

Article

Hesitant Fuzzy Entropy-Based Opportunistic Clustering and Data Fusion Algorithm for Heterogeneous Wireless Sensor Networks

Junaid Anees ^{1,2} , Hao-Chun Zhang ^{1,*} , Sobia Baig ³ , Bachirou Guene Lougou ¹ 
and Thomas Gasim Robert Bona ⁴

¹ School of Energy Science and Engineering, Harbin Institute of Technology, Harbin 150001, China; j.anees@hit.edu.cn (J.A.); 15bf02043@hit.edu.cn (B.G.L.)

² Satellite Control Facility (SCF-L) directorate, SE&T wing, Space & Upper Atmosphere Research Commission, Lahore 54000, Pakistan

³ Department of Electrical and Computer Engineering, Lahore Campus, COMSATS University Islamabad (CU), Lahore 54000, Pakistan; drsobia@cuilahore.edu.pk

⁴ School of Life Science and Technology, Harbin Institute of Technology, Harbin 150001, China; thomasrobert@stu.hit.edu.cn

* Correspondence: hc Zhang@hit.edu.cn; Tel.: +86-451-8641-2328

Received: 13 January 2020; Accepted: 5 February 2020; Published: 8 February 2020



Abstract: Limited energy resources of sensor nodes in Wireless Sensor Networks (WSNs) make energy consumption the most significant problem in practice. This paper proposes a novel, dynamic, self-organizing Hesitant Fuzzy Entropy-based Opportunistic Clustering and data fusion Scheme (HF ECS) in order to overcome the energy consumption and network lifetime bottlenecks. The asynchronous working-sleeping cycle of sensor nodes could be exploited to make an opportunistic connection between sensor nodes in heterogeneous clustering. HF ECS incorporates two levels of hierarchy in the network and energy heterogeneity is characterized using three levels of energy in sensor nodes. HF ECS gathers local sensory data from sensor nodes and utilizes multi-attribute decision modeling and the entropy weight coefficient method for cluster formation and the cluster head election procedure. After cluster formation, HF ECS uses the same techniques for performing data fusion at the first hierarchical level to reduce the redundant information flow from the first-second hierarchical levels, which can lead to an improvement in energy consumption, better utilization of bandwidth and extension of network lifetime. Our simulation results reveal that HF ECS outperforms the existing benchmark schemes of heterogeneous clustering for larger network sizes in terms of half-life period, stability period, average residual energy, network lifetime, and packet delivery ratio.

Keywords: hesitant fuzzy entropy; heterogeneous clustering; wireless sensor networks; opportunistic routing; multi-attribute decision modeling; data fusion

1. Introduction

In WSNs, the wireless sensors are spatially distributed autonomous devices responsible for sensing the change in the required physical phenomena of their surrounding environment using a small microprocessor, a few transducers, a radio transceiver and a low-power battery [1,2]. These wireless sensor devices collaborate with each other for data sensing, data collection and aggregation purposes [2,3]. In order to reduce the communication overhead and to allocate resources to sensor nodes effectively, we need a topology architecture in which sensor nodes are organized in clusters. Each cluster includes one Cluster Head (CH) and several Cluster Members (CM) [4]. The multi-hop routing used in clustering topology to forward the sensed data from source to destination results in the

overall decrease in energy consumption and also reduces the interference among sensor nodes due to specific allocation of timeslots for communication purposes [4,5]. The clustering topology in WSN can effectively optimize the data redundancy by significantly reducing the size of the collected data using data aggregation and data fusion techniques at CH level. The aggregated or fused data can then be forwarded to the Base Station (BS) for further processing and accurate decision making of interested events [4–6].

The recent literature shows that researchers have proposed a working–sleeping cycle strategy in WSN to save battery power in case the sensor nodes are idle and not performing any of the designated tasks. Alfayez et al. [7] discussed that the sensor nodes go to sleep to save their battery power and wake up before performing their routine operations. These node scheduling techniques can be categorized as synchronous and asynchronous working–sleeping scheduling. These node scheduling techniques are designed in accordance with the scenario to prolong the network lifetime and improve energy utilization by creating an opportunistic node connection between sensor nodes. According to [8–12], Opportunistic Routing (OR) is a paradigm for wireless networks that benefits from broadcast characteristics of a wireless medium by selecting multiple sensor nodes as candidate forwarders. In [10–12], a set of nodes were selected as potential forwarders that transmitted the data packet according to some special criteria after receiving data packet from their neighbors. This set of sensor nodes is called a Candidate Set (CS). The performance of OR significantly depends on several key factors, such as the OR metric, the candidate selection algorithm, and the candidate coordination method. The idea of OR can also be exploited in clustering topology of WSN to extend network lifetime, improve energy utilization, increase the packet delivery ratio (PDR) by adapting asynchronous working–sleeping cycle strategy.

Entropy in information theory is employed in various fields involving data analysis. Entropy uses the Probability Distribution Function (PDF) to statistically measure the degree of uncertainty in information sources [13]. The entropy $H(X)$ of a random variable $X = \{x_1, x_2, \dots, x_n\}$ having probability distribution as $p(X)$ can be given as $H(X) = - \sum_{x \in X} p(x) \log_2 p(x)$ for $0 \leq H(X) \leq 1$ [14–16]. As we want to utilize the functionality of information entropy in clustering WSN, it should be kept in focus that CH or BS should not be hesitant or irresolute about any of their decisions regarding cluster formation and data fusion. Keeping in view the opportunistic connection between sensor nodes in heterogeneous clustering, we selected multiple parameters including an asynchronous working–sleeping cycle, status transition frequencies, residual energy, link quality factor in terms of signal-to-noise ratio, distance between sensor node and BS, and number of supported sensor nodes by a potential CH as our attributes of hesitant fuzzy set. Furthermore, we need Multi-Attribute Decision Modeling (MADM) to efficiently utilize our hesitant fuzzy set to generate hesitant fuzzy entropy matrix and determine our entropy weight coefficients [14,15,17,18].

In this paper, we propose a hesitant fuzzy entropy based opportunistic clustering and data fusion scheme in which a hesitant fuzzy set is created by acquiring the multi-attribute values of sensor nodes. Moreover, the hesitant entropy matrix is generated after finding entropy values of each attribute in the hesitant fuzzy set using data standardization process. Subsequently, the entropy weight model [14,15,17] is employed to determine the entropy weight coefficients for each sensor node and, finally, the threshold attribute values are determined and then compared with the original attribute values for making a decision about new CH. This entire process is part of the CH election procedure which is initiated by the BS and continued by every CH for all communication rounds. The BS creates a CH election set and adds sensor nodes in this set which have original attribute values greater than threshold attribute values. After that, the BS invokes these added sensor nodes by sending them a CH election set message. Furthermore, the hesitant fuzzy entropy technique is also used for reliable data fusion performed by CHs in such a way that CMs integrate and forward their sensed data to the assigned CH by removing the redundant information from the sensed data. CHs perform data aggregation on integrated data packets by concatenating them into a single larger data packet of a specified length. Later on, CHs exploit hesitant fuzzy entropy and entropy weight coefficients to detect

the change in sensed data periodically and then send the aggregated data to the BS upon detecting that change in sensed data [19,20].

The rest of the paper is organized as follows: Section 2 describes the related research work conducted for heterogeneous clustering in WSNs, entropy-based clustering schemes, hesitant fuzzy entropy, and OR. System modeling is presented in Section 3. Our proposed scheme, HFECs, is presented in Section 4. Section 5 describes the case study of HFECs in neighborhood area networks. The performance evaluation and simulation results are described in Section 6. Finally, Sections 7 and 8 discuss and conclude the paper, respectively, and provide some future research directions.

2. Related Research Work

Various researchers have focused on proposing different routing protocols for WSNs based on different parameters, such as end–end delay, successful packets delivered to sink, network lifetime, overall energy consumption, control packet overhead, and sink node mobility, etc. Ogundile et al. [21] presented a detailed survey for energy efficient and energy balanced routing protocols for WSNs. The authors argued that energy efficiency, packet delivery ratio and average end–end delay are the most critical and significant parameters for delay-tolerant and delay-sensitive applications involving WSNs. Furthermore, in this survey, the taxonomy of cluster-based routing protocols for WSNs was also discussed in terms of the energy efficiency and energy balancing aspects. Routing protocols in WSNs can be segmented into two main categories, i.e., hierarchical and non-hierarchical routing protocols. Non-hierarchical routing protocols are proposed on the basis of overhearing, flooding, and information related to the advertisement of the sink's position through agent node selection, whereas hierarchical routing protocols are proposed on the basis of (i) grid-based, tree-based, cluster-based and area-based routing. Different hierarchical routing protocols have their own merits and demerits, but as far as cluster-based hierarchical routing protocols are concerned, researchers have been challenged with a task of achieving an optimal balance between end–end delay and energy consumption [4–6,21].

Yang et al. [22] introduced an emerging concept—the ‘utilization of working–sleeping cycle’ of sensor nodes to prolong the network lifetime. The working–sleeping cycle can be segmented into two categories—synchronous and asynchronous working–sleeping cycle. Recent studies have revealed that a synchronous working–sleeping cycle in sensor nodes could lead to an improvement in energy consumption, but significant contribution is required for efficient synchronization of sensor nodes. Ng et al. [23] presented an energy-efficient synchronization algorithm for sensor nodes in which counter-based and exponential smoothing algorithms could improve the energy consumption using adaptive adjustment of traffic and wakeup period. Moreover, the asynchronous working–sleeping cycle strategy was explored in recent studies as well. Asynchronous working–sleeping schedules depend on network connectivity requirements in terms of traffic coverage area [8,9]. Mukherjee et al. [24] proposed an asynchronous working–sleeping schedule technique in which required network coverage was achieved with a minimum number of awake sensor nodes. Due to the independent working–sleeping cycle of each sensor node in asynchronous working–sleeping strategy, opportunistic node connection can be established between sensor node and its neighbors, thus we need an Opportunistic Connection Random Graph (OCRG) theory to form a spanning tree formation of sensor nodes having opportunistic node connections between each other. In [25], Norman et al. proposed a novel random graph modeling for heterogeneous sensor networks based on different transmission ranges and a new routing metric supporting opportunistic node connections. Anees et al. [26] proposed an energy-efficient multi-disjoint path opportunistic node connection routing protocol for smart grids (SGs) neighborhood area networks (NAN) inspired by opportunistic connection random graph theory in WSNs with sink mobility. This routing protocol reduces the overall energy consumption, increases the PDR, increases the network lifetime and decreases the end–end network delay against existing benchmark schemes.

Many recent studies have addressed the problem of WSN clustering. Low Energy Adaptive Clustering Hierarchy (LEACH) variants were proposed by Liang et al. [27], Handy et al. [28] and Khediri et al. [29] in which each round of communication is segmented into two phases, i.e., (i) the setup

phase, in which limited number of sensor nodes are selected as CH depending upon their probability values and CMs of that CH, (ii) each CH assigns a particular timeslot to every CM in order to avoid collision. Each CM forwards the sensed data only in that timeslot. Various researchers have also proposed many variants of LEACH in [28–33]. Smaragdakis et al. [31] proposed the heterogeneous aware Stable Election Protocol (SEP), which is based on weighted election probabilities of sensor nodes for selection of CH depending upon their residual energies. In SEP, the energy heterogeneity problem is characterized using node classification, in which nodes are either labeled as normal or advanced. This protocol ensures that, during CH election procedure, CH is selected randomly based on the fraction of energy of each sensor node, thus assuring a uniform energy usage of all sensor nodes. Foregoing this view, Femi et al. [34] presented an extension of SEP which is known as SEP-Enhanced or SEP-E. In SEP-E, the energy heterogeneity problem is characterized using a three-node classification, in which nodes are labeled as normal, intermediate and advanced sensor nodes. According to [34], the main goal of SEP-E is to achieve a robust self-configured heterogeneous WSN with a longer network lifetime of sensor nodes and better energy utilization. Sharma et al. [33] proposed a Heterogeneity-aware Energy efficient Clustering (HEC) protocol, which is based on three different phases and CH is selected in every phase in a different manner, i.e., (i) in the first phase, sensor nodes labeled as advanced are allowed to participate in CH selection, (ii) all sensor nodes, irrespective of their residual energies, are allowed to participate in CH selection procedure with equal probabilities, (iii) for the last phase, direct transmission to BS is preferred instead of clustering, due to low residual energy of sensor nodes.

