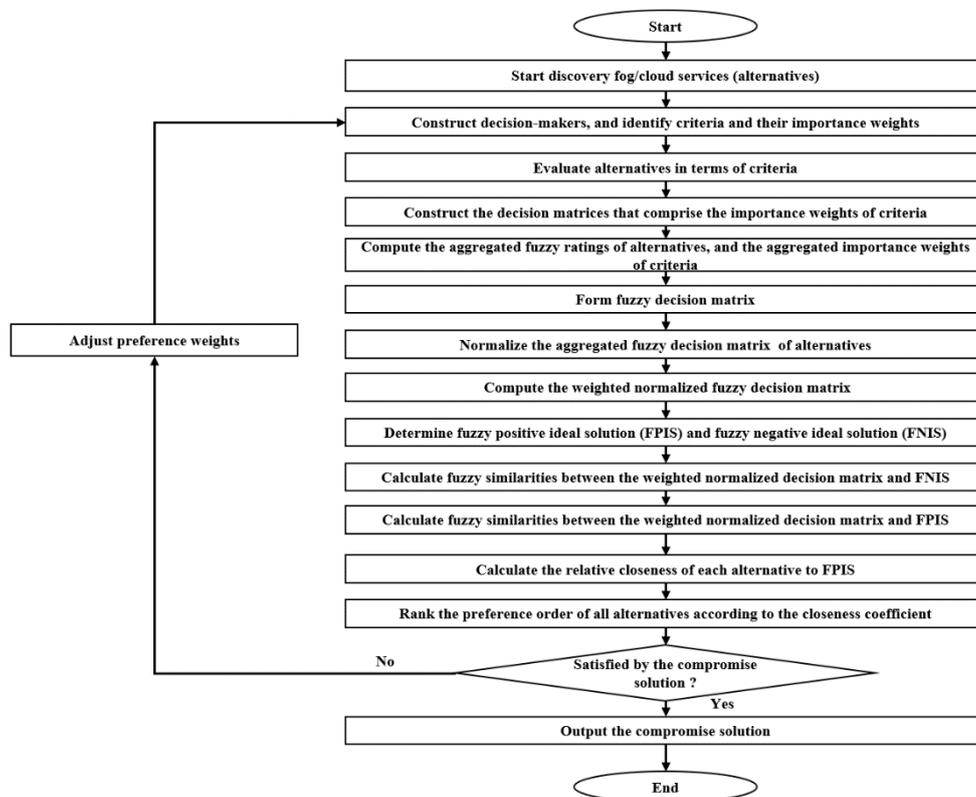


Figure 4. Flowchart of the service selection based on Fuzzy similarity TOPSIS.

4. Ranking Procedure Applied

Let us assume the following setup that will be used as the basis of the ranking demonstration performed in this section. First, consider four alternatives A1, A2, A3, and A4 as services operated on fog and cloud platforms. These could be for instance the available candidates that could reach a satisfactory agreement between the providers and end-users with specific needs. Then consider a committee of three decision makers D1, D2, and D3, formed to select the best alternative. We assume also the priority of importance ranked as: Availability(C1) > Adaptability(C2) > Usability(C3) > Performance(C4) > Cost(C5) > Scalability(C6) > Proximity(C7) > Mobility(C8) > Network(C9) > Security(C10) when choosing the best fog or cloud service. However, the priority of these ten criteria can be varied in other cases, for instance to reflect different preferences of different stakeholders. The whole computational procedure with a numerical example can be summarized as follows:

1. Form a committee of decision-makers (Table 5); the fuzzy rating of each decision-maker can be represented as trapezoidal fuzzy numbers (Figure 3); the linguistic terms for ranking the alternatives and evaluation criteria are shown in Tables 3 and 4 successively.
2. Present the importance weights of the criteria determined by the formed decision-makers D1, D2, and D3 (Table 5).

Table 3. Linguistic terms for ranking the criteria.

Linguistic Values	Fuzzy Ranges
Very low (VL)	[0 0 0.1 0.2 1]
Low (L)	[0.1 0.2 0.2 0.3 1]
Medium low (ML)	[0.2 0.3 0.4 0.5 1]
Medium (M)	[0.4 0.5 0.5 0.6 1]
Medium high (MH)	[0.5 0.6 0.7 0.8 1]
High (H)	[0.7 0.8 0.8 0.9 1]
Very high (VH)	[0.8 0.9 1.0 1.0 1]

Table 4. Linguistic terms for ranking the alternatives.

Linguistic Values	Fuzzy Ranges
Very Poor (VP)	[0,0,1,2,1]
Poor (P)	[1,2,2,3,1]
Medium Poor (MP)	[2,3,4,5,1]
Fair(F)	[4,5,5,6,1]
Medium Good (MG)	[5,6,7,8,1]
Good (G)	[7,8,8,9,1]
Very Good (VG)	[8,9,10,10,1]

Table 5. Importance weight of criteria from three decision-makers.

Criteria	Decision Makers Assessments		
	D1	D2	D3
C1	VH	VH	VH
C2	H	VH	VH
C3	H	VH	VH
C4	VH	H	VH
C5	H	M	H
C6	VH	VH	H
C7	H	H	VH
C8	H	H	VH
C9	VH	H	VH
C10	VH	VH	VH

1. Rate the four alternatives by the formed decision-makers under the criteria (Table 6).
2. Convert the linguistic variables shown in Tables 5 and 6 into trapezoidal fuzzy numbers to form a fuzzy decision matrix (Table 7).
3. Form the aggregated fuzzy decision matrix (Table 8).
4. Form the normalized aggregated fuzzy decision matrix (Table 9) and weighted normalized fuzzy decision matrix (Table 10).
5. Determine the fuzzy positive ideal solution (FPIS) and fuzzy negative ideal solution (FNIS) (Table 11).
6. Calculate the similarity of each alternative from the fuzzy positive ideal solution (see Appendix A).
7. Rank the alternatives according to the average similarity value (Table 12).
8. In this numerical example, the ranking order to the four alternatives is $A1 > A2 > A4 > A3$. Therefore, the alternative A1 is the best service to process the user’s request.

Table 6. Ratings of the four alternatives by three decision-makers under ten criteria.

