

Energy Harvesting Aware Clustering and Opportunistic Transmission with Fuzzy Petri Net Reasoning

Aya Mostafa^{1,*}, Khaled Hassan^{1,†}

Faculty of Information Engineering and Technology (IET)
German University in Cairo, New Cairo, Egypt

Abstract

As a solution to the environmental and natural disaster monitoring and detection, we propose an energy efficient large-scale autonomous Wireless Sensor Networks (WSNs) that can withstand the limited battery lifetime problem. Hereto, an energy-harvesting (EH) aware clustering technique is introduced using a knowledge-based inference approach for selecting cluster heads using fuzzy petri nets with three level hierarchy. Our EH mechanism endorses the sensor nodes with hybrid sources EH giving a potential boost for the network life. The robustness of our WSN is addressed by introducing a primary and a secondary backups for each level to reduce the frequency of reclustering. Finally, an opportunistic energy harvesting aware transmission technique is presented to enhance the network life-time. The performance of this scheme is evaluated against the famous clustering techniques, where the proposed algorithms extend the WSN field life-time significantly.

Received on 19 January 2015; accepted on 21 February 2015; published on 17 September 2015

Keywords: energy harvesting, fuzzy petri nets reasoning, wireless sensor networks, clustering algorithms.

Copyright © 2015 K. Hassan and A. Mostafa, licensed to EAI. This is an open access article distributed under the terms of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi:10.4108/eai.17-9-2015.150283

1. Introduction

Large scale wireless sensor networks (WSN) are quickly gaining popularity due to the fact that they are potentially low cost solutions to a variety of real-world challenging applications. Generally WSN is a group of nodes deployed to act as a cooperative network. These nodes are small, low cost, low-powered, energy-constrained sensor nodes with sensing, limited data processing, and wireless communication components. Being powered by chemical batteries, traditional sensors have only access to limited energy resource. Hence, the design of energy efficient sensor network is a crucial issue. Various solutions have been proposed for prolonging the lifetime of the sensor nodes through minimizing the energy consumption. Most of the solutions are based on utilizing clustering algorithms and enhance the routing protocol efficiency [27]. This

paper extends the work done in [17] to serve these purposes.

The WSN clustering is the process that aims at dividing the nodes into groups, where each group agrees on a central node nominating it to be a Cluster Head (CH). Such a CH is responsible for gathering the sensed measurements from all group members, aggregating them, compressing them if possible, and forwarding them to a Fusion Center (FC) [7] in the WSN field. In literature, the existing clustering mechanisms are classified into several schemes, the most basic scheme is the heuristic scheme as Linked Cluster Algorithm (LCA), which is considered one of the oldest algorithms developed [5]. The node having the highest ID among its neighbors is elected a CH. Another well known category of these schemes is the hierarchical clustering scheme [3]. One of the most famous hierarchical algorithms is the Low Energy Adaptive Clustering Hierarchy (LEACH) [10], which relies on randomly selected, periodically rotated CHs. However, LEACH suffer from several disadvantages as some clusters may

*Aya Mostafa. Email: aya.mostafa-ahmed@vodafone.com

†Khaled Hasan Email: khaled.shawky@guc.edu.eg

have very high number of Sensor Nodes (SNs) compared to others. This is as LEACH assumes homogeneous distribution of the sensor nodes, which is not realistic. In addition, CHs are randomly selected; hence, it is possible that two or more CHs are very close to each other leading to non uniform distribution of CHs. Another hierarchical algorithm is Hybrid Energy-Efficient Distributed clustering (HEED) [25]. It focuses on efficient clustering by proper selection of CH based on residual energy and the degree of the nodes. However, HEED did not address the problem of CH death. Moreover, it is a two level hierarchy where the CH communicate directly with the FC, leading to a higher error rate. In general, these centralized algorithms has proven to have low performance in large-scale networks [26].

Fuzzy logic clustering algorithms was introduced for selecting CHs with human-like decisions. The reason is that the fuzzy logic does not tend to model the system with a mathematical exact. Thereby, fuzzy logic is considered as a flexible method for clustering due to the allowance of partial set membership rather than crisp solutions. Such an advantage makes fuzzy logic a very effective methodology for solving problems, even when accurate data is not available [14]. Cluster-head election using fuzzy logic is proposed in [8], where electing a CH is based on three descriptors: energy, neighboring nodes concentration, and node centrality. Moreover, in [13], the Cluster Head Election mechanism using Fuzzy logic (CHEF) is proposed, which is a localized cluster head election mechanism. CHEF depends on centrality and residual energy of a node in the election process. Almost at the same time, the authors of [23] presented their Fuzzy Self-Clustering Algorithm (FSCA) to improve the WSN lifetime considering the same parameters as in CHEF. Additionally, Liang et al. presented in [15] an Energy and Mobility-aware Geographical Multipath Routing (EM-GMR) algorithm. It is based on fuzzy logic system considering the remaining battery capacity, mobility, and distance to the destination node. Furthermore, LEACH was extended in [6] to include fuzzy logic input variables which are the number of neighbors, centrality, energy remaining, number of received signals, and rounds where the nodes are not elected as CHs (similar to LEACH). More recently, Yu et al. in [26] proposes a Reliable Energy-Efficient Multi-Level Routing algorithm (REEMR) to use fuzzy petri nets to select cluster heads. In this case, a fuzzy reasoning mechanism is used to construct a route from the CHs to the FC.

In parallel to the developments in network clustering techniques, another important recent progress for prolonging the WSN lifetime is the utilization of Energy Harvesting (EH) to power wireless sensors [21, 28]. Hence, the Wireless sensors can be equipped with energy harvesting devices, such as the piezo electric,

and photo-voltaic cells. Those devices can be used to gain access to additional energy resource harvested from the ambient environment. The goal of energy harvesting is to reach the energy neutral state, where the amount of energy consumed is no more than the amount of energy harvested in a certain period of time [12, 19]. In this contribution, a combination of fuzzy petri nets and energy harvesting concepts is utilized in a novel clustering technique to dramatically increase the life time of WSN. Moreover, it introduces continuously harvesting backup nodes that take over the existing CHs when they are dead. An Energy Harvesting aware clustering with Fuzzy Petri net algorithm (EHFP) is presented fulfilling these concepts. Our proposed EHFP is a localized clustering algorithm that aims at acquiring long life, robust, and reliable network. EHFP attains a long life network because uses fuzzy petri nets reasoning to choose the CH based on the harvesting capability, the concentration of nodes, and the centrality of each node. The proposed algorithm achieves robustness because it introduces the concept of having multi backups for each CH reducing the load and energy depleted of having the whole cluster to be off-line just because one CH is dead. EHFP attains a reliable network because it uses inter-cluster communication to aggregate the data and send it to the sink in a three level network hierarchy. Cluster masters are chosen in the same criterion of choosing a CH collecting the data from the CH following them and sending it to the sink, reducing the overall distance that the CH covers to send directly to the sink. Hence, decreasing the power consumed and the error rate.

This paper is an extension to our contribution in [17] proposing a modification to the existing REEMR and the conventional HEED algorithms. Moreover, we include more sophisticated energy harvesting techniques and accessing procedures to the simple algorithm presented in [17]. Finally we concluded up the paper with a complete performance evaluation comparing all the algorithms.

The rest of the paper is organized as follows. In Section 2, a brief background on fuzzy Petri nets is explained. Moreover, in Section 3, some techniques were introduced to enhance the lifetime and data rate of the network. In Section 4, the harvesting model is described. Afterwards, in Section 5, EHFP algorithm is introduced. The simulations of the presented algorithm and a comparison of EHFP with the HEED algorithm is presented in Section 6. Finally, Section 7 concludes the paper.

2. Fuzzy Petri Nets: State of the Art

Chen et al. in [4] proposed the Fuzzy Petri Net (FPN) concept. FPN combines the concept of petri nets and fuzzy logic [20]. It is used to model the fuzzy

production rules of a rule based system. This concept allowed the organized representation of data with systematic procedure for supporting fuzzy reasoning. FPN introduced modification to the fuzzy logic concept and the fuzzy logic terms.

