

Energy Efficient Clustering Protocol Using Fuzzy Logic for Heterogeneous WSNs

Samayveer Singh¹ · Satish Chand¹ · Bijendra Kumar¹

Published online: 25 July 2015

© Springer Science+Business Media New York 2015

Abstract In this paper, we propose a multilevel heterogeneous network model characterized by a parameter, called model parameter that helps the model defining a network of 0-level, 1-level, 2-level, 3-level, and 4-level heterogeneity. We consider the hybrid energy efficient distributed (HEED) clustering protocol to estimate the network lifetime and accordingly name it as HEEDML (HEED MultiLevel). Depending on the heterogeneity, its variants have been named as HEEDML-0, HEEDML-1, HEEDML-2, HEEDML-3, and HEEDML-4 for 0-level, 1-level, 2-level, 3-level, and 4-level heterogeneity. The HEEDML-0 is the original HEED. The model parameter also determines the numbers of nodes of each type. We use the same parameters as in the HEED to decide the cluster heads: residual energy and node density. We also consider fuzzy implementation of the HEEDML. The HEEDML-1, HEEDML-2 HEEDML-3, and HEEDML-4 increase network lifetime by 39.61, 117.38, 182.69, and 223.7 %, corresponding to the increase in the network energy as 9.2, 17.40, 21.80, and 24 %, with respect to the HEEDML-0. The fuzzy implementation further increases the network lifetime. The HEEDML-FL-0, HEEDML-FL-1, HEEDML-FL-2, HEEDML-FL-3, and HEEDML-FL-4 increase the network lifetime by 193.84, 270.31, 375.84, 448.33, and 589.07 %, corresponding to the same increase in network energy as that for HEEDML (all levels) with respect to that of the HEEDML-0. The HEEDML-FL-0 increases the network lifetime by 193.84 % with respect to the original HEED without increasing the network energy. The packet delivery, total energy consumption, throughput, average delay, and traffic load have better results.

✉ Samayveer Singh
samayveersingh@gmail.com

Satish Chand
schand86@hotmail.com

Bijendra Kumar
bizender@gmail.com

¹ Division of Computer Engineering, Netaji Subhas Institute of Technology, New Delhi 110078, India

Keywords Heterogeneity · Energy efficiency · Fuzzy logic · Lifetime · Sensor nodes

1 Introduction

Recent technological developments have led to an important class of computer networks based on new platform, networking structure, and interface that enable novel, low cost, and can operate in inhospitable conditions such as battlefield surveillance, biological and chemical attack detection and protection, etc. The network structure is termed as wireless sensor network (WSN) that has various functionalities such as sensing, intelligence, communications, and actuations. The sensor networks may be classified into two broad classes: homogeneous and heterogeneous sensor networks. In a homogeneous network, all sensor nodes have identical energy, whereas in a heterogeneous network, not. It has been reported in literature that the cost incurring in increasing the energy of a sensor is much less than that of by using additional sensors of the same amount of energy. This aspect leads to energy heterogeneity. There are two more types of heterogeneity: link heterogeneity and computational heterogeneity. These also implicitly depend on energy. Thus, the energy heterogeneity is the fundamental heterogeneity. If there is no energy heterogeneity, computational heterogeneity and link heterogeneity may bring negative impact on the network lifetime. The energy management in a heterogeneous WSN is more challenging issue as compared to a homogeneous WSN. Most of the works have been discussed by considering the homogeneous WSN for energy preservation and they do not perform well for heterogeneous WSNs. One of the efficient mechanisms for using network energy is clustering in which the sensor nodes are organized in clusters, each having a cluster head. In clustering, the cluster heads can be static or dynamic. In case of static cluster heads, the network will have energy leakage problem in which there can be nodes with non-zero energy, but they cannot communicate with the sink. The best way is to have dynamic cluster heads in which the cluster heads are chosen based on the prespecified parameters including the residual energy of nodes. This mitigates the problem of load balancing as all the nodes get chance to be selected as cluster heads as long as they have non-zero energy. The nodes in WSNs have fixed energy that cannot be recharged using external source. Therefore, the routing mechanisms for WSNs require to be simple and straightforward so that much power is not consumed in computation and also the communication energy is minimized. The routing protocols depend on the method used to acquire and maintain the information, and also on the path computation on the acquired information [1–4]. The protocols for route selection can be proactive, reactive, and hybrid protocols. The proactive protocols maintain consistent and accurate routing tables of all nodes using periodic dissemination of routing information and all routes are computed before they are required. Most of these routing protocols can be used in both flat and hierarchical structured networks [1, 2, 4–6]. The reactive protocols do not maintain the global information of the nodes. They establish route between the source and destination whenever it is required, i.e., on demand [1, 2, 4–6]. The hybrid protocols contain both proactive and reactive characteristics and use clustering to make a network stable and scalable. The proactive strategy is used for intra-clustering routing and the reactive strategy is used for inter-cluster routing [2, 4–6]. The clusters can be organized in a hierarchical manner and the resultant protocols are termed as hierarchical clustering protocols.