Qing et al. [35] suggested a novel heterogeneous WSN supporting a Distributed Energy-Efficient Clustering (DEEC) scheme in which energy heterogeneity is characterized using normal and advanced sensor nodes. Additionally, Saini et al. [36] proposed an extension of the DEEC protocol which is known as the Enhanced Distributed Energy-Efficient Clustering (E-DEEC) scheme, in which the energy heterogeneity problem is addressed using three-node classification, i.e., normal, intermediate and advanced. In E-DEEC, the CH selection probabilities are not adapted as per the energy levels of sensor nodes. Javaid et al. [37] proposed a heterogeneous network model based on Enhanced Developed DEEC (ED-DEEC or DEEC-E) which is established on two-node power classification and three energy levels of sensor nodes for dynamically modifying the CH election probability. Manjeshwar et al. [38] introduced a novel energy-efficient protocol known as Threshold sensitive Energy-Efficient Sensor Network (TEEN) protocol for reactive networks. In TEEN, each CM in a cluster takes turns to become the CH for a time interval, called the cluster period. After every cluster period, the CH broadcasts a soft and hard threshold to its CMs along with other attributes. CMs only send the sensed data to its CH by changing the status from working to sleeping, only if the data values are in the range of interest, keeping in view the hard threshold. In addition to this, CMs send the data to CH by changing their status from sleeping to working only if their data values change by at least the soft threshold. Although there are no collisions between data transmissions due to TDMA scheduling, TEEN introduces an extra delay during the reporting of time-critical data.

It has been revealed through a detailed literature review that most of the clustering schemes consider attributes such as residual energy, distance to the BS, etc. as parameters of criteria. Meanwhile, the Entropy-Based Clustering Scheme (EBCS) also considers cluster load in terms of supported sensor nodes as a key parameter of criteria for the CH election procedure. [39]. The cluster load can be defined as the number of CMs that can be efficiently handled and supported by the current CH. EBCS includes remaining energy, distance to the BS, and the sum of distances to neighboring sensor nodes as other parameters of criteria besides cluster load. In EBCS, a new method is introduced to predict the residual energy at the start of the next round of communication based on consumed energy. This new method is used to select the CH for next round of communication. Energy heterogeneity is characterized using sensor nodes labeled as normal, intermediate and advanced in EBCS. In previous studies, several researchers have adopted entropy weight coefficient method for making decision in clustering environment [14–16,18]. Entropy weight-based multi-criteria decision routing is a routing technique in which the next hop decision is based on Multi-Criteria Decision Analysis (MCDA) and Multi-Attribute

Decision Modeling (MADM) using entropy weight coefficients [14,15,17,18]. Qiang et al. [16] presented an optimization of objective and subjective weights based on fuzzy MADM routing for selection of next best hop in a heterogeneous WSN. Authors in [16] utilized the entropy weight coefficient method to avoid excessive deviation from the objective weights. Wang et al. [14] developed an index system for capacity assessment using entropy weight coefficient method.

Xia et al. [40] discussed hesitant fuzzy information aggregation in decision making problems. Xia et al. [41] discussed the hesitant fuzzy entropy, cross entropy, and their usage in MADM applications. Su et al. [42] merged the fuzzy logic method with clustering in order to propose a new data fusion method based on a fault tolerant WSN for improving the availability of communication bandwidth. In order to reduce the data processing load on BS and efficiently distinguish the authenticity of archived data, Izadi et al. [43] proposed a wireless sensor data fusion method based on fuzzy theory to improve the service quality in WSNs. Chaurasia et al. [44] presented an adaptive fuzzy logic algorithm to address the inaccuracies in data fusion. Zhai et al. [45] developed Hesitant Language Preference Relationships (HLPR) to improve the credibility of WSNs by fusing uncertain information and putting forward exact opinion about different WSN schemes. Wang et al. [19] proposed a novel data fusion algorithm inspired by hesitant fuzzy entropy to reduce the redundant sensory data transfer in WSN clustering. In this paper, we have utilized some ideas from state-of-the-art research and provided a detailed solution for optimally handling problems like energy consumption and network lifetime using a novel hesitant fuzzy entropy-based opportunistic clustering and data fusion scheme in WSN.

3. System Modeling

3.1. Network Model

In this paper, we considered a heterogeneous WSN in which all sensor nodes are deployed uniformly in a field bounded by $L \times L (m^2)$ region. Each static sensor node with a unique ID can acquire its neighbors' ID by sharing a probe message as discussed in [22,26]. Also, we assumed that all sensor nodes follow the asynchronous working–sleeping cycle strategy with their working time as W_V and sleeping time as S_V . In this model, we assumed that the BS has unlimited energy, storage space and powerful computation capability for communication between CHs and the BS to collect the aggregated sensory data and perform comprehensive evaluations for detection of an event. Furthermore, we also assumed that each sensor node has its own CH and each sensor node has limited energy and limited storage capacity.

Figure 1 illustrates our network model in which we have considered a single BS, some CHs and many sensor nodes connected to each CH. It is evident from Figure 1 that each sensor node acting as CMs or CHs and BS have fixed locations after deployment in the network. Each CH acquires the sensory data from its CMs. CHs can communicate with the BS or any other CH directly or through multiple hops. Moreover, every sensor node is aware of its position using the energy-efficient localization method and we can estimate the distance between the CM and CH and the CH and BS using the Relative Signal Strength Indicator (RSSI) parameter.

Anees et al. [26] utilized the OR in WSNs using the asynchronous working–sleeping cycle strategy for sensor nodes deployed in neighborhood area networks (NANs) of smart grids (SGs). For the efficient utilization of OR, we have to model it using the working–sleeping schedule (W_v/S_v), and the status transition frequency (F_{ST}) of each sensor node in the network. As the concept of OR is quite realizable in real-time scenarios of WSNs, we adopt the asynchronous working–sleeping cycle strategy for every sensor node deployed in our network. The list of notations is given in Table 1.

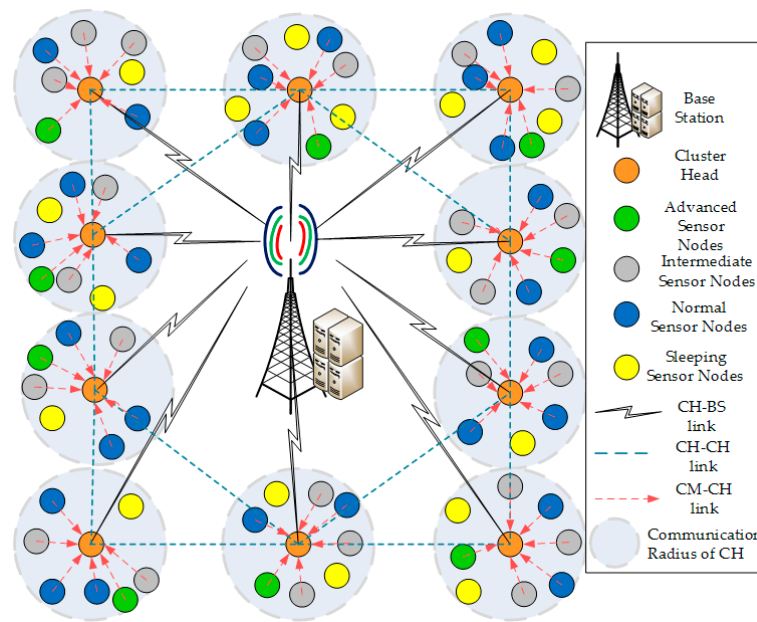


Figure 1. Network Model: heterogeneous wireless sensor networks (WSN) with three types of sensor nodes.

Table 1. Notation description.

Notation	Meaning	Notation	Meaning
E_n	Initial energy of normal sensor nodes	R_L	Long-radio range
E_{int}	Initial energy of intermediate sensor nodes	W_v/S_v	Working/Sleeping schedule
E_{adv}	Initial energy of advanced sensor nodes	F_{ST}	Status transition frequency
μ	Energy weight factor of intermediate sensor nodes	TF_{V_i,V_j}	Time-Frequency parameter
τ	Energy weight factor of advanced sensor nodes	R_E	Residual Energy of a sensor
R_{int}	Ratio of intermediate sensor nodes	LQR	Link quality factor(S/N)
R_{adv}	Ratio of advanced sensor nodes	D_{toBS}	Distance to Base Station
E_{elec}	Energy spent for running radio electronics	$N_{support}$	Number of supported sensors
l	Number of bits	$E_A(\alpha)$	Hesitant Fuzzy Entropy
d_{V_i,V_j}	Euclidean distance b/w transmitter and receiver	$C_A(\alpha, \alpha^c)$	Cross Entropy
ϵ_{fs}	Energy dissipation in free space fading	w_i	Entropy coefficient weight
ϵ_{mp}	Energy dissipation in multi-path fading	D_{Fi}	Final data attribute value
N	Number of sensor nodes	C_R	Communication round
N_C	Number of clusters	$\alpha(x)_{\sigma_i}$	ith value in fuzzy set
E_{sense}	Energy consumed during sensing	$l_{\alpha(x)}$	Length of fuzzy set
E_A	Energy consumed during aggregation	$\alpha, \beta, \gamma, \sigma, \delta$	Weights for $TF_{V_i,V_j}, R_E, LQR, D_{toBS}$ and $N_{support}$
r	Compression ratio	$L_{V_{CM(i)}V_{CM(j)}}$	Link connectivity b/w CM(i) and CM(j)
R_{CC}	Communication–Computation ratio	$L_{V_{CH}V_{CM(j)}}$	Link connectivity b/w CH and CM(j)
E_{CH}	Energy consumed by cluster head	T_{CP}	Data Collection period
E_{CM}	Energy consumed by cluster member	ζ	CH Election procedure set
$E_{cluster}$	Energy consumption of a cluster	T	Time period for sending aggregated data to BS by CH
R_S	Short-radio range	t	Time period for sending integrated data to CH by CM

3.2. Energy Model

We considered three different types of sensor node in terms of energy levels—i.e., normal, intermediate and advanced. If E_n is the initial energy of normal sensor nodes, then the initial energy of the intermediate and advanced sensor nodes should be higher than the normal sensor nodes by a factor of $\mu = \frac{(E_{int}-E_n)}{E_n}$ and $\tau = \frac{(E_{adv}-E_n)}{E_n}$, respectively. Using this information, we can determine the total energy of the sensor nodes in Equation (1) as:

$$E_{total} = N (E_n + R_{int}(E_{int} - E_n) + R_{adv}(E_{adv} - E_n)) = NE_n(1 + R_{int}\mu + R_{adv}\tau) \quad (1)$$

where N is the total number of sensor nodes in the network, R_{int} is the ratio of intermediate sensor nodes and R_{adv} is the ratio of advanced sensor nodes in the network. Furthermore, we considered the simplified energy consumption model based on [26] for radio energy dissipation during transmission and reception. According to this model, the energy required to transmit l bits of data over distance d can be given in Equation (2) as:

$$E_T(V_i, V_j) = \begin{cases} E_{elec}l + \varepsilon_{fs}ld_{V_iV_j}^2 & d < d_0 \\ E_{elec}l + \varepsilon_{mp}ld_{V_iV_j}^4 & d \geq d_0 \end{cases} \quad (2)$$

where E_{elec} is the energy spent by transmitter on running the radio electronics, ε_{fs} is the energy dissipated by power amplifier depending on the Euclidean distance $d_{V_iV_j}$ between the transmitter and receiver when free space fading is considered for distance less than d_0 , and ε_{mp} is the energy dissipated by power amplifier depending on Euclidean distance $d_{V_iV_j}$ between transmitter and receiver when multi-path fading is considered for distance greater than d_0 . The threshold distance is given as $d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}}$. Similarly, the energy required to receive l bits of data over distance d can be computed in Equation (3) as:

$$E_R = E_{elec}l \quad (3)$$

Likewise, the energy spent by CM or CH at the beginning of each round for sensing l bits of data can be analyzed as $E_{sense} = E_{elec}l$. Here, we have assumed that the distance between CM and CH is shorter than the distance between CH and BS. Thus, we use ε_{fs} (free space fading model) for communication between CM and CH and ε_{mp} (multi-path fading model) for communication between CH and BS. Accordingly, the energy consumed by each CM can be calculated in Equation (4) as

$$E_{CM} = E_{sense} + E_T = E_{elec}l + E_{elec}l + \varepsilon_{fs}ld_{V_iV_j}^2 \quad (4)$$

Each CH aggregates the received sensory data and then forwards it towards BS. The energy consumed by each CH can be calculated in Equation (5) as

$$\begin{aligned} E_{CH} &= E_{sense} + \left(\frac{N}{N_C} - 1\right)E_R + \left(\frac{N}{N_C}\right)lE_A + \left(\frac{N}{r}\right)E_T \\ &= E_{elec}l + \left(\frac{N}{N_C} - 1\right)E_{elec}l + \left(\frac{N}{N_C}\right)l\frac{E_{elec}}{R_{CC}} + \left(\frac{N}{r}\right)E_{elec}l \\ &\quad + \left(\frac{N}{r}\right)\varepsilon_{mp}ld_{V_iV_j}^4 \end{aligned} \quad (5)$$

where N_C represents the number of clusters in the network, $\frac{N}{N_C}$ is the working sensor nodes per cluster in which we have 1 CH and $\frac{N}{N_C} - 1$ CMs. E_A represents the energy consumed during aggregating data packets at CH, r is the compression ratio and R_{CC} is the communication to computation ratio used only during aggregation in this case. It is pertinent to mention here that the total energy consumed in a cluster is the sum of energies consumed by all CMs plus the energy consumed by their corresponding CH. The total energy consumed by a cluster $E_{cluster}$ is given in Equation (6) as

$$E_{cluster} = E_{CH} + \sum_{i=1}^{\left(\frac{N}{N_C}-1\right)} E_{CM_i} \quad (6)$$

4. The Proposed Scheme HFECs

Information entropy is the statistical measure of the degree of uncertainty of an information source and it has always played a significant role in uncertainty decision analysis [13]. Likewise, hesitant fuzzy entropy is defined as the statistical measure of degree of hesitant fuzzy uncertainty, which depends on MADM to make a decision about certain events. In heterogeneous clustering schemes of WSNs, the role of CH within a cluster needs to be substituted using some decision criteria

in order to avoid the hotspot problem. Using a single parameter such as the energy level of the sensor nodes is not sufficient enough to make an accurate decision about CH selection after every round of communication. In order to ensure that the best possible working sensor node is chosen as CH in every round of communication, we have to incorporate MADM in our clustering scheme. The hesitant fuzzy set including multi attributes and alternatives could help us out in implementing MADM for opportunistic clustering in WSNs [40,41].