Criteria	Alternatives	Decision Makers Assessments		
		D1	D2	D3
C1	A1	G	MG	F
	A2	G	G	F
	A3	G	MG	G
	A4	F	P	VG
C2	A1	F	MG	G
	A2	P	MP	MP
	A3	G	MG	F
	A4	G	MG	G
C3	A1	MG	G	F
	A2	G	MG	F
	A3	MP	G	G
	A4	G	P	VG
C4	A1	MG	F	MG
	A2	MP	MP	P
	A3	MG	F	F
	A4	MG	MG	VG
C5	A1	G	F	MG
	A2	MG	MG	P
	A3	MG	G	F
	A4	G	F	G
C6	A1	MG	F	MG
	A2	MP	MP	P
	A3	MG	F	MG
	A4	G	P	VG
C7	A1	G	MG	G
	A2	MG	G	F
	A3	G	G	MG
	A4	G	MG	G
C8	A1	F	MG	MG
	A2	P	MP	MP
	A3	G	MP	F
	A4	G	MG	G
C9	A1	F	F	MG
	A2	MG	G	MG
	A3	G	P	F
	A4	F	MG	F
C10	A1	F	G	G
	A2	MG	G	F
	A3	MP	MP	G
	A4	VP	MG	F

Table 7. Fuzzy decision matrix.

Criteria	Alternatives	Decision Makers Assessments		
		D1	D2	D3
C1	A1	[7,8,8,9,1]	[5,6,7,8,1]	[4,5,5,6,1]
	A2	[2,3,4,5,1]	[7,8,8,9,1]	[7,8,8,9,1]
	A3	[5,6,7,8,1]	[4,5,5,6,1]	[5,6,7,8,1]
	A4	[7,8,8,9,1]	[2,3,4,5,1]	[4,5,5,6,1]
C2	A1	[7,8,8,9,1]	[7,8,8,9,1]	[4,5,5,6,1]
	A2	[7,8,8,9,1]	[1,2,2,3,1]	[8,9,10,10,1]
	A3	[2,3,4,5,1]	[2,3,4,5,1]	[1,2,2,3,1]
	A4	[7,8,8,9,1]	[5,6,7,8,1]	[7,8,8,9,1]
C3	A1	[7,8,8,9,1]	[5,6,7,8,1]	[7,8,8,9,1]
	A2	[5,6,7,8,1]	[4,5,5,6,1]	[5,6,7,8,1]
	A3	[5,6,7,8,1]	[4,5,5,6,1]	[5,6,7,8,1]
	A4	[4,5,5,6,1]	[4,5,5,6,1]	[5,6,7,8,1]
C4	A1	[4,5,5,6,1]	[1,2,2,3,1]	[8,9,10,10,1]
	A2	[2,3,4,5,1]	[2,3,4,5,1]	[1,2,2,3,1]
	A3	[7,8,8,9,1]	[1,2,2,3,1]	[8,9,10,10,1]
	A4	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]	[5, 6, 7, 8, 1]
C5	A1	[4, 5, 5, 6, 1]	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]
	A2	[5, 6, 7, 8, 1]	[4, 5, 5, 6, 1]	[4, 5, 5, 6, 1]
	A3	[7, 8, 8, 9, 1]	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]
	A4	[7, 8, 8, 9, 1]	[1, 2, 2, 3, 1]	[4, 5, 5, 6, 1]
C6	A1	[1, 2, 2, 3, 1]	[2, 3, 4, 5, 1]	[2, 3, 4, 5, 1]
	A2	[5, 6, 7, 8, 1]	[5, 6, 7, 8, 1]	[8, 9, 10, 10, 1]
	A3	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]	[4, 5, 5, 6, 1]
	A4	[4, 5, 5, 6, 1]	[5, 6, 7, 8, 1]	[4, 5, 5, 6, 1]
C7	A1	[7, 8, 8, 9, 1]	[5, 6, 7, 8, 1]	[4, 5, 5, 6, 1]
	A2	[7, 8, 8, 9, 1]	[4, 5, 5, 6, 1]	[5, 6, 7, 8, 1]
	A3	[7, 8, 8, 9, 1]	[7, 8, 8, 9, 1]	[5, 6, 7, 8, 1]
	A4	[4, 5, 5, 6, 1]	[7, 8, 8, 9, 1]	[7, 8, 8, 9, 1]
C8	A1	[7, 8, 8, 9, 1]	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]
	A2	[5, 6, 7, 8, 1]	[5, 6, 7, 8, 1]	[1, 2, 2, 3, 1]
	A3	[7, 8, 8, 9, 1]	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]
	A4	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]	[4, 5, 5, 6, 1]
C9	A1	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]	[4, 5, 5, 6, 1]
	A2	[5, 6, 7, 8, 1]	[7, 8, 8, 9, 1]	[4, 5, 5, 6, 1]
	A3	[4, 5, 5, 6, 1]	[5, 6, 7, 8, 1]	[5, 6, 7, 8, 1]
	A4	[2, 3, 4, 5, 1]	[2, 3, 4, 5, 1]	[7, 8, 8, 9, 1]
C10	A1	[7, 8, 8, 9, 1]	[5, 6, 7, 8, 1]	[4, 5, 5, 6, 1]
	A2	[7, 8, 8, 9, 1]	[4, 5, 5, 6, 1]	[7, 8, 8, 9, 1]
	A3	[1, 2, 2, 3, 1]	[2, 3, 4, 5, 1]	[2, 3, 4, 5, 1]
	A4	[0, 0, 1, 2, 1]	[5, 6, 7, 8, 1]	[4, 5, 5, 6, 1]