Fuzzy production rules, i.e. fuzzy IF-THEN rules, identify the relation between two propositions (fuzzy variables). Assume R to be a set of production rules, where $R = \{R_1, R_2, \dots, R_n\}$, the i^{th} production rule is represented as

$$R_i : \text{IF } d_j \text{ THEN } d_k (CF = \mu_i) (i = 1, 2, \dots, n), \quad (1)$$

where d_j and d_k are both propositions containing fuzzy variables of different levels (*high*, *low*, and *medium*). Each proposition has a truth degree between zero and one (i.e. fuzzified value). Meanwhile, μ is the Certainty Factor (CF) of this proposition, $\mu \in [0, 1]$, which gives an insight about the strength of certainty of each rule. For instance, the production rule R_1 is

R_1 : If the residual energy of a node is high then its chance to be a CH is large (CF=0.9).

Hence, the certainty of this rule is 0.9. Moreover, let $\lambda \in [0, 1]$ be the threshold value of firing the transition t_1 from place p_1 to another place p_2 as in Fig. 1a. Also, let the degree of truth of proposition of d_j is y_j . Hence, if $y_j \geq \lambda$, then t_1 is fired from p_1 to p_2 as in Fig. 1b and the truth degree of the output proposition d_k of (1) is calculated as $y_j * \mu_j$.

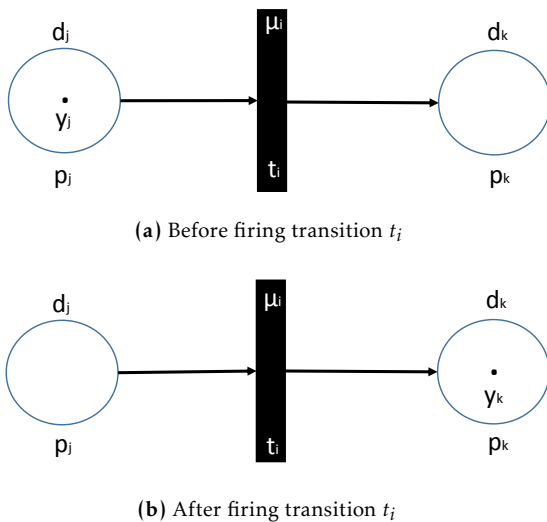


Figure 1. Firing a fuzzy petri net transition.

For example in R_1 , if the truth degree of the proposition "residual energy", i.e. fuzzified input, is 0.9 and λ is 0.5 then R_1 can be fired. Since $0.9 > 0.5$ with output truth degree equals $0.9 * 0.9 = 0.81$, which is defined as the chance of becoming a CH, i.e. fuzzified output.

2.1. Fuzzy Petri Nets Definition

FPN is considered as a bipartite directed graph which contains two types of nodes: places and transitions, where circles represent places and bars represent transitions [4]. The transition is fired from one place to reach another if the truth degree of this place is higher than the threshold as explained. A generalized FPN is defined as a 10-tuple [26] as follows,

$$\text{FPN} = (P, T, I, O, D, W, \mu, \lambda, \alpha, \beta), \quad (2)$$

where

- $P = \{p_1, p_2, \dots, p_n\}$ denotes a finite nonempty set of places which is the fuzzy variables as residual energy or the chance in the example explained.
- $T = \{t_1, t_2, \dots, t_m\}$ denotes a finite nonempty set of transitions.
- $I : T \rightarrow P^\infty$ is the input function, a mapping from transitions to bags of places.
- $O : P \rightarrow T^\infty$ is the output function, a mapping from places to transitions.
- $D = \{d_1, d_2, \dots, d_n\}$ denotes a finite set of propositions, $|P| = |D|$, it signifies the proposition that interprets the fuzzy linguistic variable, such as: *high*, *medium* and *low*.
- $W : I \rightarrow [0, 1]$ denotes the weights of the input functions, the sum of the weights W in the input arcs of a transition is 1, which denotes the influence strength of the input to the output.
- $\mu : T \rightarrow [0, 1]$ denotes the certainty factors (CF) of fuzzy rules which defines the strength of a certain rule.
- $\lambda : T \rightarrow [0, 1]$ denotes the threshold of a transition firing, a transition is not fired except if the degree of truth of the input is larger than λ .
- $\alpha : P \rightarrow [0, 1]$ is an association function which maps from places to real values between zero and one. In other words, it symbolizes the degree of truth of a certain place.
- $\beta : P \rightarrow D$ is also an association function mapping from places to propositions.

2.2. Fuzzy Petri Nets Production Rules

If the antecedent or consequence portion of a fuzzy production rule contains *AND* or *OR* connectors, fuzzy operations, it is called a composite fuzzy production rule. Chen et al. in [4] described four production rules as follows

- Type 1:** IF d_{j1} AND $d_{j2} \dots$ AND d_{jn} THEN d_k
(CF= μ_j).
- Type 2:** IF d_j THEN d_{k1} AND d_{k2} AND \dots AND d_{kn} (CF= μ_j).
- Type 3:** IF d_{j1} OR d_{j2} OR \dots OR d_{jn} THEN d_k
(CF= μ_j).
- Type 4:** IF d_j THEN d_{k1} OR d_{k2} OR \dots OR d_{kn}
(CF= μ_j).

In this paper, only type 1 is used. As shown in Fig. 2 [26], a transition of n places $P_{j1}.. P_{jn}$ which has degree of truth values $\alpha_{j1}.. \alpha_{jn}$ respectively, which are evaluated from the corresponding membership function. The

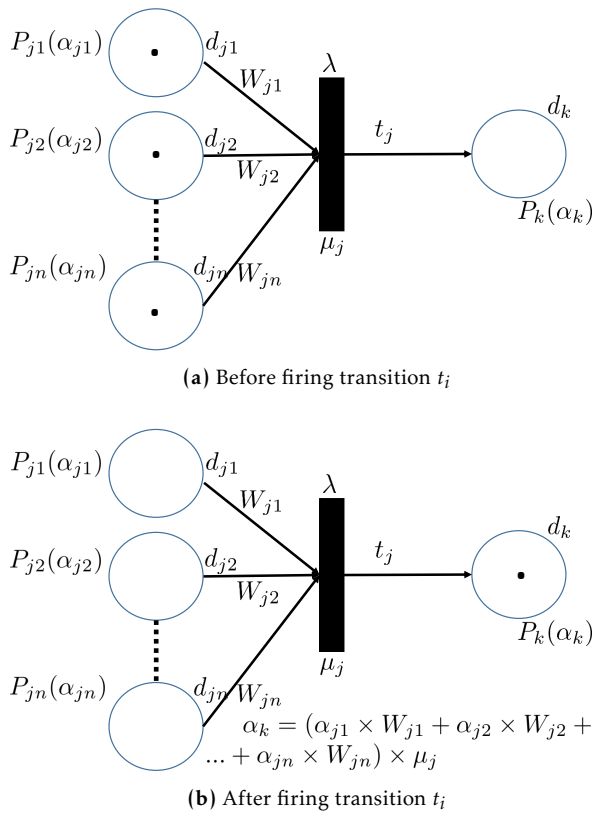


Figure 2. Fuzzy petri nets representation of type 1 rule.

transition of P_{jn} is fired to reach output place P_k only if their resulting truth value α_k is greater than the threshold λ . The output truth values is evaluated using the input truth values multiplied by its weight. Afterwards, the whole expression is multiplied by the certainty factor of this rule (μ_j) as follows

$$\alpha_k = (\alpha_{j1} \times W_{j1} + \alpha_{j2} \times W_{j2} + \dots + \alpha_{jn} \times W_{jn}) \times \mu_j. \quad (3)$$

The main advantage of FPN in clustering algorithms over fuzzy logic is that the degree of truth α_j of each fuzzy variable is multiplied by a weight W_j denoting its

importance. Unlike fuzzy logic, where all the variables are having the same importance. Hence, fuzzification takes place defining membership functions for all inputs and outputs mapping input and output crisp values into fuzzified values. Each rule defined as IF-THEN clause determines the linguistic value of the output based on the values of the input. Defuzzification converts the output in fuzzy petri net into a crisp real value that can be used by the system. Center of Gravity method (COG) which is used for defuzzification is represented as

$$\text{COG} = \frac{\sum_{i=1}^n \mu[i] \times y_i}{\sum_{i=1}^n \mu[i]}. \quad (4)$$

3. Life Time and Data Rate Optimized Clustering Techniques

Low power, low cost, and highly efficient sensor nodes are crucial requirements in WSN. Hence, optimizing their topology design under energy constraints and environmental awareness is of high importance. The following techniques were studied and applied on famous algorithms like HEED and REEMR.

3.1. Clustering Process

REEMR algorithm does not specify the rules used to control the membership functions it presents, which makes the performance of the algorithm undefined. Hence, the clustering process was modified to include another membership functions with their corresponding rules, explained later in the paper.