Our main objective is to design a hesitant fuzzy entropy-based self-organizing opportunistic clustering scheme which can prolong the network lifetime and bring an improvement in overall energy consumption of sensor nodes. In our proposed scheme, we determined the entropy weight coefficients using hesitant fuzzy entropy to select the best possible CH through election procedure and perform reliable data fusion at CH level to reduce the information flow. Since we are dealing with MADM problem, we considered several parameters like time frequency parameter $TF_{V_i V_j}$ (depends on W_v/S_v and status transition frequency F_{ST}), residual energy R_E , link quality factor LQR in terms of signal-to-noise ratio, distance to BS D_{toBS} , and number of supported sensor nodes $N_{support}$ as our different attributes of hesitant fuzzy set to compute the hesitant fuzzy entropy.

We have three different types of nodes in our network which can lead to energy heterogeneity and might result in three different types of CH, i.e., CH_{normal} , CH_{int} , and CH_{adv} , with the initial energy E_n , E_{int} , and E_{adv} , respectively, but in order to avoid frequent change in the CH role, we used only two different types of CH, i.e., CH_{int} and CH_{adv} , for calculating $N_{support}$. However, the percentage of energy consumption for CH_{int} (E_{CH}/E_{int}) is more than that of CH_{adv} (E_{CH}/E_{adv}) due to the reason that $E_{adv} > E_{int}$. As the consumed energy in relation to available energy is uneven for CHs, it leads us to determine the optimum number of sensor nodes which can be supported by each type of CH. Therefore, we used the remaining energy and average of E_{int} and E_{adv} to find the optimum number of sensor nodes supported by both CH_{int} and CH_{adv} in Equation (7) as:

$$N_{support} = \frac{(E_{total} - E_{consumed}) * (N - N_C)}{\left(N_C * \frac{(E_{int} + E_{adv})}{2}\right)} \quad (7)$$

4.1. CH Election Procedure

Based on the sensory data being generated by sensor nodes, we make a decision about CH role in the cluster using $TF_{V_i V_j}$, R_E , LQR , D_{toBS} and $N_{support}$. But before that, we have to construct the hesitant fuzzy entropy matrix and determine the corresponding weight coefficients of each attribute. Subsequently, the attribute values are synthesized to determine the threshold attribute values. The threshold attribute values are then compared with original attribute values of each sensor node. Based on their comparison, the BS selects the sensor nodes with original attribute values greater than threshold attribute values and add those sensor nodes in CH election set.

The entropy measures for hesitant fuzzy set were already discussed by Xu et al. in [40] and Xia et al. in [41]. Due to the fact that hesitancy is a common problem in decision making problems, we need to develop some entropy or cross entropy measures for hesitant fuzzy sets. Wang et al. in [19] discussed that if X is a fixed set and the hesitant fuzzy set α when applied on X returns a subset in $[0,1]$. Let $l_{\alpha(x)}$ be the number of values in $\alpha(x)$ where $\alpha(x)_{\sigma_i}$ be the i th smallest value in $\alpha(x)$ in which $i = \{1, 2, \dots, l_{\alpha(x)}\}$. In order to find the cross entropy of two hesitant fuzzy sets, i.e., α and β , we assume that both of them have same length, or, if there is only one value in α , we extend it by repeating that value until it reaches the length of β . Moreover, if there is only one hesitant fuzzy set, then we use α and α^C (complement of α) in the cross-entropy and calculate the hesitant fuzzy entropy in Equations (8) and (9) as

$$E_A(\alpha) = 1 - C_A(\alpha, \alpha^C) \quad (8)$$

$$\text{where } C_A(\alpha, \alpha^C) = \frac{2}{l_\alpha T} \sum_{i=1}^{l_\alpha} \left(\frac{(1+q\alpha_{\sigma(i)}) \ln(1+q\alpha_{\sigma(i)}) + (1+q(1-\alpha_{\sigma(l_\alpha-i+1)})) \ln(1+q(1-\alpha_{\sigma(l_\alpha-i+1)}))}{2} - \frac{2+q\alpha_{\sigma(i)} q(1-\alpha_{\sigma(l_\alpha-i+1)})}{2} \ln \frac{2+q\alpha_{\sigma(i)} q(1-\alpha_{\sigma(l_\alpha-i+1)})}{2} \right) q > 0$$

$$E_A(\alpha) = 1 - \frac{2}{l_\alpha T} \sum_{i=1}^{l_\alpha} \left(\frac{(1+q\alpha_{\sigma(i)}) \ln(1+q\alpha_{\sigma(i)}) + (1+q(1-\alpha_{\sigma(l_\alpha-i+1)})) \ln(1+q(1-\alpha_{\sigma(l_\alpha-i+1)}))}{2} - \frac{2+q\alpha_{\sigma(i)} q(1-\alpha_{\sigma(l_\alpha-i+1)})}{2} \ln \frac{2+q\alpha_{\sigma(i)} q(1-\alpha_{\sigma(l_\alpha-i+1)})}{2} \right) q > 0 \quad (9)$$

where $T = (1+q) \ln(1+q) - (2+q)(\ln(2+q) - \ln(2))$, $q > 0$. Then, we calculate the entropy weight coefficients for all sensor nodes using Equation (10) after generating entropy matrix using Equation (9).

$$w_i = \frac{1 - E_i}{r - \sum_{i=1}^r E_i} \quad (10)$$

where $0 \leq w_i \leq 1$, $\sum_{i=1}^r w_i = 1$ and $E_i = \frac{1}{r} \sum_{j=1}^r E(\alpha_{ji})$. In the next step, we find the average data value

of attributes in measured value matrix, i.e., $D_i = \frac{1}{l_\alpha} \sum_{i=1}^{l_\alpha} \alpha_{\sigma(i)}$ and then synthesize the value of each attribute to find the threshold attribute value using Equation (11) as:

$$D_{Fi} = \frac{1}{r} \sum_{i=1}^r w_i D_i \quad (11)$$

The decision about placing sensor nodes in CH election set depends on the comparison between the original attribute values of sensor nodes in a cluster with that of threshold attribute values. A sensor node can be placed in the CH election set only if the maximum of its original attribute values is greater than the threshold attribute values generated from Equation (11). The pseudocode for this procedure is given in Algorithm 1.

Algorithm 1. CH Election Procedure in HFECS**Input:** Multi-Attribute values of each sensor node**Output:** Sensor nodes in CH election set**Begin:**

```

1.                                     r: number of attributes
2.                                     s: number of sensor nodes
3.                                      $\alpha$ : attribute value
4.                                     for i = 1 to r, do
5.                                         for j = 1 to s, do
6.                                             Measure the attribute values for all the sensors
7.                                             Formation of measured value matrix
8.                                             Standardize the attribute values of all sensors
9.                                         end for
10.                                    Generate the hesitant entropy decision matrix
11.                                    after standardization of each attribute value
12.                                    Calculate the hesitant fuzzy entropy using
13.                                     $E_A(\alpha) = 1 - C_A(\alpha, \alpha^C)$ 
14.                                    Refer to Equations (8) and (9) for calculation of
15.                                    cross entropy  $C_A(\alpha, \alpha^C)$ 
16.                                    Generate the entropy matrix based on calculated
17.                                    hesitant fuzzy entropy for all attributes
18.                                    end for
19.                                    Find the entropy value of each attribute using
20.                                     $E_i = \frac{1}{r} \sum_{j=1}^r E(\alpha_{ji})$ 
21.                                    for i = 1 to r, do
22.                                        Determine the weight value  $w_i$  for all sensor
23.                                        nodes using Equation (10)
24.                                    end for
25.                                    Find the average data value in measured value
26.                                    matrix, i.e.,  $D_i = \frac{1}{l_\alpha} \sum_{i=1}^{l_\alpha} \alpha_{\sigma(i)}$ 
27.                                    for i = 1 to r, do
28.                                        Synthesize the value of each attribute to find the
29.                                        threshold value using Equation (11)
30.                                        if (sensor node with  $\alpha_{\sigma(i)} > D_{Fi}$ ), then
31.                                            Add  $k$  sensor nodes with highest  $(\alpha_{\sigma(i)} > D_{Fi})$ 
32.                                            in CH Election Set
33.                                        end if
34.                                    end for

```

4.2. Cluster Formation

In HFECS, the BS executes the election procedure and creates a CH election set. Initially, when $C_R = 1$, the BS adds k sensor nodes with highest values of $\alpha_{\sigma(i)}$ in the CH election set after executing the election procedure given in Algorithm 1 and then invokes those k sensor nodes by sending them a CH_Election_set message. After receiving the CH_Election_Set message from the BS, the potential CH starts preparing its CH_announce message and then send this message to all working neighbors in that cluster to advertise its role as a potential CH. The pseudocode for this procedure is given in Algorithm 2.

Algorithm 2. HFECs (BS Side)**Input:** CH election procedure**Output:** Potential CH advertising its role and making local decision for data fusion**Begin:**

1. **if** ($C_R = 1$), **then**
2. BS executes the CH election procedure
3. BS creates a CH election set
4. BS adds k sensor nodes in CH election set with highest ($\alpha_{\sigma(i)} > D_{Fi}$)
5. BS invokes k working sensors with highest attribute values in CH election set by sending CH_Election_Set message
6. **end if**
7. Wait for collected data from CHs
8. **if** (final prediction result received from CH), **then**
9. Evaluate the collected data to determine the possibility of event happening
10. **if** (event happened)
11. Perform standard operation procedure
12. **end if**
13. **end if**
14. **end if**

If the CH_announce message is sent by multiple potential CHs, then CMs send Connection Request ‘Con_Req’ message to the CH with highest attribute values $\alpha_{\sigma(4)} > D_{F4}$. For example, if a neighboring sensor node receives CH_announce from multiple potential CHs, then the sensor node becomes a cluster member of that CH which offers better link quality with that neighboring sensor node in comparison to other CHs. However, if the CH_announce message is sent by a single potential CH, then CMs send a Con_Req message to that CH without any delay. Meanwhile, the new CH creates a CM_set and adds all the working CMs in CM_set after receiving the Con_Req messages from CMs. Subsequently, the new CH transmits the TDMA schedule to all CMs in that cluster, keeping in view the asynchronous working–sleeping cycle of CMs. This TDMA schedule for every CM should be selected vigilantly from a CH as the timeslot for forwarding the sensory data to the CH should be within working time of that CM. Each CM extracts its own CM timeslot from the TDMA schedule sent by the CH, stores the CM timeslot in its local buffer and waits until the CM timeslot becomes current timeslot. The pseudocode for CM task execution is given in Algorithm 3.