Table 8. Aggregated fuzzy decision matrix.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
A1	[4,6.333,6.666,9]	[4, 7, 7, 9]	[5,7.3333,7.6666,9]	[1,5.3333,5.6666,10]	[4,6.3333,6.6666,9]	[1,2.666,3.3333,5]	[4,6.3333,6.6666,9]	[5,7.3333,7.6666,9]	[4,6.3333,6.6666,9]	[4,6.3333,6.6666,9]
A2	[2,6.333,6.666,9]	[1,6.3333,6.6666,10]	[4,5.6666,6.3333,8]	[1,2.6666,3.3333,5]	[4,5.3333,5.6666,8]	[5,7,8,10]	[4,6.3333,6.6666,9]	[1,4.6666,5.3333,8]	[4,6.3333,6.6666,9]	[4,7,7,9]
A3	[4,5.6666,6.3333,8]	[1,2.6666,3.3333,5]	[4,5.6666,6.3333,8]	[1,6.3333,6.6666,10]	[5,7.3333,7.6666,9]	[4,6.3333,6.6666,9]	[5,7.3333,7.6666,9]	[5,7.3333,7.6666,9]	[4,5.6666,6.3333,8]	[1,2.6666,3.3333,5]
A4	[2,5.3333,5.6666,9]	[5,7.3333,7.6666,9]	[4,5.3333,5.6666,8]	[5,6.6666,7.3333,9]	[1,5,5,9]	[4,5.3333,5.6666,8]	[4,7,7,9]	[4,6.3333,6.6666,9]	[2,4.6666,5.3333,9]	[0,3.6666,4.3333,8]

Table 9. Normalized aggregated fuzzy decision matrix.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
A1	[0.4444,0.7037, 0.7407,1]	[0.4,0.7, 0.7,0.9]	[0.5555,0.8148, 0.8518,1]	[0.1,0.5333, 0.5666,1]	[0.4444,0.7037, 0.7407,1]	[0.1,0.2666, 0.3333,0.5]	[0.4444,0.7037, 0.7407,1]	[0.5555,0.8148, 0.8518,1]	[0.4444,0.7037, 0.7407,1]	[0.4444,0.7037, 0.7407,1]
A2	[0.2222,0.7037, 0.7407,1]	[0.1,0.6333, 0.6666,1]	[0.4444,0.6296, 0.7037,0.8888]	[0.1,0.2666, 0.3333,0.5]	[0.4444,0.59253, 0.6296,0.8888]	[0.50,0.7, 0.8,1]	[0.4444,0.7037, 0.7407,1]	[0.1111,0.5185, 0.5925,0.8888]	[0.4444,0.7037, 0.7407,1]	[0.4444,0.777, 0.777,1]
A3	[0.4444,0.6296, 0.7037,0.8888]	[0.1,0.2666, 0.3333,0.5]	[0.4444,0.6296, 0.7037,0.8888]	[0.1,0.6333, 0.6666,1]	[0.5555,0.8148, 0.8518,1]	[0.4,0.6333, 0.6666,0.9]	[0.5555,0.8148, 0.8518,1]	[0.5555,0.8148, 0.8518,1]	[0.4444,0.6296, 0.7037,0.8888]	[0.1111,0.2962, 0.3703,0.5555]
A4	[0.2222,0.5925, 0.6296,1]	[0.5,0.7333, 0.7666,0.9]	[0.4444,0.5925, 0.6296,0.8888]	[0.5,0.6666, 0.7333,0.9]	[0.1111,0.5555, 0.5555,1]	[0.4,0.5333, 0.5666,0.8]	[0.4444,0.7777, 0.7777,1]	[0.4444,0.7037, 0.7407,1]	[0.2222,0.5185, 0.5925,1]	[0,0.4074, 0.4814,0.8888]

Table 10. Weighted normalized fuzzy decision matrix.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
A1	[0.3555,0.6333, 0.7407,1]	[0.28,0.6066, 0.6533,0.9]	[0.3888,0.7061, 0.795,1]	[0.07,0.4622, 0.5288,1]	[0.1777,0.4925, 0.51859,0.9]	[0.07,0.2311, 0.3111,0.5]	[0.3111,0.5864, 0.6419,1]	[0.3888,0.679, 0.7382,1]	[0.3111,0.6098, 0.6913,1]	[0.3555,0.6333, 0.7407,1]
A2	[0.1777,0.6333, 0.7407,1]	[0.07,0.5488, 0.6222,1]	[0.3111,0.5456, 0.6567,0.8888]	[0.07,0.2311, 0.3111,0.5]	[0.1777,0.4148, 0.4407,0.8]	[0.35,0.6066, 0.7466,1]	[0.3111,0.5864, 0.6419,1]	[0.0777,0.4320, 0.5135,0.8888]	[0.3111,0.6098, 0.6913,1]	[0.3555,0.7, 0.7777,1]
A3	[0.3555,0.5666, 0.7037,0.8]	[0.07,0.2311, 0.3111,0.5]	[0.3111,0.5456, 0.6567,0.8]	[0.07,0.5488, 0.6222,1]	[0.2222,0.5703, 0.5962,0.9]	[0.28,0.5488, 0.6222,0.9]	[0.3888,0.67901, 0.7382,1]	[0.3888,0.679, 0.7382,1]	[0.3111,0.5456, 0.6567,0.8888]	[0.0888,0.2666, 0.3703,0.5555]
A4	[0.1777,0.5333, 0.6296,1]	[0.35,0.63555, 0.7155,0.9]	[0.3111,0.5135, 0.5876,0.8888]	[0.35,0.5777, 0.6844,0.9]	[0.0444,0.3888, 0.3888,0.9]	[0.28,0.4622, 0.5288,0.8]	[0.3111,0.6481, 0.6740,1]	[0.3111,0.5864, 0.6419,1]	[0.1555,0.4493, 0.5530,1]	[0,0.3666, 0.4814,0.8]

Table 11. Fuzzy positive and negative ideal solutions (FPIS & FNIS).

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
FPIS	[0.3555,0.6333, 0.7407,1]	[0.35,0.6355, 0.7155,1]	[0.3888,0.7061, 0.795,1]	[0.35,0.5777, 0.6844,1]	[0.2222,0.5703, 0.5962,0.9]	[0.35,0.6066, 0.7466,1]	[0.3888,0.679, 0.7382,1]	[0.3888,0.679, 0.7382,1]	[0.3111,0.6098, 0.6913,1]	[0.3555,0.7, 0.7777,1]
FNIS	[0.1777,0.5333, 0.6296,0.8888]	[0.07,0.2311, 0.3111,0.5]	[0.3111,0.5135, 0.5876,0.8888]	[0.07,0.2311, 0.3111,0.5]	[0.04444,0.3888, 0.3888,0.8]	[0.07,0.2311, 0.3111,0.5]	[0.3111,0.5864, 0.6419,1]	[0.0777,0.432, 0.5135,0.8888]	[0.1555,0.4493, 0.5530,0.8888]	[0,0.2666, 0.3703,0.5555]

Table 12. Fuzzy-TOPSIS Results.