3.2. Multi-level hierarchy

Direct transmission between CHs and the FC consumes energy and induces data loss. As the power received from far CHs is lower than the sensitivity power of the FC due to the path loss effect, which causes the data of these CHs to be lost. This problem becomes more compelling in large-scale WSN. Hence, far away CHs suffer significant energy consumption and communication overhead during data transmission [18].

Therefore, with the significant size growth of WSNs (number of sensors, area covered, etc.), designing a multi-level cluster hierarchies is regarded as a promising option to avoid data loss preserving energy efficiency, independent of the growth of the network [2]. Although, authors in [25] (after introducing the HEED algorithm) explained inter-cluster communication technique, where CHs communicate with each other to aggregate their information via multiple hops. However, they does not specify whether it is used in the algorithm or not. Moreover, authors in [2] considered HEED as a two level hierarchical algorithm where the CHs communicate directly with the FC.

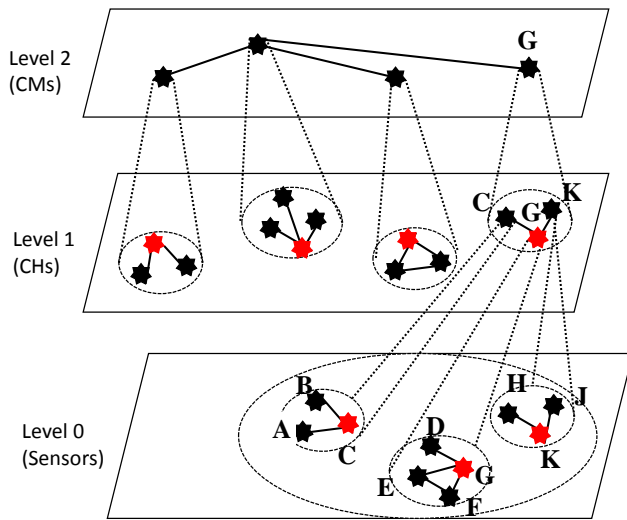


Figure 3. Three level WSN hierarchy

Moreover, the REEMR algorithm presented a multi-hop routing that is based on flooding technique in order to reach the FC which has many disadvantages.

Hence, modification to HEED and REEMR is presented in this paper to achieve better energy distribution and total energy consumption. Where multi-level hierarchical clustering is used instead of using only one cluster level. The effect of using this model appears in large-scale WSN.

The presented algorithm reuses REEMR clustering methodology for electing a CH and re-applies it to elect a Cluster Master (CM) where a CM is elected for every group of clusters as in Fig. 3. Subsequently, this adds another level in the transmission hierarchy presenting a three level hierarchical algorithm with multi-hop transmission. As the SNs transmit the data packets to the corresponding CH, it receives and transmit the aggregated packets to the responsible CM, which in turn transmits it to the FC. However, the algorithm is generic to add n levels in the hierarchy based on the application used. Moreover, the same enhancement was done on HEED algorithm using its clustering technique to elect a CH.

3.3. Electing Multiple CH Backups

Neither HEED nor REEMR tackled the problem of CH death. Being the controlling node in the cluster, which is responsible for data aggregation, a CH consumes its energy more than any other node. This exposes it to drain its battery faster than the nodes in its cluster.

However, HEED algorithm proposed the idea of re-clustering the whole network every fixed time, which depends on refreshing the network hierarchy and re-electing all the CHs. Although this might solve the CH death problem implicitly; however, if the CH dies before the re-clustering occurs, the cluster will not be able to transmit its data. As a result, the whole cluster enters an off-line state which renders data loss. Hence, HEED and REEMR were modified to include multiple backups, not only for CHs but also for CMs as well; i.e., clustering on layer 2 as in Fig.3, without affecting the network lifetime and enhancing the data rate. The modified algorithm reprocesses REEMR clustering steps where a sensor either declares itself as a CH or joins a successive CH. Afterwards, our presented algorithm chooses two backup sensor nodes for each CH, primary and a secondary. The backups are elected in the same manner as the CH. All sensor nodes buffer the IDs and the order of the backup nodes. Hence, those two backup nodes act as normal sensing nodes itself, till the CH dies. Hereafter, the primary backup node takes the lead and sends to the nodes in the cluster that it is the new CH, and so forth.

3.4. Wireless Sensor Nodes with Renewable energy sources

The paper induces further modification on both algorithms where energy harvesting sensor nodes are being used as explained in the following section.

4. WSN with Energy Harvesting Model

Despite the progress done in developing energy aware clustering techniques, large-scale dense-deployed sensor networks faces a critical challenge [29], namely energy consumption of the SNs versus its extremely limited energy resource. In parallel to the developments in network clustering techniques, a significant progress is done in prolonging WSN lifetime by utilizing energy harvested from the surrounding ambient sources such as solar, wind and vibration [21, 28]. Energy Harvesting Wireless Sensor Networks (EH-WSNs) are the result of endowing sensors with the capability of harvesting energy from the environment.

Such SNs are attached with energy harvesting devices (such as piezo electric, thermal generators and photovoltaic cells) to gain access to additional energy resource harvested from the surrounding ambient environment. The goal of traditional clustering techniques is to maximize the life time. However, EH-WSNs have additional characteristic which is making SNs reach the Energy Neutral (EN) state, where the amount of energy consumed is no more than the energy harvested in a certain period of time [12]. In other words, EH-WSNs are capable theoretically of working forever.

However, the inherent dynamic and randomness of these sources give rise to many new challenges to modeling and management of harvested energy as well as the analysis of energy harvesting systems.

There are two architectures in EH-WSN, the first one uses the harvested energy as a supplement to the chemical battery to maximize the lifetime of WSNs. The second architecture uses energy harvested as the only source to the WSNs for perpetual operation [16]. This paper takes into consideration the first architecture as in the harvesting energy resources are not permanent. As depending only on the harvested power may risk the network stability. Moreover, it puts timing limitations on the working hours of the WSN. For example, solar cells only operates in daylight, so there must be another source operating the network till it can harvest again .

Our proposed research paper targets the applications where WSNs are deployed in hostile environment, i.e., disasters prediction applications. The deployment of solar harvesting sensor nodes in such a surrounding would result in a random energy distribution. The sensor nodes used in this paper were assumed to harvest energy with harvesting rate uniformly distributed between $[0 \mu_{\max}]$. The data transmission is slotted with

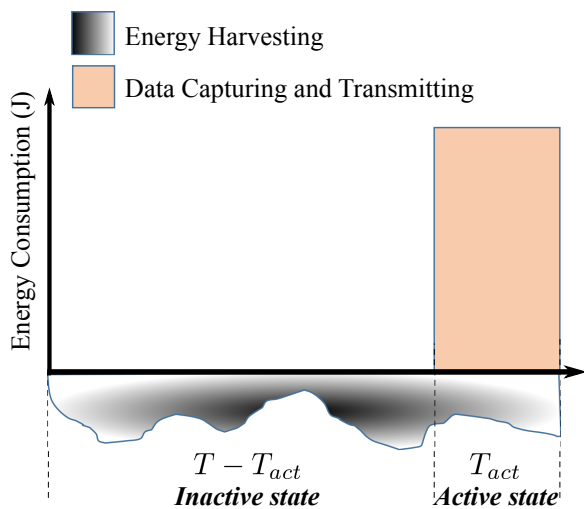


Figure 4. Energy harvesting profile against time

time slot T with two states, active and inactive. Hence, the sensor nodes sense and transmit data within T_{act} , and keeps inactive $T - T_{act}$. The harvesting process is assumed to be continuous during the whole T slot similar to Fig. 4 adopted from [1]. In other words, the sensor harvests energy even while transmission accumulating all energy harvested over the previous interval with one condition as not to cross the storage capacity of the battery. Specifically, each sensor node obeys a certain energy budget during the harvesting

interval $t \leq iT$ given by

$$\varepsilon_k(i) = \min \left\{ \varepsilon_k^{tot}; \left(\sum_{j \leq i} \varepsilon_k^H(j) \right) - \left(\sum_{j < i} \varepsilon_k^C(j) \right) + \varepsilon^{res} \right\}, \quad (5)$$

where ε_k^{tot} is the capacity of the battery of sensor k , $\varepsilon_k^H(j)$ is the harvested energy by node k in an interval $(j-1)T < t < jT$, and $\varepsilon_k^C(j)$ is the energy consumed during data transmission. Moreover, ε^{res} is the residual energy remaining in the sensor battery. However, this model cannot operate using Radio Frequency (RF) energy source as it assumes that the sensor can transmit and harvest at the same time which is not applicable using RF sources. Moreover, such sources are subject to interference that is neglected in this research.