The construction of OCRG is very important to understand the asynchronous working–sleeping cycle of sensor nodes. Therefore, we need to construct an OCRG to analyze the opportunistic connections between CM-CH and CM-CM in a heterogeneous WSNs. When the CM timeslot becomes the current timeslot, $V_{CM(i)}$ calculates its link connectivity with $V_{CM(j)}$ in terms of time-frequency parameter, residual energy, LQR, distance from the BS and Nsupport as per Equation (12),

$$L_{V_{CM(i)}V_{CM(j)}} = \max_{V_{CM} \in S_j} \{ \alpha TF_{V_{CM(i)}V_{CM(j)}} + \beta RE_{V_{CM(i)}V_{CM(j)}} + \gamma S/N_{V_{CM(i)}V_{CM(j)}} + \sigma D_{toBS(CM(i))} + \delta N_{support(CM(i))} \} \quad (12)$$

where $i = j = 1 : \frac{N}{N_C} - 2$, $TF_{V_{CM(i)}V_{CM(j)}}$ is the time-frequency parameter between $V_{CM(i)}$ and $V_{CM(j)}$, $RE_{V_{CM(i)}V_{CM(j)}}$ is the residual energy of node $V_{CM(i)}$ and $V_{CM(j)}$, $S/N_{V_{CM(i)}V_{CM(j)}}$ is the link quality factor between $V_{CM(i)}$ and $V_{CM(j)}$, $D_{toBS(CM(i))}$ is the distance to the BS from $V_{CM(i)}$ and $N_{support(CM(i))}$ is the number of sensor nodes supported by $V_{CM(i)}$ in case $V_{CM(i)}$ becomes V_{CH} . α , β , γ , σ and δ are the appropriate weights assigned to time-frequency parameter, residual energy, link quality factor, distance to the BS and number of supported sensor nodes, respectively. The time-frequency parameter

depends on working-sleeping cycle W_v/S_v and status transition frequency F_{ST} . We can calculate the time-frequency parameter $TF_{V_{CM(i)}V_{CM(j)}}$ using Equation (13) as

$$TF_{V_{CM(i)}V_{CM(j)}} = \left(\frac{F_{STV_{CM(i)}}}{F_{STmax}} \times \frac{W_{V_{CM(i)}}}{T_{CP}} \right) \left(\frac{F_{STV_{CM(j)}}}{F_{STmax}} \times \frac{W_{V_{CM(j)}}}{T_{CP}} \right) \quad (13)$$

where $W_{V_{CM(i)}}$ and $W_{V_{CM(j)}}$ are the total working time of the adjacent nodes $V_{CM(i)}$ and $V_{CM(j)}$, T_{CP} is the data collection period, $F_{STV_{CM(i)}}$ and $F_{STV_{CM(j)}}$ are the status transition frequencies of the adjacent nodes $V_{CM(i)}$ and $V_{CM(j)}$, and F_{STmax} is the max status transition frequency value obtained during T_{CP} . Then, $V_{CM(i)}$ acquires the sensory data, integrates it while removing the redundant information, and forwards the integrated data along with the link connectivity information to the current CH.

Algorithm 3. HFECs (CM Side)

Input: CH election procedure executed

Output: Executing all tasks relevant to CM after CH election procedure

Begin:

1. **if** (CH_Announce message received from multiple potential CH), **then**
 2. Send the Con_Req to potential CH with highest attribute value $\alpha_{\sigma(4)} > D_{F4}$
 3. **else if** (CH_Announce message received from single potential CH), **then**
 4. Send the Con_Req to that potential CH
 5. **else** Wait till the CH_Announce message is received from any potential CH
 6. **end if**
 7. Wait till the TDMA schedule is received from new CH
 8. **if** (TDMA based schedule received from new CH), **then**
 9. Store the schedule (especially CM timeslot) in local buffer
 10. **if** (timeslot = CM timeslot), **then**
 11. Calculate Link connectivity of $V_{CM(i)}$ with $V_{CM(j)}$ using Equation (12)
 12. Integrate data and remove the redundant and identical info. for every time period T
 13. while $D_{T_i} \neq D_{T_{i-1}}$, **do** // detecting the change in integrated data
 14. Transmit the integrated data D_{T_i} to CH along with $L_{V_{CM(i)}V_{CM(j)}}$
 15. **end while**
 16. **else if** (timeslot \neq CM timeslot), **then**
 17. Wait until timeslot = CM timeslot
 18. **end if**
 19. **end if**
-

Besides the benefits of hesitant fuzzy entropy in the CH election procedure of heterogeneous clustering scheme, it can also help in predicting an event in the heterogeneous clustering scheme and in reducing the number of transmissions in the network by applying reliable data fusion. Let us assume that CH requires time period T to transmit aggregated data packet to BS, and CM requires time period t to transmit the integrated data to CH where $T = nt$. Moreover, we considered context-based data

forwarding, which means that when the CH receives the integrated data from $V_{CM(i)}$, it performs lossless aggregation by concatenating multiple data packets into single packet of specified length. In addition, CH also standardizes the received integrated data, calculates the hesitant fuzzy entropy and assign corresponding weight coefficients to integrated data of all sensor nodes. Then, CH synthesizes the attribute values of each sensor node to determine the threshold attribute values. After that, the threshold attribute values are compared with original attribute values to make a decision about data forwarding to the BS. If the original attribute values (current integrated data) are greater than threshold attribute values, it means that integrated data is changed after time period T, only then the CH transmits the result to the BS so that the BS can initiate the Standard Operating Procedure (SOP) accordingly against the event occurred in the cluster, but if the original attribute values are less than threshold attribute values, it means that the integrated data are not changed after time period T and CHs do not need to forward the integrated data to the BS. In this case, CHs should wait for time period t so that more sensory data are available.

After forwarding the aggregated data packets to BS, CH calculates the link connectivity with all of its CM as given in Equation (14),

$$L_{V_{CH}V_{CM(j)}} = \max_{V_{CM} \in S_j} \left\{ \alpha TF_{V_{CH}V_{CM(j)}} + \beta R_{E,V_{CH}V_{CM(j)}} + \gamma S/N_{V_{CH}V_{CM(j)}} + \sigma D_{toBS(CH)} + \delta N_{support(CH)} \right\} \quad (14)$$

where $TF_{V_{CH}V_{CM(j)}}$ is the time-frequency parameter between V_{CH} and $V_{CM(j)}$, $R_{E,V_{CH}V_{CM(j)}}$ is the residual energy of V_{CH} and $V_{CM(j)}$, $S/N_{V_{CH}V_{CM(j)}}$ is the link quality factor between V_{CH} and $V_{CM(j)}$, $D_{toBS(CH)}$ is the distance to the BS from V_{CH} and $N_{support(CH)}$ is the number of sensor nodes supported by V_{CH} . In this paper, the decision about the next CH is based on the link connectivity information of current CH and CMs. For each $V_{CM(i)}$ in CM_set, we compare the link connectivity $L_{V_{CH}V_{CM(j)}}$ with that of $L_{V_{CM(i)}V_{CM(j)}}$ ($i \neq j$). If $L_{V_{CH}V_{CM(j)}} > L_{V_{CM(i)}V_{CM(j)}}$, we add V_{CH} in CH_Election_Procedure set ζ but if $L_{V_{CH}V_{CM(j)}} < L_{V_{CM(i)}V_{CM(j)}}$, then we add $V_{CM(i)}$ to CH_Election_Procedure set ζ .

If the current V_{CH} is the only sensor node in CH_Election_Procedure set ζ , then current V_{CH} will be the new V_{CH} for next round C_{R+1} but if there is only one sensor node in the CH_Election_Procedure set ζ and $V_{CH} \notin \zeta$, then the sensor node V_{CM} present in ζ will be the new V_{CH} for the next round C_{R+1} . However, if sensor nodes present in ζ are more than 1, then the current CH executes the CH election procedure using hesitant fuzzy entropy and selects the V_{CM} based on highest ($\alpha_{\sigma(i)} > D_{Fi}$) criteria. Subsequently, the current CH sends the CH_Election_Set message to the new CH. Upon receiving the CH_Election_Set message, the new CH needs to multicast the CH_announce message to its CMs in order to advertise its role as a potential CH. The pseudocode for CH task execution is given in Algorithm 4.

Algorithm 4. HFECs (CH Side)**Input:** CH election procedure executed by BS**Output:** CH decision and local data fusion decision using HFECs**Begin:**

```

1.                                     if (CH_Election_Set message received), then
2.                                         Start preparing CH_Announce message in order
3.                                         to advertise your role as potential CH
4.                                         Send CH_Announce message to neighboring
5.                                         sensor nodes  $S_j$  which are in working mode
6.                                     end if
7.                                     Wait till the Con_Req message is received from
8.                                     neighboring  $S_j$ 
9.                                     if (Con_Req message is received from neighboring
10.                                     $S_j$ ), then
11.                                        CH creates a cluster member set, i.e., CM_set
12.                                        Add the neighboring  $S_j$  in CM_set
13.                                    end if
14.                                    if (all working  $S_j$  in a cluster are part of CM_set of
15.                                    their new CH), then
16.                                        Each new CH transmits the TDMA schedule to
17.                                        all CMs keeping in view the
18.                                        working-sleeping cycle of CMs
19.                                    end if
20.                                    if (integrated data received from  $S_j | S_j \in \text{CM\_set}$ ),
21.                                    then
22.                                        CH performs aggregation on multiple data
23.                                        packets received from  $S_j$  by concatenating
24.                                        into a single file of specified length
25.                                        CH standardizes the data received from  $S_j$ ,
26.                                        calculates the hesitant fuzzy entropy and weights
27.                                        CH synthesizes the attribute values of sensor
28.                                        nodes to determine threshold attribute values
29.                                        CH compares the original attribute value with
30.                                        that of threshold attribute values
31.                                        if (original attribute values are greater than
32.                                        threshold attribute values after T), then
33.                                            CH periodically transmits the aggregated
34.                                            data to BS
35.                                        else if (original attribute values are less than
36.                                        threshold attribute values after T), then
37.                                            CH will not transmit the aggregated data to
38.                                            BS
39.                                        end if
40.                                    end if
41.                                    Calculate Link connectivity of CH with its cluster
42.                                    members using Equation (14)
43.                                    for each cluster member  $V_{CM(i)}$  in CM_set, do
44.                                        if ( $L_{V_{CH}V_{CM(j)}} \geq L_{V_{CM(i)}V_{CM(j)}}$ ), then
45.                                            Add  $V_{CH}$  to CH_Election_Procedure set  $\zeta$ 
46.                                        else if ( $L_{V_{CH}V_{CM(j)}} < L_{V_{CM(i)}V_{CM(j)}}$ ), then
47.                                            Add  $V_{CM(i)}$  to CH_Election_Procedure set  $\zeta$ 
48.                                        end if
49.                                    end for
50.                                    if ( $|\zeta| = 1$ ) and  $V_{CH} \in \zeta$ , then
51.                                         $V_{CH}$  will be new  $V_{CH}$  for next round  $C_{R+1}$ 
52.                                    else if ( $|\zeta| = 1$ ) and  $V_{CH} \notin \zeta$ 
53.                                         $V_{CM}$  will be new  $V_{CH}$  for next round  $C_{R+1}$ 
54.                                    else if ( $|\zeta| > 1$ )
55.                                         $V_{CH}$  executes the CH election procedure and
56.                                        selects new CH with highest ( $\alpha_{\sigma(i)} > D_{F_i}$ )
57.                                         $V_{CH}$  sends CH_Election_Set message to new CH
58.                                    end if

```

4.3. Packet Overhead Calculation

In our proposed scheme, each CH acquires the integrated sensed data from its CMs and then forwards the aggregated data to BS after detecting the change in integrated data for every time period T . In our paper, the hesitant fuzzy entropy balances the network load during CH rotation in successive rounds of communication when applied on MADM. Let us assume that we have $O(N_W)$ exchanged packets as an overhead in which N_W is the number of working sensor nodes in the network. If we assume that k is the number of CHs per round, then we can calculate the overhead packets as:

- k packets used for broadcasting CH_Announce by all CHs.
- $N_W - k$ packets in terms of Con_Req by working CMs.
- k packets used for broadcasting TDMA schedule to working CMs in CM_set by all CHs.
- $N_W - k$ packets used for sending data to CHs by working CMs.
- k packets used for forwarding data to the BS by all CHs in case if final prediction result is changed after time period T .
- $\frac{N}{N_C} - 1$ packets received by all CHs from their CMs for link connectivity calculation.
- $\frac{N}{N_C} - 1$ packets received by all CMs from their neighboring CMs and current CH for link connectivity calculation comparison.
- k packets used for the CHs rotation in case if all CHs select one of its CMs as the next CH.

Therefore, the overhead of HF ECS will be $O(N_W)$ as $k \ll \frac{N}{N_C} \ll N_W$.

5. Case Study: HF ECS in Neighborhood Area Networks

In this paper, we have described the opportunistic clustering mechanism based on hesitant fuzzy entropy using several parameters, such as the time-frequency parameter, residual energy, LQR, distance from the BS and Nsupport. Anees et al. in [26] proposed an energy-efficient multi-disjoint path opportunistic node connection routing protocol for smart grids in which the OCRG was utilized to calculate the link and path connectivity between sensor nodes for data forwarding purposes. Likewise, we have considered the neighborhood area networks (NANs) of smart grids in our case study to prove the significance of hesitant fuzzy entropy in heterogeneous clustering of WSNs. If we make an analogy between heterogeneous clustering and NANs of SGs, we can consider the NAN gateway as the CH and the several Home Area Network (HAN) gateways connected to the NAN gateway as CMs connected to their CH, as shown in Figure 2. Furthermore, the Access gateway in NANs can be considered as the BS, which receives the aggregated data from multiple NAN gateways and forwards it to the control center [46]. In this case study, we can see how the efficient use of hesitant fuzzy entropy can help us in making the CH election decision and also perform reliable data fusion.