	A1	A2	A3	A4
Similarity of each alternative from FPIS:	0.9078	0.8658	0.8571	0.8535
Similarity of each alternative from FNIS:	0.8016	0.8339	0.8371	0.8328
Calculation of closeness coefficient for the alternatives:	0.5311	0.5094	0.5059	0.5062
Order of the alternatives:	A1 > A2 > A4 > A3			

5. Evaluation and Discussion

To evaluate the ranking computed by our approach (based on the technique form [58]), this section compares the ranking of the alternatives to two other popular techniques [59,60] (see Appendices B and C) that also employ the fuzzy similarity method for solving given MCDA problems by capturing imprecision and inaccurate definition of a decision problem. Also, this section lists all the algorithms here for easy adoption by other research teams. For the evaluation, we adopted the linguistic variables for ranking the alternatives and evaluation criteria as shown in Tables 3 and 4 successively, and we repeated the same calculation steps described in the previous section for [59,60]. Then, we conducted MATLAB simulation to test the results and compare them with [58] (see Appendices B and C). As shown in Figure 5, all the applied methods select A1 as the best alternative for processing the user request. However, the ranking order of the alternatives is not identical because of the difference in the Fuzzy positive ideal solution and closeness coefficient calculation (Figures 6 and 7). Yet, the ranking of the candidates is overall well aligned.

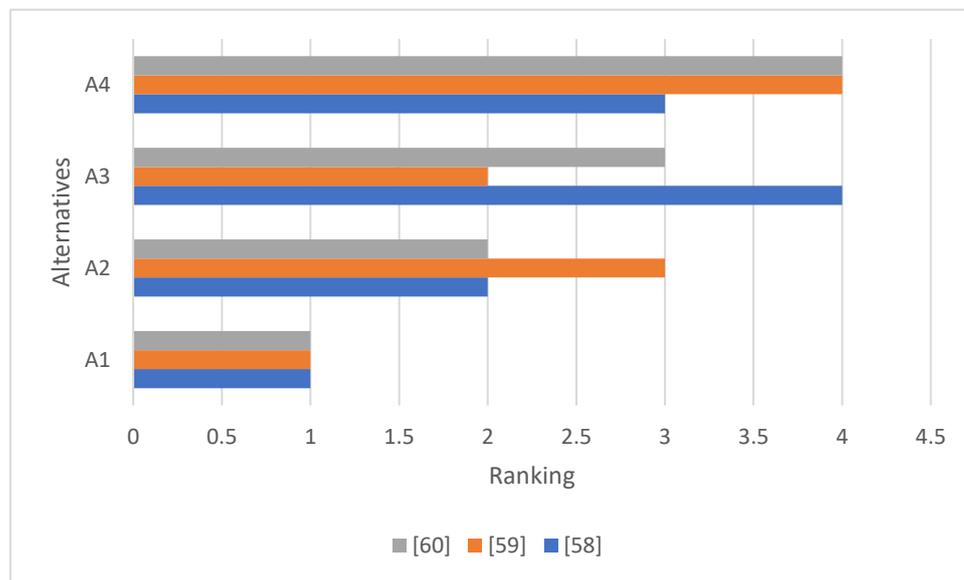


Figure 5. Ranking the alternatives according to [58–60].

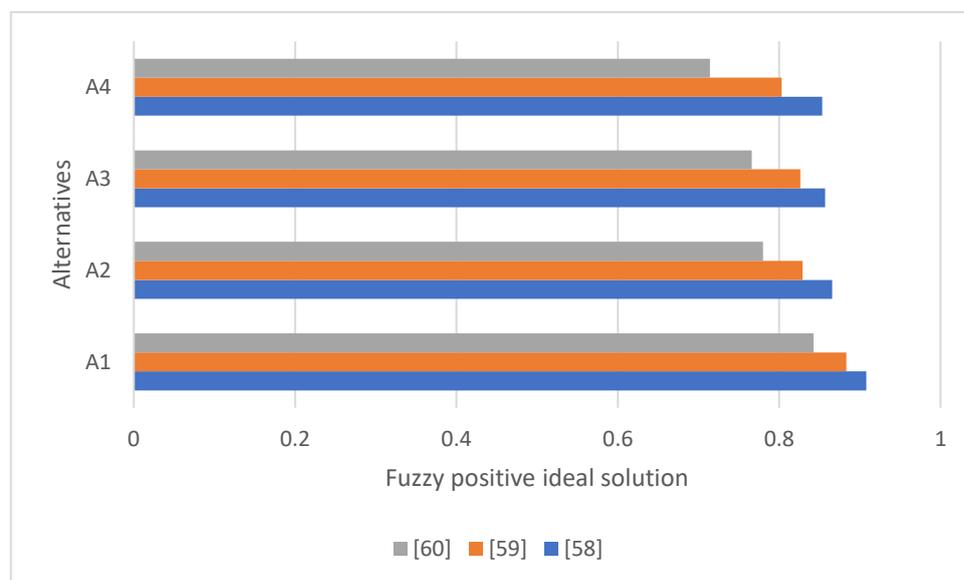


Figure 6. Calculation of Fuzzy positive ideal solution according to [58–60].