In this paper, we propose a heterogeneous network energy model that is characterized by a single parameter. This parameter helps the model defining various levels of node heterogeneity (from zero level to four level) depending on its value. We estimate the network lifetime by using the HEED protocol [5] as it is one of the energy efficient clustering protocol and we call its implementation as HEEDML (HEED MultiLevel). Depending on the heterogeneity HEEDML is denoted by HEEDML-0, HEEDML-1, HEEDML-2, HEEDML-3, and HEEDML-4. The HEEDML-0 assumes all sensor nodes in a WSN to have equal amount of energy, for which the original HEED is implemented. The 1-level, 2-level, 3-level and 4-level heterogeneity assume the sensor nodes in a WSN to be equipped with two, three, four and five levels of energy, respectively. The HEED considers two parameters—residual energy and node density to determine the cluster heads. We also use the same parameters for selecting the cluster heads. We further incorporate fuzzy logic in HEEDML to decide the cluster heads by adding two more parameters viz average energy of the neighbouring nodes and the distance between a sensor and sink, in addition to the residual energy and node density. The resultant HEEDML is denoted by HEEDML-FL. Depending on the heterogeneity, the variants of HEEDML-FL are denoted as HEEDML-FL-0 (original HEED with fuzzy logic), HEEDML-FL-1 (HEEDML-1 with fuzzy logic), HEEDML-FL-2 (HEEDML-2 with fuzzy logic), HEEDML-FL-3 (HEEDML-3 with fuzzy logic), and HEEDML-FL-4 (HEEDML-4 with fuzzy logic). Increasing the level of heterogeneity increases the network lifetime. The increase in network lifetime is much higher than that in the network energy. The fuzzy logic implementation of HEED further increases the network lifetime. For instance, the HEEDML-FL-0 increases the network lifetime by 193.84 % with respect to the original HEED (HEEDML-0) without increasing the network energy. Other performance parameters such as packet delivery, total energy consumption, throughput, average delay, and traffic load have better results. The rest of the paper is organized as follows. Section 2 reviews the related works. Section 3 discusses the fuzzy inference system. In Sect. 4, a multilevel heterogeneous model for WSNs is discussed that is used to simulate the HEEDML-0, HEEDML-1, HEEDML-2, HEEDML-3, HEEDML-4 and HEEDML-FL-0, HEEDML-FL-1, HEEDML-FL-2, HEEDML-FL-3, and HEEDML-FL-4. Section 5 discusses the cluster formulation of the protocols under discussion. In Sect. 6, we discuss the simulation results and, finally, the paper is concluded in Sect. 7.