We have selected the measured values of six attributes from five sensor nodes placed randomly in a cluster. It is important to mention here that for this case study we considered the network area as centralized NAN for SGs communication. As each of the attributes have different [min max] values, so we needed to perform the data standardization in order to bring all the attributes on the same scale and to convert our measured data values into hesitant fuzzy set. So, we divided the attribute values into 10 parts according to their [min max] values and then assign values corresponding to 0–1.0. Table 2 represents the interval for each attribute used for data standardization whereas Table 3 represents the measured value of attributes for all sensor nodes. Different monitoring values for each attribute indicate that data acquisition cycle for each sensor node is different due to the asynchronous working-sleeping cycle strategy. For example, for the attribute time-frequency parameter in Table 3, its measured values are normalized to $\{([0,10] \rightarrow [0,0.1]), ([10,20] \rightarrow [0.1,0.2]), \dots ([60,70] \rightarrow [0.6,0.7]), \dots ([90,100] \rightarrow [0.9,1])\}$. Similarly, for residual energy, the standardized attribute values will be $\{([0,0.25] \rightarrow [0,0.1]), ([0.25,0.5] \rightarrow [0.1,0.2]), \dots ([1.5,1.75] \rightarrow [0.6,0.7]), \dots ([2.25,2.5] \rightarrow [0.9,1])\}$. The procedure of data standardization is the same for all other attributes mentioned in Table 3, along with their intervals, but for attribute 'Distance to BS', the standardized values are $\{([150,136] \rightarrow [0,0.1]), ([136,122]$

→ [0.1,0.2]), ([66,52] → [0.6,0.5]) . . . ([24,10] → [0.9,1.0])} because if ‘Distance to BS’ is minimum, then we can have better connectivity with BS.

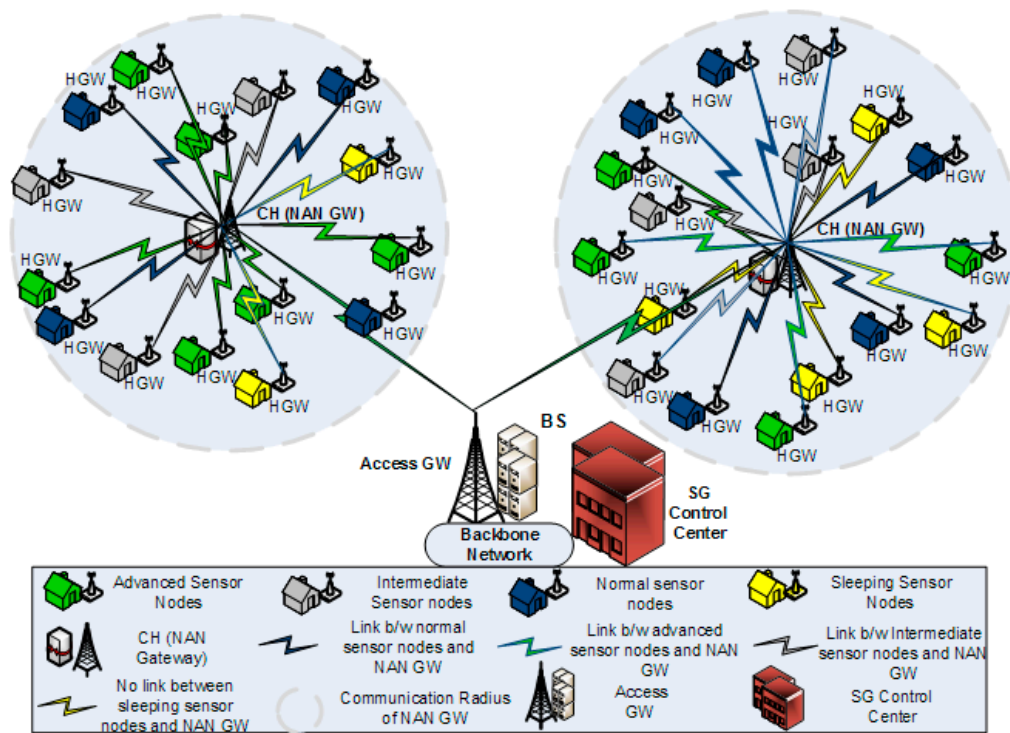


Figure 2. Hesitant fuzzy entropy (HFE) based opportunistic clustering for neighborhood area networks.

Table 2. Interval for each attribute used for data standardization.

	Time-Frequency Parameter	Residual Energy	Remaining Buffer Capacity	LQR	Distance to BS	Nsupport
Interval [min,max]	[0,100]	[0,2.5]	[0,100]	[10,15]	[150,10]	[0,40]

Table 3. Measured value after screening matrix.

	Time-Frequency Parameter	Residual Energy	Remaining Buffer Capacity	LQR	Distance to BS	Nsupport
S1	{70,65,60}	{2,1.8,1.6}	{78,75,73}	{11.5,11,10.5}	{110}	{30,27,25}
S2	{65,60,55}	{1.9,1.85,1.83}	{65,63,62,60}	{12}	{50,49}	{29,28,27}
S3	{65,60}	{1.9,1.7,1.5,1.3}	{90,87,84,82}	{12.5,11.9}	{45}	{29,26,24}
S4	{54,51,48,45}	{2.2,2,1.8}	{80,78,77}	{13.5,13,12.8}	{42,40}	{33,30}
S5	{67,63,59,55}	{2.1,1.9}	{95,92,91}	{14,13.8,13.5}	{39,38,36}	{31.5,29}

Table 4 shows the corresponding hesitant fuzzy sets after the data standardization process. We utilized the hesitant fuzzy sets to compute hesitant fuzzy entropy using Equation (12) to acquire the statistical measure of uncertainty in our decision analysis for the CH election procedure and data fusion by CH. Table 5 represents the hesitant fuzzy entropy of each attribute used in our case study. After generating the hesitant fuzzy entropy matrix, we determined the entropy weight coefficients using Equation (13). Table 6 represents the entropy weight coefficients of all sensor nodes, i.e., $W_i = [W_1, W_2, \dots, W_{r-1}, W_r]^T$

Table 4. Hesitant fuzzy decision matrix.

	Time-Frequency Parameter	Residual Energy	Remaining Buffer Capacity	LQR	Distance to BS	Nsupport
S1	{0.7,0.65,0.6}	{0.8,0.72,0.64}	{0.78,0.75,0.73}	{0.3,0.2,0.1}	{0.29}	{0.75,0.69,0.63}
S2	{0.65,0.6,0.55}	{0.76,0.74,0.73}	{0.65,0.63,0.62,0.6}	{0.4}	{0.7,0.71}	{0.74,0.7,0.68}
S3	{0.65,0.6}	{0.76,0.68,0.6,0.52}	{0.9,0.87,0.84,0.82}	{0.5,0.38}	{0.75}	{0.74,0.65,0.6}
S4	{0.54,0.51,0.48,0.45}	{0.88,0.8,0.72}	{0.8,0.78,0.77}	{0.7,0.6,0.56}	{0.76,0.78}	{0.82,0.75}
S5	{0.67,0.63,0.59,0.55}	{0.84,0.76}	{0.95,0.92,0.91}	{0.8,0.76,0.7}	{0.79,0.8,0.81}	{0.79,0.74}

Table 5. Entropy matrix.

	Time-Frequency Parameter	Residual Energy	Remaining Buffer Capacity	LQR	Distance to BS	Nsupport
S1	0.912	0.809	0.747	0.643	0.942	0.858
S2	0.961	0.767	0.918	0.987	0.890	0.832
S3	0.959	0.897	0.325	0.991	0.918	0.895
S4	0.999	0.644	0.683	0.943	0.808	0.786
S5	0.937	0.763	0.276	0.747	0.644	0.815

Table 6. Entropy coefficient weight values.

Ei	0.8185	0.8925	0.8308	0.8105	0.6970
Wi	0.1909	0.1131	0.1779	0.1993	0.3187

Moreover, the average values of all attributes in Table 2 and entropy weight coefficients in Table 6 were used to compute the threshold attribute values using Equation (14). As we have six attributes in total, so the six threshold attribute values in our case study are $D_{F_1} = 59.6252$, $D_{F_2} = 1.8748$, $D_{F_3} = 81.8583$, $D_{F_4} = 12.5952$, $D_{F_5} = 54.7839$, $D_{F_6} = 28.9909$. Consequently, we compare the measured attribute value of each sensor with that of threshold attribute value in Table 7. Here, we considered the attributes with the three highest entropy coefficient weights, i.e., $TF_{V_i V_j}$, D_{toBS} and $N_{support}$.

Table 7. Comparison of measured attribute values with threshold attribute values for CH election decision.

	$\alpha_{\sigma(1)} > D_{F1}$	$\alpha_{\sigma(5)} < D_{F5}$	$\alpha_{\sigma(6)} > D_{F6}$	CH Election Set	CH Election Decision
S1	Yes	No	Yes	No	S5 will be CH as it has highest attributes values
S2	Yes	Yes	Yes	Yes	
S3	Yes	Yes	Yes	Yes	
S4	No	Yes	Yes	No	
S5	Yes	Yes	Yes	Yes	

According to Table 7, only three sensors fulfill our criteria after comparing the measured attribute values with threshold attribute values. The BS puts the sensor nodes with highest ($\alpha_{\sigma(i)} > D_{Fi}$) in CH election set and then invokes those sensor nodes in the CH election set. Hence, in this case study, only S5 will be selected as the CH, but in reality, the whole process is much more complex than the one mentioned in this case study due to interference, overhead and energy constraints.

6. Performance Evaluation

6.1. Simulation Environment

In order to validate the effectiveness of HFECs, we evaluate the performance of our proposed scheme in MATLAB R2018a simulator using cross platform libraries such as MEX-API for simulating WSNs [47]. This Application Programming Interface (API) can provide the user with an easy bidirectional connection interface between MATLAB and OMNET. In our simulation environment,

150 sensor nodes were randomly placed in a variable network area, i.e., 100×100 , 150×150 , 200×200 , 250×250 , 300×300 and 350×350 m² on a 2-D plane with the BS as the network center. Here, we assumed that the BS has no energy constraints due to unlimited power supply but the randomly deployed sensors have limited battery power and we need asynchronous working-sleeping scheduling for their energy replenishment. The parameters used in this simulation are defined in Table 8.

Table 8. Simulation Table.

Parameters	Values
Network Area (M × M)	100 × 100, 150 × 150, 200 × 200, 250 × 250, 300 × 300, and 350 × 350
Number of sensor nodes (N)	150
E_n , E_{int} and E_{adv}	1.5 J, 2 J and 2.5 J
R_{normal} , R_{int} , R_{adv}	40%, 35%, 25%
MAC layer	IEEE 802.15.4
E_{elec}	50×10^{-9} J/bit
ϵ_{fs}	10×10^{-12} J/bit/m ²
ϵ_{mp}	0.0013×10^{-12} J/bit/m ⁴
Energy consumed during aggregation E_A	5×10^{-9} J/bit
Energy consumed by CH during data fusion (Data + Overhead) Packet size	5×10^{-12} J/bit 4096 bits

In our simulation, CHs send aggregated data to the BS after every $T = 100$ s and CMs send integrated sensed data to CH after every $t = 20$ s where $n = \frac{T}{t} = 5$. We calculated the total energy of our network in terms of R_{normal} , R_{int} , R_{adv} and E_n , E_{int} and E_{adv} as 288.11 J. We utilized the energy model defined for our MAC layer (IEEE 802.15.4) to compute the energy consumed during data transmission, reception and sensing. Additionally, we also utilized the IEEE 802.15.4 MAC Layer specifications for data rate and data packet size. It is pertinent to mention here that we considered various studies for acquiring the parameters like E_A , ϵ_{fs} , ϵ_{mp} and energy consumed by CHs during data fusion [19,39]. The performance of our proposed scheme is evaluated by comparing it with various routing protocols like SEP-E [34], DEEC-E [37], EBCS [39] and TEEN [38] based on the performance metrics like stability period, half-life time, average residual energy, network lifetime and PDR.

- Stability period—can be defined as the time difference between two specific points. The first point is when the simulation starts and the second point is when first sensor node dies.
- Half-life time—can be defined as the time duration from start of the network simulation until the time when 50% of the sensor nodes have no residual energy left to continue their data delivery tasks.
- Energy Consumption—can be defined as the total energy consumed in the network during data transmission, reception, sensing, status transitions ($CM \rightarrow CH$) and aggregation functions.
- Network Lifetime—this parameter is used to demonstrate the complete lifetime of sensor nodes per communication round for varying network size in terms of stacked group bar chart.
- Packet Delivery Ratio (PDR)—defined as the ratio of total number of successfully received data packets at BS corresponding to total number of data packets generated in the network by all sensor nodes.

6.2. Simulation Results

The performance metrics like the stability period, half-life time, average residual energy, network lifetime and PDR are analyzed against two parametric benchmarks viz. network size and number of communication rounds. We considered the number of communication rounds up to 4000 in our simulation.