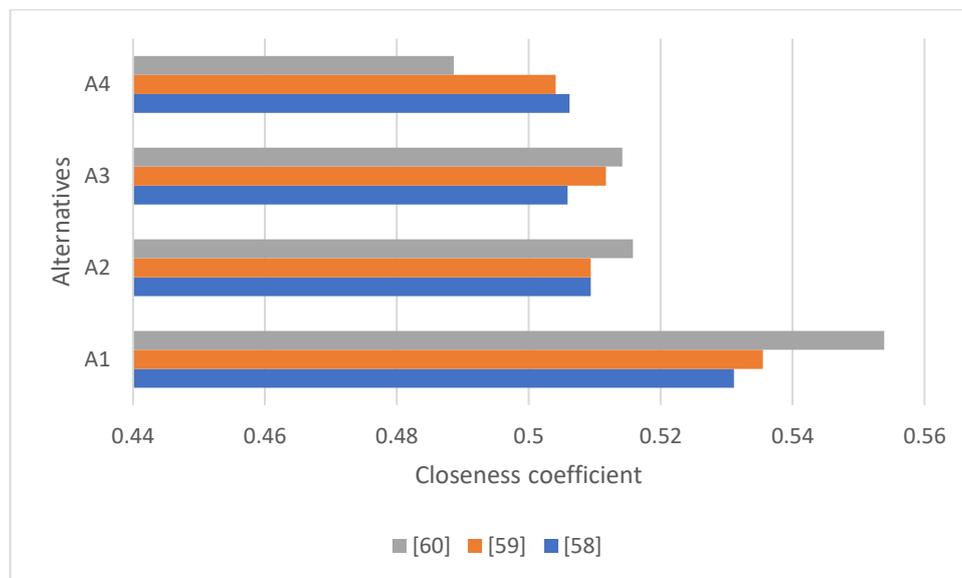


Figure 7. Calculation of closeness coefficient according to [58–60].

Now consider a use case, where a user does not want to use the best-selected alternative due to its high price, for example, and wants to select the second-best alternative according to the proposed ranking list. In this case, A2 would be the selected candidate according to [58,60], while A3 would be selected by [59] (Figure 5). Yet, in this dilemma situation, how could the provider of services identify and determine the best-ranking list of the alternatives and, then, recommend another service to the end-client? and what will be the criteria to select the right one? In this case, it would be very interesting to provide more detailed guidance on the selection among distinct MCDA approaches, which would be an interesting topic for further study that could examine the uncertainty and imprecision issues in multicriteria decision-making for determining the best-ranking list of alternatives.

In fact, the precision of results based on the application of an optimal cloud (or edge) service is the key to successful IoT indoor and outdoor applications. To do that, the selection of cloud or edge services-based MCDA approaches is an essential step that could aid millions of people, who are using IoT devices, for tracking their daily activities. Besides, it could help the providers of services to collect IoT data adequately and process it to extract useful information for improving and monitoring local conditions of IoT applications. Particularly, by 2025, it will be more challenging to process the collected IoT data, since IoT will remarkably exceed 100 billion connected devices [61]. Consequently, this big IoT innovation will create many significant challenges such as how to ensure the quality aspects of IoT data (i.e., selection, verification, and validation of data), save the battery of sensors, optimize routing and prediction of nodes, etc. To face these issues, the providers of services must now be experts not only in the development of applications for IoT paradigm but also in the investigation of the user's feedback for the available services that would be accounted for effective utilization of cloud/edge services. In this context, the MCDA approaches could help the providers of services to understand and clarify the users' preferences, consider each decision criterion, and support communication between them. However, the current shift from service orientation to user orientation in recent innovations, like in the smart city domain, will keep introducing decision-support challenges (i.e., a lack of connection between criteria scales [62]) in designing recommendation services in the edge and cloud environments, stemming from diverging recommendations by different approaches. Accordingly, more attention needs to be paid to MCDA best practices since every progress will be of high value for the areas of academia, research, and industry.

6. Conclusions

Internet of Things (IoT) is a promising technology that uses self-adaptation and self-organization to support the growth of services in smart cities. Recently, the fog-to-cloud paradigm has started to emerge as a means to facilitate the communication among various IoT sensor nodes, supporting the adaptation of services based on the current context of IoT devices. As the selection of the best fitting services and their configuration is a difficult task, this paper proposes a middleware to facilitate the selection of services in fog-to-cloud platforms to maximize their benefits for the end user. In the work, we employed the MCDA algorithm based on fuzzy similarity and TOPSIS methods to deal with incomplete and imprecise information and introduced a technique for ranking of candidates during the selection process in the face of missing information.

In the future, we aim at further improving the prototypical implementation with more complex demonstration of the efficiency of the proposed solution. Moreover, we plan to analyze the complexity of the MCDA technique used in this paper compared to the other MCDA methods like the VIKOR (Vlase Kriterijumska Optimizacija Kompromisno Resenje) method, and study in which situations the recommended ranking by different techniques diverges and why. Also, we will include the identification of additional criteria which may link further the fog and cloud services, increase the ranking score of MCDA methods, and ensure the sustainability and reliability of services in smart cities.

Funding: The work was supported from ERDF/ESF “CyberSecurity, CyberCrime and Critical Information Infrastructures Center of Excellence” (No. CZ.02.1.01/0.0/0.0/16_019/0000822).

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Similarity of Fuzzy Positive Ideal Solution Using MATLAB

```
function Similarity =fuzzySimilarity (A,B,n)
% n: number of criteria
%A: Weighted normalized fuzzy decision matrix
%B: fuzzy positive ideal solution
for i=1:n
P_A=sqrt((A(1)-A(2))^2+A(5)^2)+sqrt((A(3)-A(4))^2+A(5)^2)+(A(3)-A(2))+(A(4)-A(1));
P_B=sqrt((B(1)-B(2))^2+B(5)^2)+sqrt((B(3)-B(4))^2+B(5)^2)+(B(3)-B(2))+(B(4)-B(1));
minmax=(min(P_A,P_B)+min(A(5),B(5)))/(max(P_A,P_B)+max(A(5),B(5)));
S1=1-sum(abs(A(1:4)-B(1:4)))/4;
Similarity (i)=S1*minmax;
End
```

Appendix B. Similarity of Fuzzy Positive Ideal Solution Using MATLAB

```
function Similarity =fuzzySimilarity (A,B,n)
% n: number of criteria
%A: Weighted normalized fuzzy decision matrix
%B: fuzzy positive ideal solution
for i=1:n
PA=sqrt((A(1)-A(2))^2+A(5)^2)+sqrt((A(3)-A(4))^2+A(5)^2)+(A(3)-A(2))+(A(4)-A(1));
PB=sqrt((B(1)-B(2))^2+B(5)^2)+sqrt((B(3)-B(4))^2+B(5)^2)+(B(3)-B(2))+(B(4)-B(1));
aA=1/2*(A(5)*(A(3)-A(2))+A(4)-A(1));
aB=1/2*(B(5)*(B(3)-B(2))+B(4)-B(1));
temp=(min(PA,PB)/max(PA,PB))*(min(aA,aB) + min(A(5),B(5)))/(max(aA,aB) + max(A(5),B(5)));
Similarity (i)=(1-(sum(abs(A(1:4)-B(1:4)))/4)*temp;
end
```