2 Literature Survey

Many researchers have focused on energy efficient communication because the computation and communication consume more battery power. Heinzelman et al. [7] discuss a low-energy adaptive clustering hierarchy (LEACH) protocol, the first clustering based protocol. It minimizes the overall energy usage by organizing the nodes into clusters. Each cluster contains a specifically designated node called cluster head. The cluster head selected among the sensor nodes by rotation based on their residual energies receives the data from its cluster members and transmits it to the sink. In this protocol, the network energy is used efficiently because only the cluster heads spend their energies in transmitting the data. Manjeshwar et al. [8] discuss a threshold sensitive energy efficient sensor network (TEEN) protocol. It is based on a hierarchical grouping of nodes in which all cluster heads do not send data to the sink, but to the cluster heads nearer to the sink; thus, reducing the energy consumption. The TEEN protocol has been extended in [9] and the

resultant protocol is called as adaptive threshold sensitive energy efficient sensor network (APTEEN) protocol. The APTEEN is meant for capturing data periodic and time-critical events unlike TEEN that is meant only for time-critical events. The main drawbacks of the TEEN and APTEEN protocols are overhead and complexity of forming clusters in multiple levels, implementing threshold-based functions, and dealing with attribute based naming of queries [10]. In [12], the LEACH-C protocol is discussed that combines the energy-efficient cluster-based routing and media access together with application-specific data aggregation to achieve good performance in terms of system lifetime, latency, and application-perceived quality. Other variant of LEACH have been discussed in [5, 11]. An improved version of LEACH, called as power efficient gathering in sensor information systems (PEGASIS) protocol, is discussed in [13] wherein a node sends data to another node forming chains rather than clusters and the node in a chain nearest to the sink sends the data to the sink. It has been extended by forming the chain using binary structure so that the length traversed by a packet is reduced and it is called as Hierarchical-PEGASIS [14]. The above mentioned protocols are discussed for homogenous networks. The stable election protocol (SEP) is the very first protocol, an extension of LEACH, discussed for a two-level heterogeneous network [15]. This protocol selects the cluster heads based on the weighted election probabilities, a function of the remaining energy of nodes to ensure uniform usage of node energy. In [5], the cluster heads have been determined based on the residual energy and node degree unlike in LEACH protocol and the resultant protocol is called as hybrid energy efficient distributed clustering (HEED) protocol. It performs better than the LEACH protocol.

The papers [16, 17] discuss energy efficient clustering scheme (EECS) protocol in which the cluster heads are selected in two phases based on the residual energy. In first phase, called head election phase, a fixed number of nodes are elected that compete for cluster heads based on their residual energies. In second phase, called as cluster formation phase, the cluster heads are chosen among the elected nodes in such a way that the load gets balanced. The paper [18] discusses the distributed energy-efficient clustering (DEEC) protocol by considering two level and multilevel energy heterogeneity. In this protocol, the cluster heads are selected based on the ratio of residual energy of each node and the average energy of the network. In [19], an energy efficient routing method is discussed, called as energy dissipation forecast method (EDFM), in which the cluster heads are selected based on the residual energy and energy consumption rate in all nodes. In [20], the stochastic distributed energy-efficient clustering (SDEEC) protocol is discussed for heterogeneous WSNs. It is based on dividing the network into dynamic clusters. The SDEEC introduces a dynamic method where the cluster head election probability is more efficient. The paper [21] proposes a three level heterogeneity network model and analyzes the network lifetime by considering single hop communication. The paper [22] discusses energy efficient clustering and data aggregation (EECDA) protocol for heterogeneous network as given in [21]. This protocol selects the cluster heads based on the maximum sum of energy residues for data transmission rather than the path with minimum energy consumption. The paper [23] discusses a mechanism for selecting the clusters similar to that discussed in [19], but for three-level heterogeneous WSNs. In [24], the bandwidth efficient heterogeneity aware cluster based data aggregation (BHCDA) protocol is discussed that uses intra and inter-cluster aggregation on the randomly distributed nodes with variable data generation rate while routing the data to sink. It uses the correlation of data within the packet for applying the aggregation function on the data generated by nodes. The paper [25] discusses the balanced energy efficient network integrated super heterogeneous (BEENISH) protocol for the four level of heterogeneity that achieves longer