6.2.1. Stability Period

Figure 3 depicts the stability period for SEP-E, DEEC-E, EBCS, TEEN and HFECs against different network sizes. It can be observed from Figure 3 that the stability periods of HFECs and EBCS outperform all other clustering schemes as both of them are entropy-based clustering schemes. Furthermore, the stability period of all clustering schemes is decreased with the increase in network size due to the reason that the distance between sensor nodes and the energy consumption is increased as we increase our network size. For most of the clustering schemes, energy is consumed for performing data forwarding and lossy aggregation tasks but in the case of HFECs, we perform lossless aggregation at CHs by concatenating multiple data packets into a single large data packet of specified length to reduce the energy consumption, thus resulting in an enhanced stability period in comparison to other clustering schemes for larger network sizes. As DEEC-E works on the basis of absolute threshold value calculated in each round, the residual energy of all sensor nodes is then compared with this threshold value. As this threshold value does not take into account the network size, the stability period of this scheme is significantly reduced when the network size increases up to $350 \times 350 \text{ m}^2$. The degraded performance of SEP-E in comparison to EBCS and HFECs can be explained as the probability of each type of sensor node do not consider the remaining energy while selecting the new CH. Also, when the network size increases, the low density regions in the network have a greater number of CHs, which results in the inefficient utilization of energy by the CMs and correspondingly the stability period of SEP-E is reduced. The performance of EBCS in terms of the stability period is better than SEP-E, DEEC-E, and TEEN for all the network sizes.

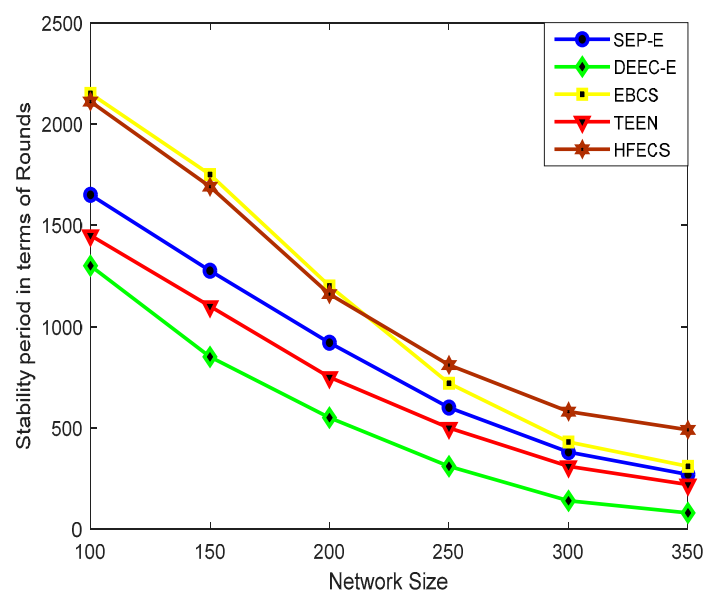


Figure 3. Stability period in terms of rounds against variable network size.

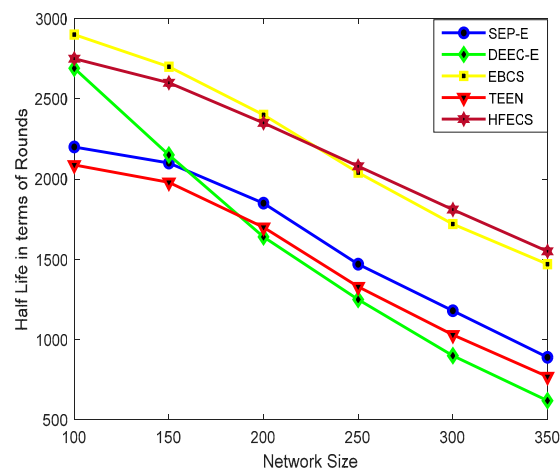
The performance of EBCS for network sizes up to $200 \times 200 \text{ m}^2$ is slightly better than for HFECs. The reason for this is that when sensor nodes supporting the asynchronous working–sleeping cycle are deployed in a smaller network area, we have a high density of sensor nodes in every cluster, i.e., greater number of cluster members to be facilitated by their assigned CH, so it is highly likely that a potential CH or a CM will lose all of its residual energy in a slightly less time than EBCS due to the extra computation of link connectivity calculations performed in every cluster. This extra computation is due to the increased number of sensor nodes present in every cluster. As we increase the network size, the uniformly random deployment of sensor nodes create clusters with a lower number of sensor nodes as compared to the scenario of smaller network sizes, so the probability of a sensor node in HFECs for losing all of its energy decreases in comparison to the probability of a sensor node operating in EBCS. Table 9 shows the percentage improvement in terms of the stability period.

Table 9. Percentage Improvement of hesitant fuzzy entropy-based opportunistic clustering and data fusion scheme (HFECs) in terms of stability period.

Schemes	Network Size					
	100 × 100	150 × 150	200 × 200	250 × 250	300 × 300	350 × 350
SEP-E	27.88	32.55	26.09	35	52.63	81.48
DEEC-E	62.31	98.82	110.91	161.29	314.29	512.5
EBCS	−1.86	−3.43	−3.33	12.5	34.88	58.06
TEEN	45.52	53.64	54.67	62	87.1	122.73

6.2.2. Half-Life Time

Figure 4 depicts the half-life time of SEP-E, DEEC-E, EBCS, TEEN, and HFECs against different network sizes. It can be seen from Figure 4 that HFECs and EBCS enhance the sensor nodes' half-life time in comparison to SEP-E, DEEC-E, and TEEN. Since the entropy weight coefficients are updated in every communication round, we see an improvement in half-life time and correspondingly better network lifetime. Moreover, as the number of sensor nodes which can be supported ($N_{support}$) by a CH implicitly reflects the cluster load which the CH can handle, so we included $N_{support}$ as one of our significant attributes in CH decision making process.

**Figure 4.** Half-life in terms of rounds against variable network size.

The performance of EBCS in terms of half-life time is better than SEP-E, DEEC-E, and TEEN for all the network sizes. In addition, EBCS achieves better half-life time as compared to HFECs for network sizes up to $200 \times 200 \text{ m}^2$ due to an increase in the cluster load and the extra computation required for calculating link connectivity between sensor nodes in the case of HFECs. For larger network sizes, HFECs performs better than EBCS due to the decrease in the cluster load, involvement of multiple attributes in decision making processes and sensor nodes with opportunistic connections, which finally results in a slight performance inversion between HFECs and EBCS. The percentage improvement of HFECs in terms of half-life time against different clustering schemes is given in Table 10.

Table 10. Percentage Improvement of HFECs in terms of half-life time.

Schemes	Network Size					
	100 × 100	150 × 150	200 × 200	250 × 250	300 × 300	350 × 350
SEP-E	25	23.81	27.03	41.5	53.39	74.16
DEEC-E	2.23	20.93	43.29	66.4	101.11	150
EBCS	−5.17	−3.7	−2.08	1.96	5.23	5.44
TEEN	31.7	31.31	38.24	56.39	75.73	101.3

6.2.3. Average Residual Energy

We computed the average residual energy in two ways, i.e., (i) against different network sizes while keeping the communication round as constant, (ii) against the number of communication rounds while keeping the network size as constant. Also, we have computed the average residual energy against different network sizes at 1000 rounds and 2000 rounds to check the performance of all clustering schemes. Figure 5 depicts the average residual energy of SEP-E, DEEC-E, EBCS, TEEN, and HFECs for different network sizes at 1000 rounds. It can be observed from Figure 5 that the performance behavior of all clustering schemes is analogous at different network sizes. Furthermore, HFECs outperforms all other clustering schemes including EBCS for larger network sizes. As we increase the network size, the average residual energy starts dropping down for all clustering schemes, which indicates the increase in energy consumption. A noticeable increase in energy consumption can be seen from SEP-E, EBCS, and TEEN but there is a remarkable increase in energy consumption for DEEC-E, due to higher negative slope in average residual energy. Likewise, Figure 6 depicts the average residual energy of SEP-E, DEEC-E, EBCS, TEEN and HFECs for different network sizes at 2000 rounds.

Figure 7 illustrates the average residual energy of SEP-E, DEEC-E, EBCS, TEEN and HFECs against number of communication rounds while keeping the network size as $300 \times 300 \text{ m}^2$. The performance behavior of all clustering schemes is analogous at different numbers of communication rounds. However, DEEC-E and HFECs perform slightly better than other clustering schemes after the 1500th round of communication. DEEC-E functionality depends on an absolute threshold value calculated in every round which do not involve network size, so the average residual energy of DEEC-E is reduced as we increase the communication rounds due to sensor nodes' poor lifetime and the increase in network size, but comparing the remaining energy of sensor nodes with that of an absolute threshold value gives advantage to sensor nodes with energy slightly above the threshold value, which results in better average residual energy than the rest of the clustering schemes. For smaller network sizes, the residual energy of the sensor nodes in HFECs is also consumed by link connectivity calculations for a higher number of sensor nodes per cluster, therefore the average residual energy of DEEC-E for smaller network sizes is better than in HFECs, but, as we increase the network size, the cluster load in HFECs reduces, thus resulting in better average residual energy for HFECs. Furthermore, as HFECs performs lossless data aggregation at the CH node for reducing the information flow and energy consumption, we have better average residual energy for higher values of communication rounds for HFECs than other clustering schemes. The percentage improvement of HFECs in terms of average residual energy is given in Table 11.

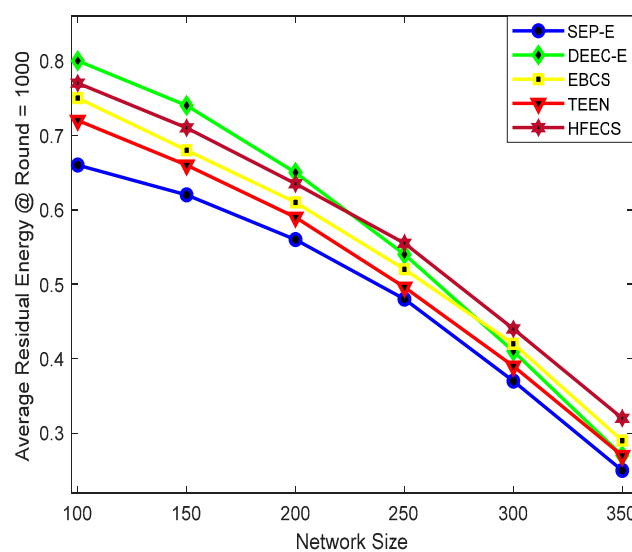


Figure 5. Average residual energy at round=1000 against variable network size.

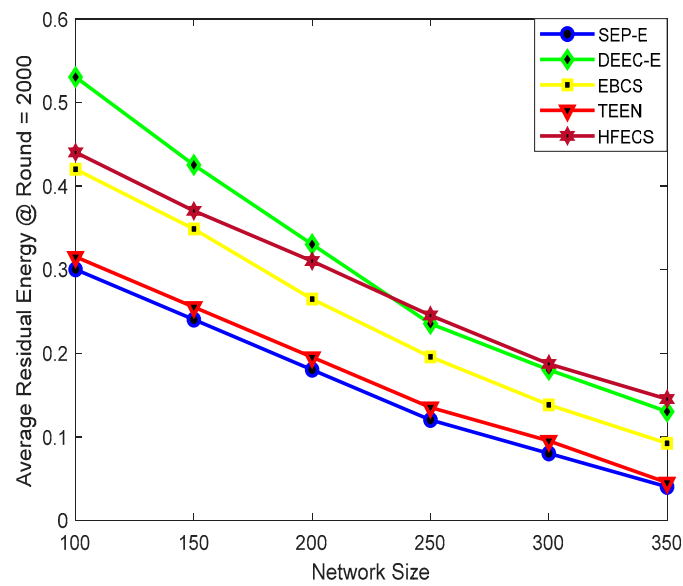


Figure 6. Average residual energy at round=2000 against variable network size.

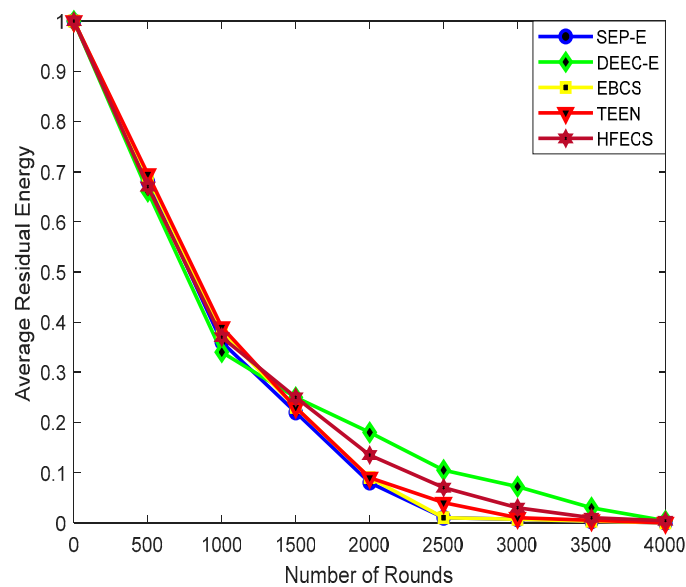


Figure 7. Residual energy against number of communication rounds.