Appendix C. Similarity of Fuzzy Positive Ideal Solution Using MATLAB

```

function Similarity =fuzzySimilarity (A,B,n)
% n: number of criteria
%A: Weighted normalized fuzzy decision matrix
%B: fuzzy positive ideal solution
for i=1:n
if A(1)==A(4)
Ya=A(5)/2;
else
Ya=A(5)*((A(3)-A(2))/(A(4)-A(1))+2);
end
if B(1)==B(4)
Yb=B(5)/2;
else
Yb=B(5)*((B(3)-B(2))/(B(4)-B(1))+2);
end
Xa=(Ya*(A(3)+A(2))+(A(4)+A(1))*(A(5)-Ya))/2*A(5);
Xb=(Yb*(B(3)+B(2))+(B(4)+B(1))*(B(5)-Yb))/2*B(5);
Sa=A(4)-A(1);
Sb=B(4)-B(1);
if (Sa+Sb)/2==0
sasb=0;
else
sasb=1;
end
temp=(1-abs(Xa-Xb))^(sasb)*min(Ya,Yb)/max(Ya,Yb);
S1=1-sum(abs(A(1:4)-B(1:4)))/4;
Simil4=S1*temp;
Similarity (i)=S1*temp;
end

```

References

1. Talari, S.; Shafie-Khah, M.; Siano, P.; Loia, V.; Tommasetti, A.; Catalão, J. A review of smart cities based on the internet of things concept. *Energies* **2017**, *10*, 421. [[CrossRef](#)]
2. Chaudhary, R.; Jindal, A.; Aujla, G.S.; Kumar, N.; Das, A.K.; Saxena, N. LSCSH: Lattice-Based Secure Cryptosystem for Smart Healthcare in Smart Cities Environment. *IEEE Commun. Mag.* **2018**, *56*, 24–32. [[CrossRef](#)]
3. Tanwar, S.; Tyagi, S.; Kumar, S. The Role of Internet of Things and Smart Grid for the Development of a Smart City. In *Intelligent Communication and Computational Technologies*; Springer: Singapore, 2018; pp. 23–33.
4. Menouar, H.; Guvenc, I.; Akkaya, K.; Uluagac, A.S.; Kadri, A.; Tuncer, A. UAV-enabled intelligent transportation systems for the smart city: Applications and challenges. *IEEE Commun. Mag.* **2017**, *55*, 22–28. [[CrossRef](#)]
5. Tomasoni, M.; Capponi, A.; Fiandrino, C.; Kliazovich, D.; Granelli, F.; Bouvry, P. Profiling Energy Efficiency of Mobile Crowdsensing Data Collection Frameworks for Smart City Applications. In Proceedings of the 6th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (IEEE Mobile Cloud 2018), Bamberg, Germany, 26–29 March 2018.
6. Hu, L.; Ni, Q. IoT-driven automated object detection algorithm for urban surveillance systems in Smart Cities. *IEEE Internet Things J.* **2018**, *5*, 747–754. [[CrossRef](#)]
7. Prabhakaran, V.M.; Devatharini, P.; Dineshkumar, L.; Prasanth, K. Building a Big data provenance with its applications for Smart cities. *Int. Res. J. Eng. Technol.* **2018**, *5*, 451–457.
8. Cai, H.; Xu, B.; Jiang, L.; Vasilakos, A.V. IoT-based big data storage systems in cloud computing: Perspectives and challenges. *IEEE Internet Things J.* **2017**, *4*, 75–87. [[CrossRef](#)]