5. Energy harvesting aware clustering algorithms

Energy Harvesting Aware clustering algorithm is presented where fuzzy petri nets concept is utilized in clustering large-scale WSN. The algorithm uses mainly energy harvesting sensor nodes exploiting the EH model and the life time optimizing techniques explained earlier.

5.1. Large-Scale WSN Model

The algorithm in this paper assumes that the nodes are uniformly distributed over the WSN field. In addition, clustering process occurs every T_{NO} Time Division Multiplexing (TDM) frames. Each node selects a CH according to a fuzzy chance, calculated from a fuzzy petri net model, with three input variables:

- i) node energy budget: calculated in (5),
- ii) node concentration: the number of neighbors found in the node transmission range, and
- iii) node centrality: the summation of the squared distance between a node and its neighbors.

Thus, it is more likely to elect a CH which has a high energy budget (to maximize network lifetime), a high number of neighbors, and a high centrality among the cluster. This increases the reachability of the CH to the nodes and decreases the transmitted energy, which is directly proportional to square the distance.

Each of these fuzzy inputs has three fuzzy sets, where the sets of the energy budget and node concentration are: low, medium, and high. While the sets of the node centrality are close, adequate and far. The membership functions are shown in Fig. 5, 6, 7, and 8 [9]. These membership functions are based on experts opinion and can be enhanced through reinforcement learning for better results [11]. The fuzzy output, chance has seven fuzzy sets which are: very small, small, rather small, medium, rather large, large, and very large.

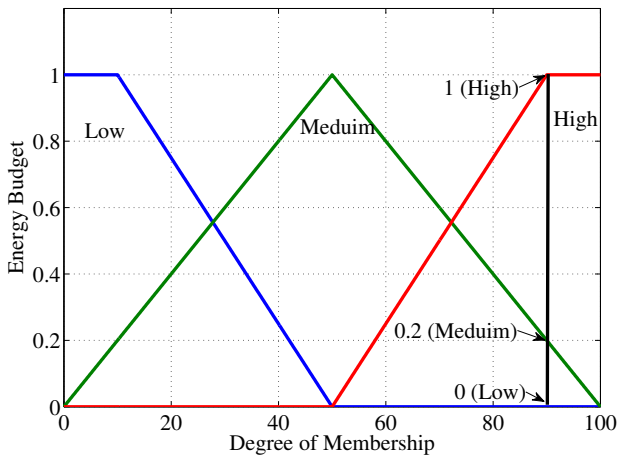


Figure 5. Fuzzification of the energy budget input

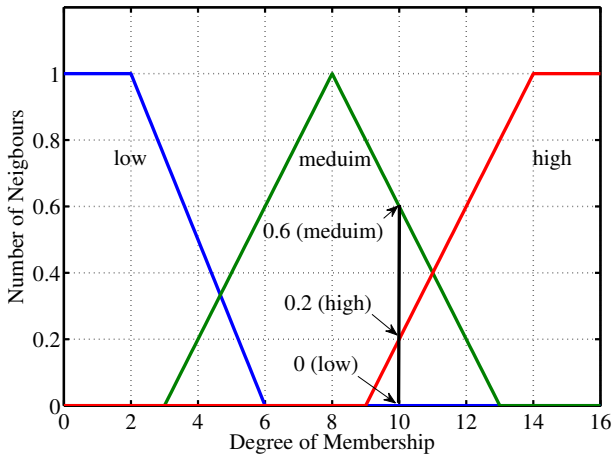


Figure 6. Fuzzification of the node degree input

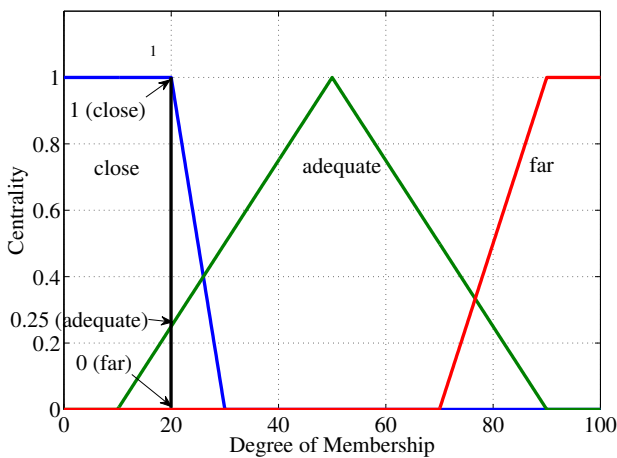


Figure 7. Fuzzification of the centrality input

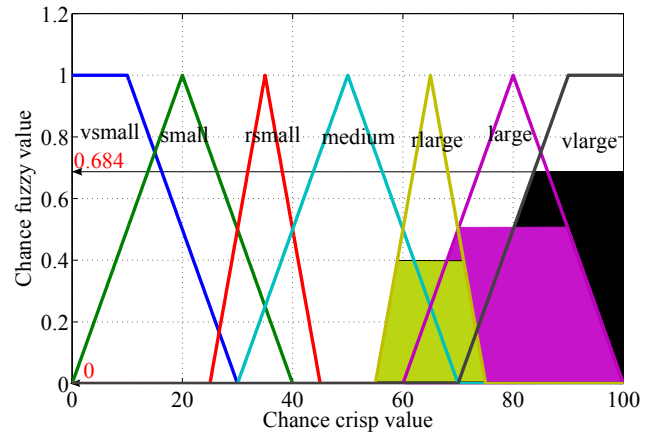


Figure 8. Rules aggregation

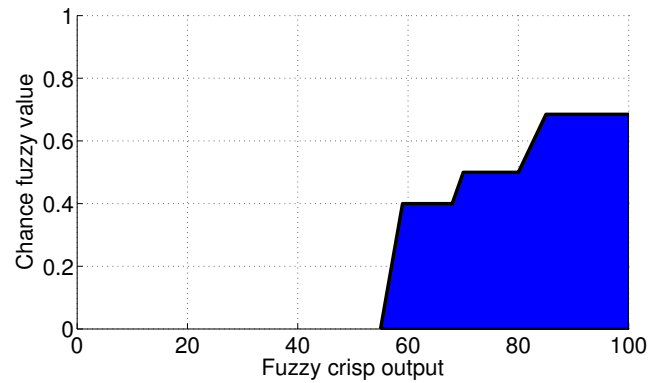


Figure 9. Defuzzification of the output after aggregating all rules from Fig. 8

Type 1 production rule is used producing $3^3 = 27$ rules as in Table 1. The weights of the variables W_{j1}, W_{j2} and W_{j3} are 0.6, 0.3 and 0.1 for energy budget, concentration and centrality respectively.

Example 5.1. Assume that a fuzzy logic algorithm is evaluating two production rules only such that:

R_1 : IF energy budget is high AND number of neighbors is high AND centrality is close THEN the node cluster head election chance is vlarge.

R_2 : IF energy budget is low AND number of neighbors is low AND centrality is far THEN the node cluster head election chance is vsmall.

The sensor energy budget is 90%, number of neighbors is 10 nodes, and it is centrality is 20 m^2 (where the centrality is the summation of the distance to the CH squared). The chance of this sensor to be a CH is calculated through this steps: fuzzification of the crisp inputs, rules evaluation, aggregation and finally defuzzification.

Table 1. Fuzzy rule base

	Energy Budget	Concentration	Centrality	Chance
1	low	low	close	small
2	low	low	adeq	small
3	low	low	far	vsmall
4	low	med	close	small
5	low	med	adeq	small
6	low	med	far	small
7	low	high	close	rsmall
8	low	high	adeq	small
9	low	high	far	vsmall
10	med	low	close	rlarge
11	med	low	adeq	med
12	med	low	far	small
13	med	med	close	large
14	med	med	adeq	med
15	med	med	far	rsmall
16	med	high	close	large
17	med	high	adeq	rlarge
18	med	high	far	rsmall
19	high	low	close	rlarge
20	high	low	adeq	med
21	high	low	far	rsmall
22	high	med	close	large
23	high	med	adeq	rlarge
24	high	med	far	med
25	high	high	close	vlarge
26	high	high	adeq	rlarge
27	high	high	far	med

Fuzzification of crisp inputs. This step depends on transforming the three mentioned crisp (non fuzzified) inputs into a fuzzified inputs. This is done through finding the intersection of the inputs values with their membership functions. Since each membership function has three graphs with three levels for the same input; thus, each input can have three intersections or fuzzy values. In this example, if the node energy budget is 90%, this means that its fuzzified value is 1 (if considered as high), 0.2 (if medium is considered), and 0 (if low is considered) as demonstrated in Fig. 5. Similarly, the node degree has 0 fuzzy value (if low), 0.6 (if medium) and 0.2 (if high) as in Fig. 6. Additionally, the centrality values as Fig. 7 is 1 (if close), 0.2525 (if adequate), and 0 (if far).