stability, lifetime and more effective messages than the DEEC and DDEEC. Among these types of protocols, the HEED protocol [5] is one of the most popular clustering protocols because it performs load balancing. We also consider HEED implementation for our proposed heterogenous network model for evaluating the network performance. It is not out of place to briefly discuss the HEED protocol.

The cluster head selection process in the HEED protocol uses two parameters: residual energy as primary parameter and intra-cluster communication cost as the secondary parameter. The primary parameter probabilistically selects an initial set of cluster heads and the secondary parameter is used to break tie among them. A tie occurs when a node falls within the range of more than one cluster heads. The cluster range is determined by the power level used for inter-cluster communication during clustering. Initially, the percentage of cluster heads in HEED are predetermined, C_{prob} (say 5 %), assuming that an optimal percentage cannot be computed a priori. The cluster heads' probability C_{prob} is used to limit the initial cluster heads. It sets the probability of making a node as a cluster head, CH_{prob} , which is given by [5]

$$CH_{prob} = C_{prob} * \frac{E_{residual}}{E_{max}} \tag{1}$$

where $E_{residual}$ and E_{max} are residual and maximum energies of the concerned node, respectively.

The value of CH_{prob} is lower bounded by the threshold p_{min} (i.e., 10^{-4}) and the nodes not covered by any cluster head double their probability of becoming a cluster head. In this way, the cluster heads are selected. All the deployed sensor nodes collect data from the monitoring area and send that data to their respective cluster heads. The cluster heads send the received data to the base station. The data collection is done using multihop communication with data aggregation. We will use the same two parameters to decide the cluster heads as is done in HEED protocol. In case of fuzzy implementation, we consider two additional parameters, i.e., average energy and distance between nodes and sink. In [26], the HEED protocol is discussed for a three-level heterogeneous network model and this implementation of HEED has been referred as Heterogenous HEED protocol. The

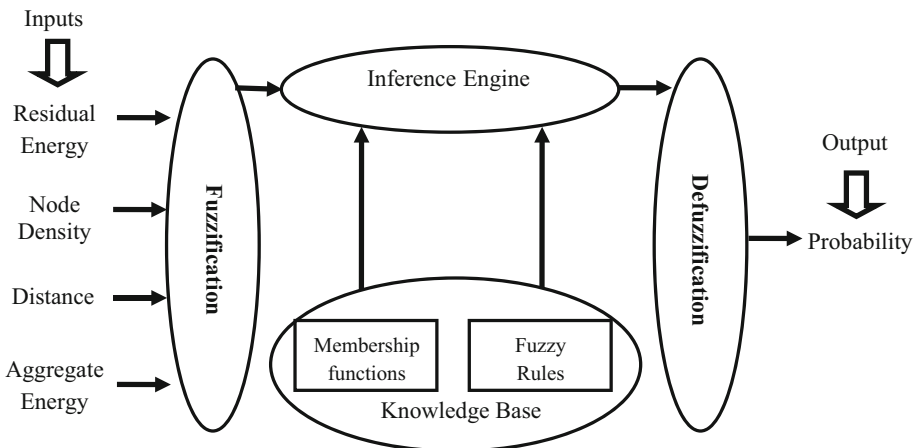


Fig. 1 Fuzzy logic based system

papers [27–29] also discuss different three-level heterogeneous models only, which cannot be enhanced to define more 3-level of heterogeneity. In next section, we discuss fuzzy system for finding the cluster heads and that associated cluster members.