Table 11. Percentage improvement of HFECs in terms of average residual energy.

Schemes	Number of Rounds						
	500	1000	1500	2000	2500	3000	3500
SEP-E	-1.47	2.78	13.64	68.75	600	328.57	233.33
DEEC-E	1.52	8.82	0	-25	-33.33	-58.33	-66.67
EBCS	-2.9	-2.63	8.7	50	600	328.57	233.33
TEEN	-3.6	-5.13	8.7	50	75	200	100

6.2.4. Network Lifetime

Network lifetime can be defined as the time interval between the start of network simulation and when the first sensor node dies, all sensor nodes die, some percentage of sensor nodes die, loss of coverage occurs or network is partitioned such that path from source to sink does not exist [26]. According to [26], network lifetime can also be evaluated in terms of the dead node ratio in the network. The time when 25%

or 50% of the nodes present in the network have no residual energy left to continue their data delivery tasks can be treated as the network lifetime. Here, we considered two such time intervals, i.e., (i) when the first sensor node dies (FND) and (ii) when half the nodes in the network die, i.e., 50% of the sensor nodes have no residual energy left to continue their data sensing tasks (HND). The stacked grouped bar chart in Figure 8 demonstrates the network lifetime in terms of two different cases, i.e., First Node Dead (FND) and Half Nodes Dead (HND) for SEP-E, DEEC-E, EBCS, TEEN and HFECs against different network sizes. In every stack, we have five bars representing five different schemes, i.e., SEP-E, DEEC-E, EBCS, TEEN and HFECs. In each bar, we have two groups, i.e., FND and HND.

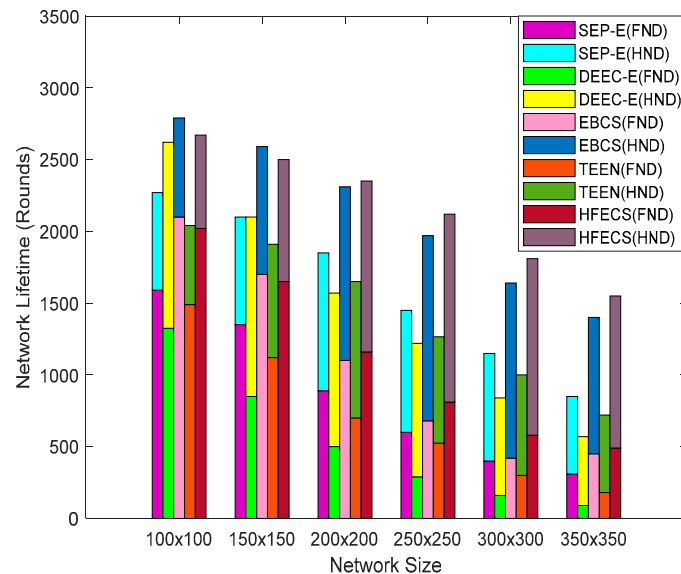


Figure 8. Network lifetime in terms of first node dead (FND) and half nodes dead (HND) against variable network size.

It can be observed from Figure 8 that the performance behavior of all clustering schemes is analogous for network lifetime, i.e., a decreasing trend as we increase the network size. For smaller network sizes, the network lifetime is higher for all clustering schemes as the nodes located in close vicinity are easy to communicate with but, for larger network sizes, the network lifetime decreases rapidly due to the fact that sensor nodes are not located in the close vicinity and extra energy is being consumed for communication between sensor nodes, which results in decreased network lifetime. The performance of EBCS for network sizes of up to $200 \times 200 \text{ m}^2$ is slightly better than HFECs. For smaller network sizes, we have a high density of sensor nodes in every cluster, so it is highly likely that a potential CH or a CM lose all of their residual energy in slightly less time than EBCS due to the extra computation of link connectivity calculations performed in every cluster. However, as we increase the network size, the uniformly random deployment of sensor nodes creates clusters with a lower number of sensor nodes as compared to the scenario of smaller network sizes, thus resulting in better network lifetime in comparison to EBCS. HFECs also adopts lossless data aggregation in multiple data packets at the CH level, which reduces the information flow from CHs to the BS, thus decreasing some of the extra energy spent on control packet overhead. The percentage improvement of HFECs in terms of network lifetime (FND and HND) is given in Tables 12 and 13.

Table 12. Percentage improvement of HFECs in terms of network lifetime (FND).

Schemes	Network Size					
	100 × 100	150 × 150	200 × 200	250 × 250	300 × 300	350 × 350
SEP-E	18.87	17.78	24.72	35	45	58.06
DEEC-E	42.64	57.06	76	79.31	88	120
EBCS	−5.5	−6.47	0.91	22.73	38.1	8.89
TEEN	26.85	41.96	58.57	54.29	93.33	99.22

Table 13. Percentage improvement of HFECs in terms of network lifetime (HND).

Schemes	Network Size					
	100 × 100	150 × 150	200 × 200	250 × 250	300 × 300	350 × 350
SEP-E	11.39	8.64	15.38	34.19	44.8	58.95
DEEC-E	−2.94	8.64	34.73	57.58	92.55	96.1
EBCS	−1.86	−7.72	−2.26	6.67	10.37	7.86
TEEN	19.46	18.91	28.57	52.38	64.55	77.65

6.2.5. Packet Delivery Ratio (PDR)

The higher percentage of successfully received packets at the BS reflects the higher network reliability. In the case of HFECs, multiple data packets received by every CH node are aggregated into a single packet of specified length and then sent to the BS, so we have to incorporate aggregated data packets in PDR calculation instead of normal data packets. The percentage of successfully received aggregated data packets at the BS corresponding to total number of packets generated by all sensor nodes will be the PDR, in the case of HFECs, i.e.,

$$PDR (Aggregated) = \left(\frac{\sum_{i=1}^n ADP_i}{Total\ Data\ packets\ generated} \right) * 100 \quad (15)$$

where n is the total number of aggregated data packets received at the BS and ADP_i denotes the aggregated data packet received by BS at the i^{th} instance in Equation (15). Figure 9 shows the PDR for SEP-E, DEEC-E, EBCS, TEEN and HFECs corresponding to different rounds of communication with our network size as $300 \times 300 \text{ m}^2$. As in the SEP-E protocol, the optimum number of CHs is not guaranteed due to its poor stability and every CH in SEP-E depends on the probability of each type of sensor node without considering the residual energy of sensor nodes, so it results in the reduction in average residual energy and network lifetime, thus leading to degraded PDR in comparison to HFECs, DEEC-E, and EBCS. The TEEN protocol performs worst against SEP-E, DEEC-E, EBCS and HFECs due to the low average residual energy and lower network lifetime with increasing communication rounds. For lesser communication rounds, HFECs shows degraded performance in comparison to EBCS due to the extra energy spent on computation purposes which result in status transitions of a few sensor nodes, thus reducing the PDR. However, for higher communication rounds, our proposed protocol, HFECs, outperforms all other clustering schemes due to its higher network lifetime, higher average residual energy for larger network sizes, data aggregation performed at the CH, and collision avoidance achieved through the assignment of TDMA slots to every CM, thus resulting in the increase in success rate of aggregated data packets at BS. The percentage improvement of HFECs in terms of PDR can be seen in Table 14.

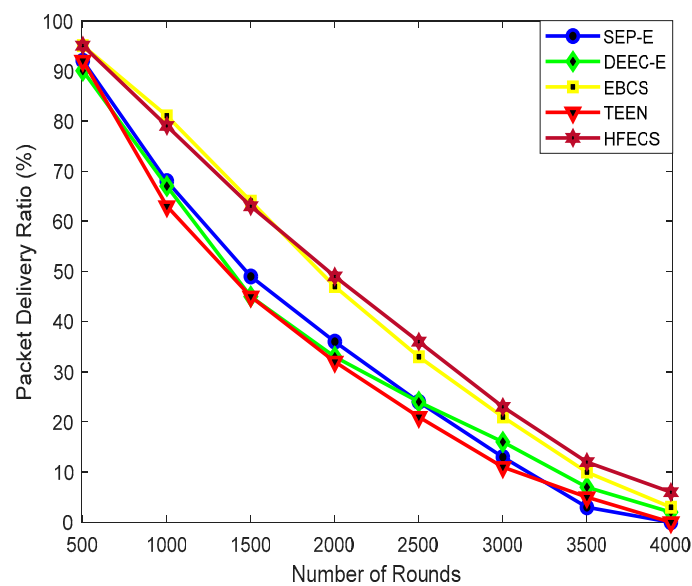


Figure 9. Packet delivery ratio (PDR) (%) against number of communication rounds.

Table 14. Percentage improvement of HFECs in terms of PDR.

Schemes	Number of Rounds						
	500	1000	1500	2000	2500	3000	3500
SEP-E	3.26	16.18	28.57	36.11	50	66.92	97
DEEC-E	5.56	17.91	40	48.48	50	43.75	71.43
EBCS	0	−2.47	−1.56	4.26	9.09	9.52	20
TEEN	3.26	25.4	40	53.13	61.43	87.09	99

7. Discussion

We have introduced the hesitant fuzzy entropy-based opportunistic clustering and data fusion scheme for WSNs. The previous entropy-based clustering schemes were designed without consideration of multiple attributes (especially asynchronous working-sleeping cycle) involved in the data flow from sensor nodes to CHs and to the BS. HFECs utilizes multiple attributes like the time frequency parameter, residual energy, remaining buffer capacity, link quality factor, the distance to the BS and the number of supported sensor nodes etc. in terms of local information of sensor nodes to measure hesitant fuzzy entropy and its corresponding weight coefficients for the CH election procedure and reliable data fusion, which improves the energy utilization and also increases the network lifetime of sensor nodes. The proposed designed of HFECs could be applicable to the following different approaches in future:

- Designing a HFE-based multipath opportunistic routing protocol for wireless cognitive sensor networks.
- Designing a HFE-based clustering scheme with ring routing protocol.
- Unmanned Aerial Vehicle (UAV) sensor networks
- Internet-of-Things (IOT)-enabled home energy systems
- Cyber-physical systems
- Energy harvesting wireless sensor networks [48].
- Hesitant fuzzy entropy analysis in a density-based and grid-based opportunistic clustering scheme.

8. Conclusions

In this paper, a novel hesitant fuzzy entropy-based opportunistic clustering and data fusion scheme is proposed, utilizing the OCRG theory. Our proposed scheme, HFECs, achieves two levels of hierarchy in the network and energy heterogeneity is characterized using three levels of energy in sensor nodes. Self-organizing HFECs uses multiple attributes in terms of the local information of sensor nodes to measure hesitant fuzzy entropy and its corresponding weight coefficients for CH election procedure and reliable data fusion at CH to reduce the redundant information flow from CH to BS, which in turn improves the energy utilization and increases the network lifetime of sensor nodes. The performance of HFECs is evaluated against existing energy-efficient and entropy-based heterogeneous clustering schemes for parameters such as half-life period, stability period, average residual energy, network lifetime (FND and HND), and PDR. Our simulation results clearly show that HFECs performs better than the existing benchmarks for larger network sizes. Possible future work could include HFE-based ring routing in WSNs, HFE-based heterogeneous clustering supporting sink mobility, HFE based on Cognitive Radio Sensor Networks (CRSN) etc. In addition to this, the proposed scheme can be applied to ring routing [49], big data systems [50], data compression [51] and blockchain technology [52] in terms of reliable data fusion.

Author Contributions: Conceptualization, J.A. and T.G.R.B.; Methodology, J.A., T.G.R.B. and B.G.L.; Formal analysis, J.A. and S.B.; Investigation, J.A. and S.B.; Resources, H.-C.Z.; supervision, H.-C.Z. and B.G.L.; project administration, H.-C.Z. and B.G.L.; software, J.A. and T.G.R.B.; validation, H.-C.Z. and B.G.L.; writing—original draft, J.A., T.G.R.B.; writing—review and editing, J.A., H.-C.Z. and B.G.L. All authors have read and agreed to the published version of the manuscript.

Funding: The research was funded by [National Natural Science Foundation of China] grant number [51776050] and grant number [51536001].

Acknowledgments: The authors would like to appreciate all reviewers for their insightful comments and deliberate suggestions for improving the quality of this paper.