9. Jung, H.S.; Yoon, C.S.; Lee, Y.W.; Park, J.W.; Yun, C.H. Processing IoT Data with Cloud Computing for Smart Cities. *Int. J. Web Appl. IJWA* **2017**, *9*, 88–95.
10. Cheng, B.; Solmaz, G.; Cirillo, F.; Kovacs, E.; Terasawa, K.; Kitazawa, A. FogFlow: Easy Programming of IoT Services Over Cloud and Edges for Smart Cities. *IEEE Internet Things J.* **2018**, *5*, 696–707. [CrossRef]
11. Bangui, H.; Ge, M.; Buhnova, B.; Rakrak, S.; Raghay, S.; Pitner, T. Multi-Criteria Decision Analysis Methods in the Mobile Cloud Offloading Paradigm. *J. Sens. Actuator Netw.* **2017**, *6*, 25. [CrossRef]
12. Whaiduzzaman, M.; Gani, A.; Anuar, N.B.; Shiraz, M.; Haque, M.N.; Haque, I.T. Cloud service selection using multicriteria decision analysis. *Sci. World J.* **2014**, *2014*, 459375. [CrossRef] [PubMed]
13. Openfog. Available online: <https://www.openfogconsortium.org/> (accessed on 13 May 2018).
14. Butterfield, E.H. Fog Computing with Go: A Comparative Study. Bachelor's Thesis, Claremont McKenna College, Claremont, CA, USA, 2016.
15. Ha, K.; Chen, Z.; Hu, W.; Richter, W.; Pillai, P.; Satyanarayanan, M. Towards wearable cognitive assistance. In Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services, Bretton Woods, NH, USA, 16–19 June 2014; pp. 68–81.
16. Accelerating Innovation and Collaboration for the Next Stage. Available online: http://www.ntt.co.jp/news2013/1311ehzt/pdf/xgxf131108d_all.pdf (accessed on 13 May 2018).
17. Announcing the “Edge Computing” Concept and the “Edge Accelerated Web Platform” Prototype to Improve Response Time of Cloud Applications. Available online: <http://www.ntt.co.jp/news2014/1401e/140123a.html> (accessed on 13 May 2018).
18. Truong, N.B.; Lee, G.M.; Ghamri-Doudane, Y. Software defined networking-based vehicular adhoc network with fog computing. In Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), Ottawa, ON, Canada, 11–15 May 2015; pp. 1202–1207.
19. Lin, C.C.; Chin, H.H.; Chen, W.B. Balancing latency and cost in software-defined vehicular networks using genetic algorithm. *J. Netw. Comput. Appl.* **2018**, *116*, 35–41. [CrossRef]
20. Khan, A.A.; Abolhasan, M.; Ni, W. 5G next generation VANETs using SDN and fog computing framework. In Proceedings of the 2018 15th IEEE Annual Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, USA, 12–15 January 2018; pp. 1–6.
21. Xu, Y.; Mahendran, V.; Radhakrishnan, S. Towards SDN-based fog computing: MQTT broker virtualization for effective and reliable delivery. In Proceedings of the 2016 8th International Conference on Communication Systems and Networks (COMSNETS), Bangalore, India, 5–10 January 2016; pp. 1–6.
22. Zhu, J.; Chan, D.S.; Prabhu, M.S.; Natarajan, P.; Hu, H.; Bonomi, F. Improving web sites performance using edge servers in fog computing architecture. In Proceedings of the 2013 IEEE 7th International Symposium on Service Oriented System Engineering (SOSE), Redwood City, CA, USA, 25–28 March 2013; pp. 320–323.
23. Zeng, D.; Gu, L.; Guo, S.; Cheng, Z.; Yu, S. Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system. *IEEE Trans. Comput.* **2016**, *65*, 3702–3712. [CrossRef]
24. Aazam, M.; St-Hilaire, M.; Lung, C.H.; Lambadaris, I. MeFoRE: QoE based resource estimation at Fog to enhance QoS in IoT. In Proceedings of the 2016 23rd International Conference on Telecommunications (ICT), Thessaloniki, Greece, 16–18 May 2016; pp. 1–5.
25. Li, H.; Shou, G.; Hu, Y.; Guo, Z. Mobile edge computing: Progress and challenges. In Proceedings of the 2016 4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), Oxford, UK, 29 March–1 April 2016; pp. 83–84.
26. Bartolí, A.; Hernández, F.; Val, L.; Gorchs, J.; Masip-Bruin, X.; Marín-Tordera, E.; Garcia, J.; Juan, A.; Jukan, A. Benefits of a Coordinated Fog-to-Cloud Resources Management Strategy on a Smart City Scenario. In Proceedings of the European Conference on Parallel Processing, Santiago de Compostela, Spain, 28 August–1 September 2017; Springer: Cham, Switzerland, 2017; pp. 283–291.
27. Deng, R.; Lu, R.; Lai, C.; Luan, T.H. Towards power consumption-delay tradeoff by workload allocation in cloud-fog computing. In Proceedings of the 2015 IEEE International Conference on Communications (ICC), London, UK, 8–12 June 2015; pp. 3909–3914.
28. Souza, V.B.; Ramírez, W.; Masip-Bruin, X.; Marín-Tordera, E.; Ren, G.; Tashakor, G. Handling service allocation in combined fog-cloud scenarios. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 22–27 May 2016; pp. 1–5.
29. MF2C. Available online: <http://www.mf2c-project.eu/> (accessed on 13 May 2018).