Fuzzy Production Rules Evaluation. Since each crisp input has three possible fuzzified values. Hence, the decision of which value of them to select is based on which rule the engine is evaluated. For instance, if the rule R_1 is being evaluated, then the fuzzy value of energy budget is the one corresponding to the high level, which is 1. On the other side, the fuzzy value of the node degree is the one corresponds to high level, which is 0.2. The centrality value corresponding to the close level is 1. As explained in Fig. 2b, the truth output values, which

are the chances to be a CH), are calculated from (3). Hereto, the output α_k is the chance of the sensor to be CH. The truth values of the inputs in this case are $\alpha_{j1}=1$, $\alpha_{j2}=0.2$, and $\alpha_{j3}=1$. Moreover, the weights given for energy budget, node degree, and centrality are 0.6, 0.3, and 0.1, respectively. The output α_k is calculated as below:

$$\alpha_k = ((0.6 \times 1) + (0.3 \times 0.2) + (0.1 \times 1)) \times 0.9 = 0.684. \quad (6)$$

The threshold of firing the transition λ is 0.5, and since 0.684 is greater than 0.5, then the transition is fired to place P_k and the fuzzy chance α_k is of 0.684 (see Fig. 8). Similarly, rule R_2 is evaluated as follows:

$$\alpha_k = ((0.6 \times 0) + (0.3 \times 0) + (0.1 \times 0)) \times 0.9 = 0. \quad (7)$$

The other 25 rules are evaluated the same way, for demonstration we assume that there exists only two rules in this example (see Fig. 8).

Aggregation of Fuzzy Petri Net Rules. Aggregation in fuzzy logic is the union of all outputs obtained from all the evaluated rules (27 rules). Since, we need to aggregate all the rules, the OR operator is used. Where the OR (Union) operator in fuzzy logic selects the maximum of the membership functions of the output rules. In example 5.1, the maximum value of R_1 is 0.684 and in R_2 is 0. Figure 8 gives an illustration of how aggregating many rules would look like

Defuzzification. This is the last step of the FIS where the fuzzy chance value is converted into a real crisp output to be used in the election process. This is done by calculating the COG as in (4) in order to evaluate the centroid of the aggregated rules output which is the center of the area under the curve in Fig. 9.

5.2. EHFP Algorithm Steps

This subsection describes in details the steps of EHFP clustering technique discussed in Algorithm 1.

Initialization. In this stage, every node sends a broadcast message with its maximum transmission power to all the surrounding nodes in an important step for neighboring discovery. Afterwards, each node formulates a set S_{nbr} containing all the neighbors that heard this message and replied back. This set defines the cluster of this node and all the nodes within this cluster are considered as the neighbors of the originating node. Meanwhile, each node computes its cost which is calculated as an output of a fuzzy logic system explained earlier. Then, the neighbor replies with its cost such that every node in the field know the cost of all of its surrounding neighbors. Afterwards, the node starts the CH selection phase.

Algorithm 1 EHFP Clustering algorithm

```

1: Initialize
2: Define set of neighbors  $S_{nbr} = \{v : v \text{ lies within the sensor cluster range}\}$ 
3: Compute and broadcast Fuzzy cost to  $S_{nbr}$ 
4:  $is\_final\_CH \leftarrow \text{FALSE}$ 
5: Compare
6:  $sensor\_cluster\_head \leftarrow \max\_cost(S_{nbr})$ 
7: if  $sensor\_cluster\_head = \text{NodeID}$  then
8:    $is\_final\_CH \leftarrow \text{TRUE}$ 
9:    $Cluster\_head\_msg(\text{NodeID}, tentative\_CH, cost)$ 
10: end if
11: Finalize
12: Define set  $S_{CH} = \{v : v \text{ lies within sensor final cluster head}\}$ 
13: if  $is\_final\_CH = \text{FALSE}$  then
14:   if  $S_{CH} \neq \varphi$  then
15:      $sensor\_cluster\_head \leftarrow \max\_cost(S_{CH})$ 
16:      $join\_cluster(cluster\_head\_ID, \text{NodeID})$ 
17:   else
18:      $Cluster\_head\_msg(\text{NodeID}, final\_CH, cost)$ 
19:   end if
20: else
21:    $Cluster\_head\_msg(\text{NodeID}, final\_CH, cost)$ 
22: end if
23: IV. Backup Selection
24: if  $sensor\_cluster\_head = \text{NodeID}$  then
25:   Define set  $S_{nodes} = \{v : v \text{ is a node in the cluster}\}$ 
26:    $CH\_primary\_backup \leftarrow \max\_cost(S_{nodes})$ 
27:    $CH\_secondary\_backup \leftarrow 2^{nd} \max\_cost(S_{nodes})$ 
28:    $Cluster\_head\_msg(\text{Primary\_nodeID}, secondary\_nodeID, backup\_CH)$ 
29: end if

```

Compare. In this phase, every node compares its cost to that of its neighbors. The node declares itself a CH if its cost is the highest cost. Subsequently, this sensor broadcasts a message announcing itself a tentative CH of this cluster namely "*Cluster_head_msg*" which contains the node ID and its status (tentative or final CH). Thus, whenever a node receives a new tentative Cluster Head Msg from any of its neighbors, it adds it to the set of cluster heads S_{CH} . This set contains all the neighbors who declared themselves as tentative CHs.

Finalize. If a node reached this stage without announcing itself a CH, it must join one. Therefore, it compares the cost of all the nodes in set S_{CH} and chooses the CH with the highest cost to join. Afterwards, this node sends a join request to this CH to become a part of its cluster and sends its sensed data to it. However, if this set is empty which means that this node lies alone in its cluster, it announces itself the CH. On the other hand, if the node announced itself in the previous stage to be a tentative CH, it resends another message namely

Cluster_head_msg(NodeID, final_CH) declaring itself as a final CH for this cluster.

CH Backup Election. In contrast to all algorithms using fuzzy logic in clustering, EHFP enhances the network robustness by electing a CH backup. As mentioned before in most algorithms, the cluster enters an off-line state if its CH dies. Afterwards, the cluster stops sending any data waiting for re-clustering to occur to elect a new CH which causes data loss and unreliability in some critical applications. However, in EHFP, each sensor elects three CHs with the top three chances, one main CH and the other two are backups (primary and secondary). In case the CH dies, the sensor aggregates the data sending it to the primary backup and when it dies it sends the data to the secondary one, applying maximum robustness and preserving the data rate along the network lifetime.

Parameters for CM Election. The election process takes place among the CHs to elect a master taking into consideration the following fuzzy crisp inputs

- node residual energy
- node degree
- distance to the FC

As noticed, it is the same parameters for CH election but replacing the centrality with the distance to the FC as a fuzzy input variable. Since, being a CM, it is more important to be near the FC than being central. As the CM sends a huge amount of aggregated data from the corresponding clusters to the FC and due to path loss, the CM must be as near as possible to the FC to minimize the error rate. The next part explains how these parameters are used.

CMs Rules Evaluation. For exploiting the maximum sensor storage resources, the election of the cluster master and head are done using the same 27 fuzzy production rules in table 1. In addition, the same membership functions are used for both energy and node degree as in Fig.5 and 6 respectively. Moreover, the new variable which is the distance to the FC has the same membership function as the centrality variable which is used in CH election in Fig.7. The rules are evaluated in the same way, but the fuzzy variables are given different weights. EHFP optimized the weight values that can acquire the longest lifetime. Thereafter, the weights are optimized as follows: residual energy is given W_1 of 0.6 while W_2 and W_3 are both of 0.2 for Node concentration and distance to the FC respectively. Similar to any CH, each CM has multiple backups as the backup role in a CM is of higher importance. As a CM is most probably the first node to die in a network due to its heavy duties, leading to the highest energy consumption among all the sensors. This is due to that

the CM aggregates the data collected from its following clusters, then if it dies, a huge amount of data will be lost. Hence, it is inevitable that the following clusters must backup CM. Hence, EHFP ensured the durability of the network no matter what with self organizing capabilities.