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

References

1. Manshahia, M.S. Wireless Sensor Networks: A Survey. *IJSER* **2016**, *74*, 710–716.
2. Fadel, E.; Gungor, V.C.; Nassef, L.; Akkari, N.; Malik, M.G.A.; Almasri, S.; Akyildiz, I.F. A Survey on Wireless Sensor Networks for Smart Grid. *Comput. Commun.* **2015**, *71*, 22–33. [[CrossRef](#)]
3. Yick, J.; Mukherjee, B.; Ghosal, D. Wireless sensor network survey. *Int. J. Comput. Telecommun. Netw.* **2008**, *52*, 2292–2330. [[CrossRef](#)]
4. Kumar, D.; Aseri, T.C.; Patel, R.B. EEHC: Energy efficient heterogeneous clustered scheme for wireless sensor networks. *Comput. Commun.* **2009**, *32*, 662–667. [[CrossRef](#)]
5. Chatterjee, M.; Das, S.K.; Turgut, D. WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks. *Clust. Comput.* **2002**, *5*, 193–204. [[CrossRef](#)]
6. Karl, H.; Willig, A. *Protocols and Architectures for Wireless Sensor Networks*; John Wiley and Sons: Chichester, UK, October 2007.
7. Alfayez, F.; Hammoudeh, M.; Abuarqoub, A. A Survey on MAC Protocols for Duty-Cycled Wireless Sensor Networks. *Procedia Comput. Sci.* **2015**, *73*, 482–489. [[CrossRef](#)]
8. Ghadimi, E.; Landsiedel, O.; Soldati, P.; Duquenois, S.; Johansson, M. Opportunistic Routing in Low Duty Cycle Wireless Sensor Networks. *ACM Trans. Sens. Netw.* **2014**, *10*, 1–39. [[CrossRef](#)]
9. Lai, Y.; Lin, Z. Data Gathering in Opportunistic Wireless Sensor Networks. *Int. J. Distrib. Sens. Netw.* **2012**, *2012*, 1319–1322. [[CrossRef](#)]
10. Luo, J.; Hu, J.; Wu, D.; Li, R. Opportunistic Routing Algorithm for Relay Node Selection in Wireless Sensor Networks. *IEEE Trans. Ind. Inform.* **2015**, *11*, 112–121. [[CrossRef](#)]
11. Morris, R.; Morris, R. ExOR: Opportunistic multi-hop routing for wireless networks. In Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Philadelphia, PA, USA, 22–26 August 2005.

12. Boukerche, A.; Darehshoorzadeh, A. Opportunistic Routing in Wireless Networks: Models, Algorithms, and Classifications. *ACM Comput. Surv.* **2014**, *47*, 1–36. [[CrossRef](#)]
13. Nguyen Thi Thanh, N.; Nguyen Kim, K.; Ngo Hong, S.; Ngo Lam, T. Entropy Correlation and Its Impacts on Data Aggregation in a Wireless Sensor Network. *Sensors* **2018**, *18*, 3118. [[CrossRef](#)] [[PubMed](#)]
14. Wang, Q.; Yuan, X.; Zhang, J.; Gao, Y.; Hong, J.; Zuo, J.; Liu, W. Assessment of the Sustainable Development Capacity with the Entropy Weight Coefficient Method. *Sustainability* **2015**, *7*, 13542–13563. [[CrossRef](#)]
15. Tian, J.; Liu, T.; Jiao, H. Entropy Weight Coefficient Method for Evaluating Intrusion Detection Systems. In Proceedings of the International Symposium on Electronic Commerce and Security, Guangzhou, China, 3–5 August 2008; pp. 592–598.
16. Qiang, N.; Qiannan, X. Weight Optimization Method of Wireless Sensor Network Based on Fuzzy MADMR. In Proceedings of the Fourth International Conference on Intelligent Computation Technology and Automation, Shenzhen, China, 28–29 March 2011; pp. 303–306.
17. Bhunia, S.S.; Das, B.; Mukherjee, N. EMCR: Routing in WSN Using Multi Criteria Decision Analysis and Entropy Weights. In Proceedings of the 7th International Conference on Internet and Distributed Computing Systems (IDCS), Calabria, Italy, 22–24 September 2014; pp. 325–334.
18. Hengqiang, S.; Helong, Y. Application of entropy weight coefficient method in environmental assessment of soil. In Proceedings of the World Automation Congress, Puerto Vallarta, Mexico, 24–28 June 2012; pp. 1–4.
19. Wang, J.; Tawose, O.T.; Jiang, L.; Zhao, D. A New Data Fusion Algorithm for Wireless Sensor Networks Inspired by Hesitant Fuzzy Entropy. *Sensors* **2019**, *19*, 784. [[CrossRef](#)] [[PubMed](#)]
20. Chen, N.; Xu, Z.; Xia, M. Correlation coefficients of hesitant fuzzy sets and their applications to clustering analysis. *Appl. Math. Modell.* **2013**, *37*, 2197–2211. [[CrossRef](#)]
21. Ogundile, O.; Alfa, A. A Survey on an Energy-Efficient and Energy-Balanced Routing Protocol for Wireless Sensor Networks. *Sensors* **2017**, *17*, 1084. [[CrossRef](#)]
22. Yang, G.; Peng, Z.; He, X. Data Collection Based on Opportunistic Node Connections in Wireless Sensor Networks. *Sensors* **2018**, *18*, 3697. [[CrossRef](#)]
23. Ng, K.P.; Tsimeridis, C.; Woo, W.L. C-Sync: Counter-based synchronization for duty-cycled wireless sensor networks. *Ad. Hoc. Netw.* **2017**, *61*, 51–64. [[CrossRef](#)]
24. Mukherjee, M.; Shu, L.; Hu, L.; Hancke, G.P.; Zhu, C. Sleep Scheduling in Industrial Wireless Sensor Networks for Toxic Gas Monitoring. *IEEE Wirel. Commun.* **2017**, *99*, 2–8. [[CrossRef](#)]
25. Norman, J.; Joseph, P. A New Graph Model for Heterogeneous WSN. *Commun. Comput. Inf. Sci.* **2011**, *197*, 187–196.
26. Anees, J.; Zhang, H.-C.; Baig, S.; Lougou, B.G. Energy-Efficient Multi-Disjoint Path Opportunistic Node Connection Routing Protocol in Wireless Sensor Networks for Smart Grids. *Sensors* **2019**, *19*, 3789. [[CrossRef](#)]
27. Liang, H.; Yang, S.; Li, L.; Gao, J. Research on routing optimization of WSNs based on improved LEACH protocol. *EURASIP J. Wirel. Commun. Netw.* **2019**, 194. [[CrossRef](#)]
28. Handy, M.J.; Haase, M.; Timmermann, D. Low energy adaptive clustering hierarchy with deterministic cluster-head selection. In Proceedings of the 4th International Workshop on Mobile and Wireless Communications Network, Stockholm, Sweden, 9–11 September 2002; pp. 368–372.
29. Khediri, S.E.; Nasri, N.; Wei, A.; Kachouri, A. A new approach for clustering in wireless sensors networks based on LEACH. *Procedia Comput. Sci.* **2014**, *32*, 1180–1185. [[CrossRef](#)]
30. Aderohunmu, F.A.; Deng, J.D.; Purvis, M.K. A Deterministic Energy efficient Clustering protocol for wireless sensor networks. In Proceedings of the Seventh IEEE International Conference on Intelligent Sensors, Sensor Networks and Information Processing (IEEE-ISSNIP), Adelaide, Australia, 6–9 December 2011; pp. 341–346.
31. Smaragdakis, G.; Matta, I.; Bestavros, A. SEP: A Stable Election Protocol for clustered heterogeneous wireless sensor networks. In Proceedings of the International Workshop on SANPA, Boston, MA, USA, 22 August 2004.
32. Salim, A.; Osamy, W.; Khedr, A.M. IBLEACH: Effective LEACH Protocol for Wireless Sensor Networks. *Wirel. Netw.* **2014**, *20*, 1515–1525. [[CrossRef](#)]
33. Sharma, S.; Bansal, R.K.; Bansal, S. Heterogeneity-aware Energy efficient Clustering (HEC) Technique for WSNs. *KSII Transactions on Internet and Information Systems* **2017**, *11*, 1866–1888. [[CrossRef](#)]
34. Aderohunmu, F.A.; Deng, J.D. *An Enhanced Stable Election Protocol (SEP) for Clustered Heterogeneous WSN*; Department of Information Science, University of Otago: Dunedin, New Zealand, 2011.
35. Qing, L.; Zhu, Q.; Wang, M. Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks. *Comput. Commun.* **2006**, *29*, 2230–2237. [[CrossRef](#)]

36. Saini, P.; Sharma, A.K. E-DEEC-Enhanced Distributed Energy Efficient Clustering scheme for heterogeneous WSN. In Proceedings of the First International Conference On Parallel, Distributed and Grid Computing (PDGC), Solan, India, 28–30 October 2010; pp. 205–210.
37. Javaid, N.; Qureshi, T.N.; Khan, A.H.; Iqbal, A.; Akhtar, E.; Ishfaq, M. EDDEEC: Enhanced Developed Distributed Energy-efficient Clustering for Heterogeneous Wireless Sensor Networks. *Procedia Comput. Sci.* **2013**, *19*, 914–919. [[CrossRef](#)]
38. Manjeshwar, A.; Agrawal, D.P. TEEN: A routing protocol for enhanced efficiency in wireless sensor networks. In Proceedings of the 15th International Parallel and Distributed Processing Symposium, San Francisco, CA, USA, 23–27 April 2001. [[CrossRef](#)]
39. Osamy, W.; Khedr, A.M.; Salim, A. An Information Entropy Based-Clustering Algorithm for Heterogeneous Wireless Sensor Networks. *IEEE Access* **2018**. [[CrossRef](#)]
40. Xia, M.; Xia, Z. Hesitant fuzzy information aggregation in decision making. *Int. J. Approx. Reason.* **2011**, *52*, 395–407. [[CrossRef](#)]
41. Xia, M.; Xia, Z. Hesitant Fuzzy Entropy and Cross-Entropy and Their Use in Multi attribute Decision-Making. *Int. J. Intell. Syst.* **2012**, *27*, 799–822.
42. Su, W.; Bougiouklis, T.C. Data Fusion Algorithms in Cluster-based Wireless Sensor Networks Using Fuzzy Logic Theory. In Proceedings of the 11th WSEAS International Conference on COMMUNICATIONS, Agios Nikolaos, Crete Island, Greece, 26–28 July 2007; pp. 291–299.
43. Izadi, D.; Abawajy, J.H.; Ghanavati, S.; Herawan, T. A data fusion method in wireless sensor networks. *Sensors* **2015**, *15*, 2964–2979. [[CrossRef](#)]
44. Chaurasia, S.; Chaurasia, R.; Kumar, B.; Sharma, R.K. Quality Estimation based Multi-Sensors Data Fusion in Wireless Sensor Network: Review. *Int. J. Adv. Res. Comput. Commun. Eng.* **2017**, *6*, 124–129.
45. Zhai, W. Performance Evaluation of Wireless Sensor Networks Based on Hesitant Fuzzy Linguistic Preference Relations. *Int. J. Online Biomed. Eng.* **2018**, *14*, 233–240. [[CrossRef](#)]
46. Erol-Kantarci, M.; Mouftah, H.T. Energy-Efficient Information and Communication Infrastructures in the Smart Grid: A Survey on Interactions and Open Issues. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 179–197. [[CrossRef](#)]
47. Matlab. *R2018b Simulator*; MathWorks: Natick, MA, USA, 2018.
48. Adu-Manu, K.S.; Adam, N.; Tapparelo, C.; Ayatollahi, H.; Heinzelman, W. Energy-Harvesting Wireless Sensor Networks (EH-WSNs). *ACM Trans. Sens. Netw.* **2018**, *14*, 1–50. [[CrossRef](#)]
49. Maurya, S.; Jain, V.K.; Chowdhury, D.R. Delay aware energy efficient reliable routing for data transmission in heterogeneous mobile sink wireless sensor network. *J. Netw. Comput. Appl.* **2019**, *144*, 118–137. [[CrossRef](#)]
50. Mehta, P.; Dorkenwald, S.; Zhao, D.; Kaftan, T.; Cheung, A.; Balazinska, M.; Rokem, A.; Connolly, A.; Vanderplas, J.; AlSayyad, Y. Comparative Evaluation of Big-Data Systems on Scientific Image Analytics Workloads. In Proceedings of the 43rd International Conference on Very Large Data Bases (VLDB), Munich, Germany, 28 August–1 September 2017; pp. 1226–1237.
51. Zhao, D.; Qiao, K.; Yin, J.; Raicu, I. Dynamic Virtual Chunks: On Supporting Efficient Accesses to Compressed Scientific Data. *IEEE Trans. Serv. Comput.* **2016**, *9*, 96–109. [[CrossRef](#)]
52. Al-Mamun, A.; Li, T.; Sadoghi, M.; Zhao, D. In-memory Blockchain: Toward Efficient and Trustworthy Data Provenance for HPC Systems. In Proceedings of the IEEE International Conference on Big Data, Seattle, WA, USA, 10–13 December 2018; pp. 3808–3813.