30. Masri, W.; Al Ridhawi, I.; Mostafa, N.; Pourghomi, P. Minimizing delay in IoT systems through collaborative fog-to-fog (F2F) communication. In Proceedings of the 2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN), Milan, Italy, 4–7 July 2017; pp. 1005–1010.
31. Zhao, Y.; Feng, X.; Chen, N.; Wang, Y.; Yu, Y.; Wang, H. A multi-dimensional hierarchical performance evaluation model for edge cloud platform. *Procedia Comput. Sci.* **2018**, *129*, 389–393. [[CrossRef](#)]
32. Kimovski, D.; Ijaz, H.; Saurabh, N.; Prodan, R. An Adaptive Nature-inspired Fog Architecture. *arXiv* **2018**.
33. Dhanya, N.M.; Kousalya, G.; Balarkshnan, P.; Raj, P. Fuzzy-Logic-Based Decision Engine for Offloading IoT Application Using Fog Computing. In *Handbook of Research on Cloud and Fog Computing Infrastructures for Data Science*; IGI Global: Hershey, PA, USA, 2018; pp. 175–194.
34. Tang, Z.; Zhou, X.; Zhang, F.; Jia, W.; Zhao, W. Migration Modeling and Learning Algorithms for Containers in Fog Computing. *IEEE Trans. Serv. Comput.* **2018**. [[CrossRef](#)]
35. Wang, S.; Urgaonkar, R.; Zafer, M.; He, T.; Chan, K.; Leung, K.K. Dynamic service migration in mobile edge-clouds. In Proceedings of the IFIP Networking Conference (IFIP Networking), Toulouse, France, 20–22 May 2015; pp. 1–9.
36. Tseng, F.H.; Tsai, M.S.; Tseng, C.W.; Yang, Y.T.; Liu, C.C.; Chou, L.D. A Lightweight Auto-Scaling Mechanism for Fog Computing in Industrial Applications. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4529–4537. [[CrossRef](#)]
37. Liu, X.; Deng, R.H.; Yang, Y.; Tran, H.N.; Zhong, S. Hybrid privacy-preserving clinical decision support system in fog–cloud computing. *Future Gener. Comput. Syst.* **2018**, *78*, 825–837. [[CrossRef](#)]
38. Li, Q.; Zhao, L.; Gao, J.; Liang, H.; Zhao, L.; Tang, X. SMDP-based Coordinated Virtual Machine Allocations in Cloud-fog Computing Systems. *IEEE Internet Things J.* **2018**, *5*, 1977–1988. [[CrossRef](#)]
39. Bitam, S.; Zeadally, S.; Mellouk, A. Fog computing job scheduling optimization based on bees swarm. *Enterp. Inf. Syst.* **2018**, *12*, 373–397. [[CrossRef](#)]
40. Xu, J.; Liu, H.; Shao, W.; Deng, K. Quantitative 3-D shape features based tumor identification in the fog computing architecture. *J. Ambient Intell. Humaniz. Comput.* **2018**. [[CrossRef](#)]
41. Rahman, F.H.; Au, T.W.; Newaz, S.H.; Suhaili, W.S. Trustworthiness in Fog: A Fuzzy Approach. In Proceedings of the 2017 VI International Conference on Network, Communication and Computing, Kunming, China, 8–10 December 2017; pp. 207–211.
42. Medina, J.; Espinilla, M.; Zafra, D.; Martínez, L.; Nugent, C. Fuzzy Fog Computing: A Linguistic Approach for Knowledge Inference in Wearable Devices. In Proceedings of the International Conference on Ubiquitous Computing and Ambient Intelligence, Philadelphia, PA, USA, 7–10 November 2017; Springer: Cham, Switzerland, 2017; pp. 473–485.
43. Soleymani, S.A.; Abdullah, A.H.; Zareei, M.; Anisi, M.H.; Vargas-Rosales, C.; Khan, M.K.; Goudarzi, S. A Secure Trust Model Based on Fuzzy Logic in Vehicular Ad Hoc Networks with Fog Computing. *IEEE Access* **2017**, *5*, 15619–15629. [[CrossRef](#)]
44. Ramalho, F.; Neto, A.; Santos, K.; Agoulmine, N. Enhancing ehealth smart applications: A fog-enabled approach. In Proceedings of the 2015 17th International Conference on E-health Networking, Application and Services (HealthCom), Boston, MA, USA, 14–17 October 2015; pp. 323–328.
45. Chen, R.-Y. An intelligent value stream-based approach to collaboration of food traceability cyber physical system by fog computing. *Food Control* **2017**, *71*, 124–136. [[CrossRef](#)]
46. Ningning, S.; Chao, G.; Xingshuo, A.; Qiang, Z. Fog computing dynamic load balancing mechanism based on graph repartitioning. *China Commun.* **2016**, *13*, 156–164. [[CrossRef](#)]
47. Miao, Z.; Li, C.; Zhu, L.; Han, X.; Wang, M.; Cai, X.; Liu, Z.; Xiong, L. On resource management in vehicular ad hoc networks: A fuzzy optimization scheme. In Proceedings of the 2016 IEEE 83rd Vehicular Technology Conference (VTC Spring), Nanjing, China, 15–18 May 2016; pp. 1–5.
48. Jutila, M. An adaptive edge router enabling internet of things. *IEEE Internet Things J.* **2016**, *3*, 1061–1069. [[CrossRef](#)]
49. Zhou, B.; Dastjerdi, A.V.; Calheiros, R.N.; Srirama, S.N.; Buyya, R. A context sensitive offloading scheme for mobile cloud computing service. In Proceedings of the 2015 IEEE 8th International Conference on Cloud Computing (CLOUD), New York, NY, USA, 27 June–2 July 2015; pp. 869–876.
50. Ravi, A.; Peddoju, S.K. Handoff strategy for improving energy efficiency and cloud service availability for mobile devices. *Wirel. Pers. Commun.* **2015**, *81*, 101–132. [[CrossRef](#)]
51. Neto, J.L.; Yu, S.Y.; Macedo, D.F.; Nogueira, J.M.; Langar, R.; Secci, S. ULOOF: A User Level Online Offloading Framework for Mobile Edge Computing. *IEEE Trans. Mob. Comput.* **2018**, *17*, 2660–2674. [[CrossRef](#)]

52. Ghasemi-Falavarjani, S.; Nematbakhsh, M.; Ghahfarokhi, B.S. Context-aware multi-objective resource allocation in mobile cloud. *Comput. Electr. Eng.* **2015**, *44*, 218–240. [[CrossRef](#)]
53. Rashidi, S.; Sharifian, S. Cloudlet dynamic server selection policy for mobile task off-loading in mobile cloud computing using soft computing techniques. *J. Supercomput.* **2017**, *73*, 3796–3820. [[CrossRef](#)]
54. Zannat, H.; Hossain, M.S. A hybrid framework using Markov decision process for mobile code offloading. In Proceedings of the 2016 19th International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 18–20 December 2016; pp. 31–35.
55. Pattar, S.; Buyya, R.; Venugopal, K.R.; Iyengar, S.S.; Patnaik, L.M. Searching for the IoT Resources: Fundamentals, Requirements, Comprehensive Review and Future Directions. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 2101–2132. [[CrossRef](#)]
56. Behzadian, M.; Otaghsara, S.K.; Yazdani, M.; Ignatius, J. A state-of-the-art survey of TOPSIS applications. *Expert Syst. Appl.* **2012**, *39*, 13051–13069. [[CrossRef](#)]
57. Collan, M.; Luukka, P. Evaluating R and D projects as investments by using an overall ranking from four new fuzzy similarity measure-based TOPSIS variants. *IEEE Trans. Fuzzy Syst.* **2014**, *22*, 505–515. [[CrossRef](#)]
58. Wei, S.H.; Chen, S.M. A new approach for fuzzy risk analysis based on similarity measures of generalized fuzzy numbers. *Expert Syst. Appl.* **2009**, *36*, 589–598. [[CrossRef](#)]
59. Hejazi, S.R.; Doostparast, A.; Hosseini, S.M. An improved fuzzy risk analysis based on a new similarity measures of generalized fuzzy numbers. *Expert Syst. Appl.* **2011**, *38*, 9179–9185. [[CrossRef](#)]
60. Chen, S.-J.; Chen, S.-M. A new method to measure the similarity between fuzzy numbers. In Proceedings of the 10th IEEE International Conference on Fuzzy Systems, Melbourne, Australia, 2–5 December 2001; pp. 1123–1126.
61. Taylor, R.; Baron, D.; Schmidt, D. The world in 2025-predictions for the next ten years. In Proceedings of the 2015 10th International Microsystems, Packaging, Assembly and Circuits Technology Conference (IMPACT), Taipei, Taiwan, 21–23 October 2015; pp. 192–195.
62. Marsh, K.D.; Sculpher, M.; Caro, J.J.; Tervonen, T. The use of MCDA in HTA: Great potential, but more effort needed. *Value Health* **2018**, *21*, 394–397. [[CrossRef](#)] [[PubMed](#)]



© 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/>).