5.3. Hybrid Source Energy Harvesting Aware clustering algorithm

The ambient energy sources can be intermittent and inconsistent. Therefore, relying on solar energy only can be insufficient for a continuously operating network, as solar energy is variable and only available for few hours per day [22]. Hence, a single energy source is not adequate sometimes. Therefore, another key characteristic of EHFP is proposing a clustering algorithm with hybrid source EH system instead of solely relying on solar energy. Our proposed algorithm utilizes thermal energy besides the solar one, as a hybrid source of renewable energy. Hence, our system model can be altered where a certain percentage of the sensors are now able to harvest simultaneously from both energy sources whenever they are available, instead of harvesting from individual energy source at one time. In such case, the distribution of the sensor nodes in the network is as follows: 20% operates using hybrid source (solar and thermal harvesters) and 80% operates using thermal energy harvesters only with harvesting rate uniformly distributed between $[0 \mu_{\max}]$.

5.4. Opportunistic Transmission

In order to further enhance the network lifetime trying to reach energy stability, an opportunistic transmission technique is applied on EHFP. The presented technique controls the transmission of each sensor node based on its harvesting rate and its energy budget. In such a case, every sensor k calculates a probability of transmission, T_{prob} as follows

$$T_{\text{prob}} = \frac{\mu_k}{\mu_{\max}} \times \frac{\epsilon_k}{\epsilon_k^{\text{tot}}} \quad (8)$$

where μ_k is the harvesting rate of the node k , μ_{\max} is the maximum harvesting rate for all nodes, ϵ_k is the energy budget of the node, and ϵ_k^{tot} is the maximum battery storage. Afterwards, each sensor selects one of four possible probabilities p_i , where $i \in 1, 2, 3, 4$, based on the value T_{prob} . Subsequently, every sensor generates a uniformly distributed variable r from 0 to 1. Hence, a sensor node sends its data to the CH if (and only if) $r \leq p_i$.

As a design aspect, the probability p_i can be changed according to the application used where the probability of transmission can be small in case of transmitting low priority data sensor and can be increased in case

of disaster scenarios, where data transmission is of high priority in this case.

Algorithm 2 Opportunistic Transmission algorithm

```

1: calculate  $T_{\text{prob}}$ 
2: if ( $T_{\text{prob}} \geq p_i$ ) then
3:   generate random  $r \in (0, 1)$ 
4:   if ( $r \leq p_i$ ) then
5:     Transmit data to CH
6:   else
7:     Quit harvesting
8:   end if
9: end if

```

6. Simulations and Analysis

Results of our research have been simulated using MATLAB in order to illustrate the performance (represented in the WSN life-time) of the proposed algorithms. For evaluating our algorithm with real-valued data, one can reuse the measurement in [24, Table 1] that uses the real measurements sensors for light, humidity, and temperature.

Here, we considered the data transmission phase for all sensor nodes in the WSN, where the sensors are asked to transmit their data periodically to a FC. The re-clustering process occurs every T_{NO} seconds. The simulations are done assuming $N = 200$ sensor nodes, which are randomly allocated and uniformly distributed in a $100\text{m} \times 100\text{m}$ field area. The FC is located always at the center of this WSN field. The proposed algorithm is only applicable on networks whose area dimensions ($A \times A$) are comparable to d_0 . On the other side, if $A \ll d_0$, the algorithm must be extended to include a multi-hop (more than 3 level hierarchy) technique for delivering data to the FC.

The radio model applied in this paper calculates energy consumed according to Friss equation when the distance d between transmitter and receiver is less than d_0 (the reference path-loss distance), which is adopted from [25]. Thus, the energy consumed in transmitting n_b bits for any node index k is given by

$$\epsilon_k^C = n_b \times (E_{\text{elec}} + E_{\text{amp}} \times d^n), \quad (9)$$

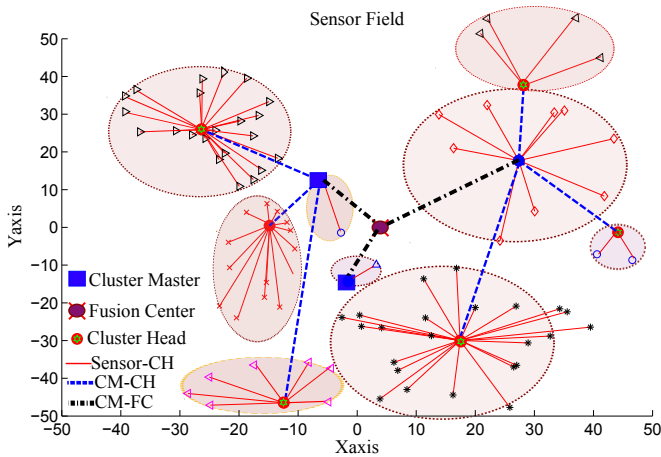
where $E_{\text{amp}} = \epsilon_{\text{fs}}$, which is the power loss in free-space propagation with the path-loss exponent $n = 2$. This case applies only when $d > d_0$. When $E_{\text{amp}} = \epsilon_{\text{mp}}$ (multi-path environment), the path-loss exponent is set to $n = 4$. The power consumption during receiving is only limited to the electronic circuit energy consumption per bit (E_{elec}), e.g., the receiver consumes power for receiving n_b^{RX} bits equal to $n_b^{\text{RX}} \times E_{\text{elec}}$ Joules. All the simulation parameters used in this paper are listed in Table 2.

Table 2. Variables

Parameter	Value
Initial energy per node	2 J
E_{elec}	50 nJ/bit
ϵ_{fs}	10 nJ/(bit.m ²)
ϵ_{mp}	0.0013 pJ/(bit.m ⁴)
d_o	75 m
Cluster radius	25 m
Size of a data packet	100 bytes
Broadcast packet size	25 bytes
Packet header size	25 bytes
Round(T_{NO})	20 TDM
Area of the Panels	10 cm ²
Harvesting rate μ_{max}	15 μ watt/cm ²

The network life time is the most important metric to evaluate the performance of the network. In this paper, it is defined as the time till the last node in the network runs out of energy.

The WSN cluster formation is shown in Fig.10, only 100 nodes are represented in this figure for clarity of demonstration.

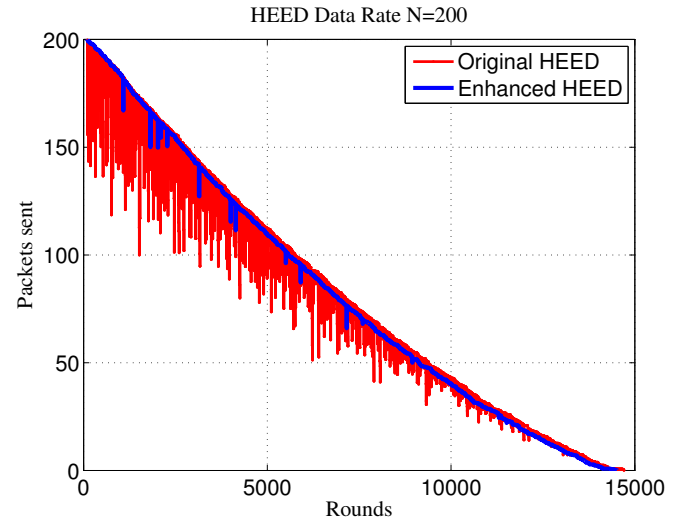
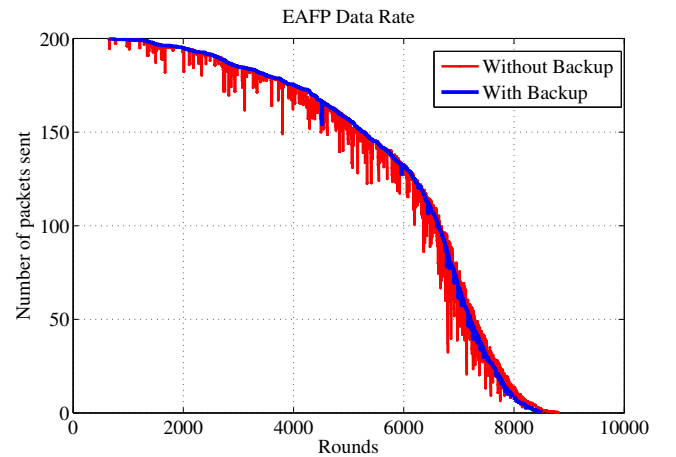

Figure 10. Hierarchical cluster formation for the proposed EHFP clustering scheme

The results shown simulated three models of the sensor nodes.

6.1. Case 1: Sensors with Limited Batteries

This case simulated the WSN with battery limited sensor nodes without any recharging capabilities. In order to explore the effect of having multiple backups, the data rate was simulated using the original HEED versus our enhanced HEED. The results in Fig. 11 show data rate stability in the enhanced heed by having multiple backups for the CH and CM. As in HEED, the whole cluster goes off-line when a CH is dead, stopping the flow of data from this cluster waiting for the next clustering round. This explains the dropped packets

per rounds in the graph. Subsequently, this affects the overall data received from this network, causing data loss. Such a problem was recovered by using our proposed multiple backups inside the CH range and the CM range. Where a backup CH announces itself a CH right after a CH dies, accordingly not much data will be lost anymore, and the data packet rate expected from the sensor network will be achieved.


Figure 11. Data Rate of Original HEED vs Enhanced HEED without EH

Figure 12. EHFP data packet rate transmission with and without backup CHs

Additionally, Fig.12 depicts the transmitted data packet rate of EHFP with and without backup, presented in total number of packets per round. The results reveal that having backups enhances the network data rate stability intensively.

The network lifetime was simulated for both EHFP and enhanced HEED in Fig. 13. The results here show that

EHFP maintain most of the network alive longer than the enhanced HEED till it reaches half the network life time. Afterwards, the whole network dies more rapidly. On the other side, the enhanced HEED keeps a constant decrease in its slope, reaching 50% of its network capacity faster than the EHFP by about 1000 rounds. However, the enhanced HEED keeps the network decay rate much better than that in case of EHFP after the marked half-life time. Here, the enhanced HEED stays active with few sensors much longer. This is clear as our EHFP (without EH yet) distributes the load almost equally on all sensors, CHs, and backups, which exhausts the batteries faster than HEED.

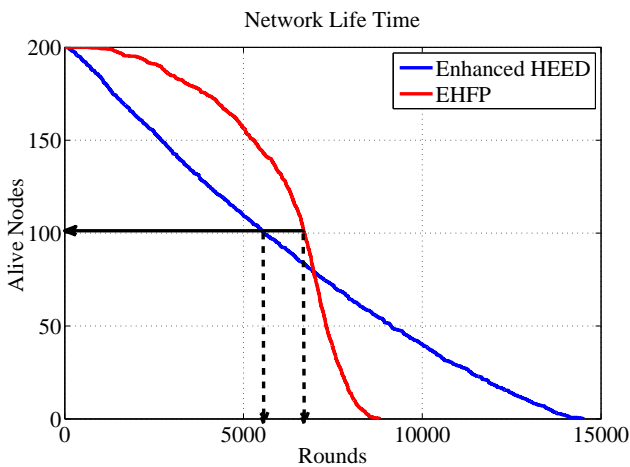


Figure 13. Life-time of the proposed EHFP clustering and the enhanced HEED without EH up till no node alive in the network

Bottom line, in a non harvesting WSN, EHFP can be used in applications which require keeping the whole network alive for the longest time, where maintaining the data rate, and data aggregation from the whole sensor field is a priority. However, the enhanced HEED can be used in applications that must be kept alive even with the least number of nodes.

6.2. Case 2: Sensors with EH Solar Panels

This case simulates a solar energy harvesting WSN, where every sensor has a solar panel on the top of it. The solar panels used are of average harvesting area of 10 cm^2 , the maximum harvesting rate μ_{\max} is $15 \mu\text{watt/cm}^2$ where each node is harvesting randomly between 0 and μ_{\max} .

In Fig. 14, EHFP shows a much longer life time span compared to enhanced EH-HEED. compared to both schemes without harvesting (as seen in Fig.13). As after 15000 rounds, EHFP is still working with 40% of the network, while the enhanced EH-HEED is working with only 20%. Moreover, the Enhanced-EH HEED reached half of its network utilization at 6800 rounds compared to 13800 rounds in EHFP.

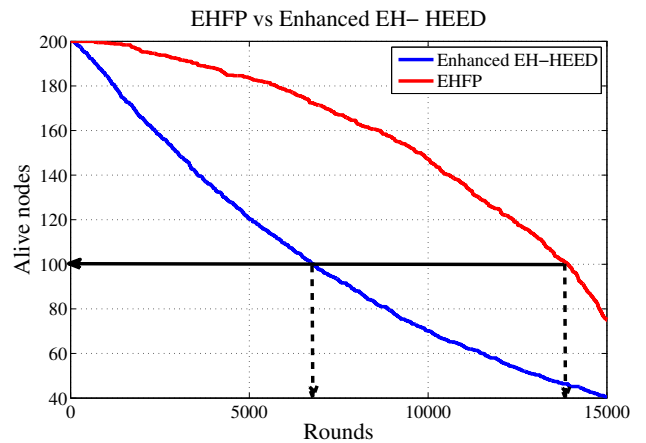


Figure 14. Life-time comparison of our proposed EHFP clustering vs enhanced EH-HEED with EH up-till 40 nodes alive

6.3. Case 3: Sensors with Hybrid EH source

This case simulates an EH WSN, where only 20% of the network is harvesting with both solar and thermal energy sources simultaneously. However, the rest of the network is operating using thermal harvested energy only (which has even less power harvesting rate than the solar). The maximum harvesting rate (μ_{\max}) of the sensors thermo electric generators is $10 \mu\text{watt/cm}^2$, with the same harvesting area as the solar panel, i.e., 10 cm^2 . Fig.15 shows how introducing the hybrid energy harvesters in the SNs improved the lifetime decreasing the SNs death rate significantly reaching a nearly neutral state, even with 80% if the sensors are having less harvesting capability. The reason behind this time boost is that the CH (those sensors which are kept busy most of the time) are selected from those with dual harvesting source nodes.

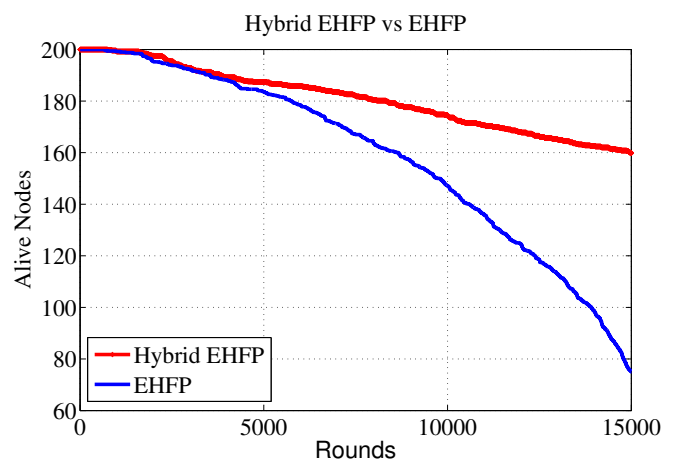


Figure 15. Life time performance of the proposed EHFP with hybrid EH source (solar and thermal)

6.4. Case 4: Sensors with EH-Aware Opportunistic Transmission

The opportunistic transmission presented in this paper is applied on Case 2 only, which is described in subsection 6.2. Here, we simulate two different opportunistic scenarios, where each scenario has its own transmission opportunity p_i . These opportunities are decided based on the sensor T_{prob} , which is calculated using the actual energy status and EH capability as stated in (8). Table 3 lists those two samples of our selected transmission opportunities (scenarios), where they all have been selected based on our convenience. However, changing those opportunities p_i can be always subject to network demands and emergency situations. As depicted in Fig. 16, the network life-time is simply extended by reducing the transmission chances if the harvesting rate is not sufficient (again, based on (8)). Particularly speaking, the lifetime of our proposed EHFP (without hybrid sources) is improved and opens the door for further investigation and building more adaptive opportunistic transmission. From another prospective, doing such a probabilistic transmission can control the network collision rate; hence, save energy, if the model is assumed to be a random multiple access (MAC) model rather than our scheduled time-division multiple access transmission.

Table 3. Transmission Opportunities values

Probability (sorted from low to high)	Scenario 1	Scenario 2
p_1	0.1	0.2
p_2	0.2	0.4
p_3	0.3	0.6
p_4	0.4	0.8

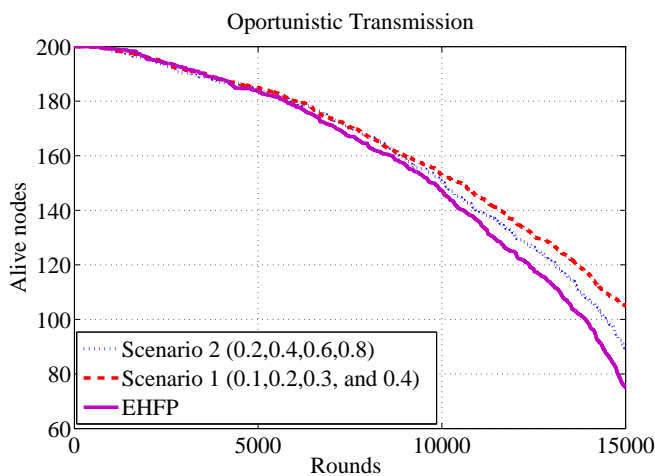


Figure 16. EHFP with different opportunistic transmission mechanisms

7. Conclusion

Emerging research in energy harvesting provides practical solutions to applications that require the sensor field to live for a very long time without human interference. Here, EHFP algorithm is proposed, where it depends on fuzzy petri nets reasoning to select CHs according to three fuzzy variables: energy budget, concentration, and centrality of the nodes. The algorithm took into consideration the harvesting rate of each node in the energy budget calculation. Hence, the most likely CH to be elected, is the node that can harvest more, with largest number of neighbors and more central. The algorithm assures network robustness by electing primary and secondary backups to the cluster head. It also elects CM nodes with their backups, which aggregate data from the CH nodes and transmit it to FC ensuring reliability. Moreover, EHFP presented an opportunistic transmission that controls what a sensor can send with tuning probabilities that can be changed in case of disasters and build it based on EH rates. Besides, a hybrid source sensors are supported by our EHFP resulting in a nearly neutral network even with lower EH capabilities. Results show that the proposed algorithm maximized the life-time of the network achieving energy savings compared to conventional clustering algorithms.

References

- [1] W. C. 0016, Y. Andreopoulos, I. J. Wassell, and M. R. D. Rodrigues. Unlocking energy neutrality in energy harvesting wireless sensor networks: An approach based on distributed compressed sensing. *CoRR*, abs/1312.4207, 2013.
- [2] L. Bailey and A. X. Liu. *RFID Authentication and Privacy Protocols*, chapter in Book *RFID and Sensor Networks: Architectures, Protocols, Security and Integrations*. Taylor & Francis Group.
- [3] O. Boyinbode, H. Le, A. Mbogho, M. Takizawa, and R. Poliah. A survey on clustering algorithms for wireless sensor networks. In *13th International Conference on Network-Based Information Systems (NBIS)*, 2010, pages 358–364, Sept 2010.
- [4] S.-M. Chen, J.-s. Ke, and J.-F. Chang. Knowledge representation using fuzzy petri nets. *IEEE Transactions on Knowledge and Data Engineering*, 2(3):311–319, Sep 1990.
- [5] M. L. D. J. Dechene, A. El Jardali and A. Sauer. A survey of clustering algorithms for wireless sensor networks. *Department of Electrical and Computer Engineering, The University Of Western Ontario, London, Ontario, Canada*, (10).
- [6] S. Dastgheib, H. Oulia, and M. Ghassami. An efficient approach for clustering in wireless sensor network using fuzzy logic. In *2011 International Conference on Computer Science and Network Technology (ICCSNT)*, volume 3, pages 1481–1486, Dec 2011.

- [7] A. Förster, A. Förster, and A. Murphy. Optimal cluster sizes for wireless sensor networks: An experimental analysis. In J. Zheng, S. Mao, S. Midkiff, and H. Zhu, editors, *Ad Hoc Networks*, volume 28 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 49–63. Springer Berlin Heidelberg, 2010.
- [8] I. Gupta, D. Riordan, and S. Sampalli. Cluster-head election using fuzzy logic for wireless sensor networks. In *Proceedings of the 3rd Annual Communication Networks and Services Research Conference, 2005.*, pages 255–260, May 2005.
- [9] I. Gupta, D. Riordan, and S. Sampalli. Cluster-head election using fuzzy logic for wireless sensor networks. In *Proceedings of the 3rd Annual, Communication Networks and Services Research Conference, 2005.*, pages 255–260, May 2005.
- [10] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, 2000.*, pages 10 pp. vol.2–, Jan 2000.
- [11] M. Hossein, F. Zar, J. Jouzdani, and M. F. Zar. Online at: www.jus.org.uk reinforcement learning for fuzzy control with linguistic states, 2007.
- [12] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava. Power management in energy harvesting sensor networks. *ACM Trans. Embed. Comput. Syst.*, 6(4), Sept. 2007.
- [13] J.-M. Kim, S.-H. Park, Y.-J. Han, and T.-M. Chung. Chef: Cluster head election mechanism using fuzzy logic in wireless sensor networks. In *10th International Conference on Advanced Communication Technology, 2008. ICACT 2008.*, volume 1, pages 654–659, Feb 2008.
- [14] P. Kumari, M. Singh, and P. Kumar. Survey of clustering algorithms using fuzzy logic in wireless sensor network. In *2013 International Conference on Energy Efficient Technologies for Sustainability (ICEETS)*, pages 924–928, April 2013.
- [15] Q. Liang. Fault-tolerant and energy efficient wireless sensor networks: a cross-layer approach. In *Military Communications Conference, 2005. MILCOM 2005. IEEE*, pages 1862–1868 Vol. 3, Oct 2005.
- [16] J. Meng, X. Zhang, Y. Dong, and X. Lin. Adaptive energy-harvesting aware clustering routing protocol for wireless sensor networks. In *2012 7th International ICST Conference on Communications and Networking in China (CHINACOM)*, pages 742–747, Aug 2012.
- [17] A. Mostafa and K. Hassan. Robust energy harvesting aware clustering with fuzzy petri net reasoning algorithm. In *2014 IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 378–383, Oct 2014.
- [18] P. Patil, U. Kulkarni, and N. H. Ayachit. Some issues in clustering algorithms for wireless sensor networks. *IJCA Special Issue on 2nd National Conference- Computing, Communication and Sensor Network (CCSN)*, (4):18–23, 2011. Full text available.
- [19] S. Peng and C. P. Low. Energy neutral clustering for energy harvesting wireless sensors networks. In *19th IEEE International Conference on Networks (ICON), 2013*, pages 1–6, Dec 2013.
- [20] S. I. A. Raed I Hamed and R. Parveen. A new approach for modelling gene regulatory networks using fuzzy petri nets. *Journal of Integrative Bioinformatics*, 7, Sept. 2010.
- [21] S. Sudevalayam and P. Kulkarni. Energy harvesting sensor nodes: Survey and implications. *Communications Surveys Tutorials, IEEE*, 13(3):443–461, Third 2011.
- [22] Y. K. Tan and S. K. Panda. Energy harvesting from hybrid indoor ambient light and thermal energy sources for enhanced performance of wireless sensor nodes. *IEEE Transactions on Industrial Electronics*, 58(9):4424–4435, 2011.
- [23] Y. Tashtoush and M. Okour. Fuzzy self-clustering for wireless sensor networks. In *IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, 2008. EUC '08.*, volume 1, pages 223–229, Dec 2008.
- [24] A. L. M. T. P. G. P. P. Usman Raza, Alessandro Camera. Practical data prediction for real-world wireless sensor networks. *Department of Information Engineering and Computer Science, University of Trento, Trento, Italy.*, April 2014.
- [25] O. Younis and S. Fahmy. Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 3(4):366–379, Oct 2004.
- [26] Z. Yu, X. Fu, Y. Cai, and M. C. Vuran. A reliable energy-efficient multi-level routing algorithm for wireless sensor networks using fuzzy petri nets. *Sensors*, 11(3):3381–3400, 2011.
- [27] P. Zhang, G. Xiao, and H. Tan. A preliminary study on lifetime maximization in clustered wireless sensor networks with energy harvesting nodes. In *8th International Conference on Information, Communications and Signal Processing (ICICS) 2011*, pages 1–5, Dec 2011.
- [28] P. Zhang, G. Xiao, and H. Tan. A preliminary study on lifetime maximization in clustered wireless sensor networks with energy harvesting nodes. In *8th International Conference on Information, Communications and Signal Processing (ICICS) 2011*, pages 1–5, Dec 2011.
- [29] S. Zhang. *Modeling, Analysis and Design of Energy Harvesting Communication Systems*. PhD thesis, University of Rochester, Rochester, New York, May 2013.