3 Fuzzy Inference System

The fuzzy logic system is a nonlinear mapping of an input data vector into a scalar output. The input and output parameters are the linguistic variables. The number and complexity of rules depend on the number of input parameters and their fuzzy set. Once the rules have been formulated, the fuzzy inference engine combines them by the aggregation process to construct a single fuzzy set for output variable. It is an efficient tool for handling the problems with uncertainties and imprecise information. The fuzzy logic can be applied in several domains like approximate reasoning, pattern recognition, medical computing, robotics, optimization industrial engineering, etc. Figure 1 shows the basic structure of a fuzzy logic system (FLS) that consists of fuzzifier, decision making, and defuzzifiers [30].

Fuzzifier: It converts a crisp value into degree of membership by applying the corresponding membership functions. A membership function determines the certainty with which the crisp values are associated with a specific linguistic value. The membership functions can have different shapes. The commonly used shapes for membership functions are triangular, trapezoidal, and Gaussian shaped functions.

Decision making: In this process, the decisions are made based on the rules that consist of a set of linguistic statements stored in a rule-base. The rules are of the form *IF premise, THEN consequent* where the *premise* is composed of fuzzy input variables connected by logical functions (AND, OR, NOT) and the *consequent* is a fuzzy output variable. Consider a q -input 1-output FLS with rules of the form

$$R^i : IF V_1 \text{ is } M_1^i \text{ and } V_2 \text{ is } M_2^i \text{ and } V_3 \text{ is } M_3^i \text{ and } \dots \text{ and } V_q \text{ is } M_q^i \text{ THEN output is } O^i$$

For input $V' = \{V'_1, V'_2, V'_3, \dots, V'_q\}$, the degree of rule R^i provides the output

$$\mu_{M_1^i}(V'_1) * \mu_{M_2^i}(V'_2) * \mu_{M_3^i}(V'_3) * \dots * \mu_{M_q^i}(V'_q) = T_i^q \mu_{M_1^i}(V'_1)$$

Here, μ represents the membership function and the $*$ and T indicate the chosen triangular and trapezoidal norms. These norms are the binary operations applied to the fuzzy sets for the membership functions.

Defuzzification: It is a process that converts a fuzzy set into a crisp value. Defuzzification is used in fuzzy modeling and fuzzy logic control to convert the fuzzy outputs to the crisp values. There are numerous techniques for defuzzifying a fuzzy set: center of gravity, center of singleton, and maximum methods. The center of gravity approach finds the centroid of the shape obtained by superimposing the shapes resulted from the application of rules and the output of the defuzzifier is the X-coordinate of this centroid. In center of singleton method, the membership functions for each rule are defuzzified separately. Each membership function is reduced to a singleton which represents the function's center of gravity. The singletons can be determined while designing the system. The class of maximum methods determines the output by selecting the maximum value of the membership function. Using these methods, the rule with the maximum activity always determines the output value. The defuzzifier calculates the centroid that is used to calculate the probability. The centroid is computed as follows [30]:

$$Centroid = \frac{\sum \mu_{\eta}(a) * a}{\sum \mu_{\eta}(a)} \tag{2}$$

where, $\mu_{\eta}(a)$ denotes the membership function of set η .

In this paper, the cluster heads are elected in each iteration of each round by calculating the probability that is computed by using four fuzzy variables: residual energy of each node, node density with respect to its cluster, distance between sensor and sink, and the average energy of the nodes in the sensing range. The definitions of the fuzzy parameters are as follows:

Residual Energy: The node generally spends its energy in sensing the environment (collecting data) and sending it to some other node. The remaining energy with the node is called its residual energy.

Node Density: The node density of a node refers to the number of sensors within its sensing range.

Distance: It is defined as the Euclidean distance between the given sensor node and the sink.

Average Energy: It is given by the sum of energies of all nodes in sensing range after dividing by their numbers.

We use the Mamdani model for fuzzy inference system because it is simple and most widely used in literature. We consider four input parameters to decide a given node as

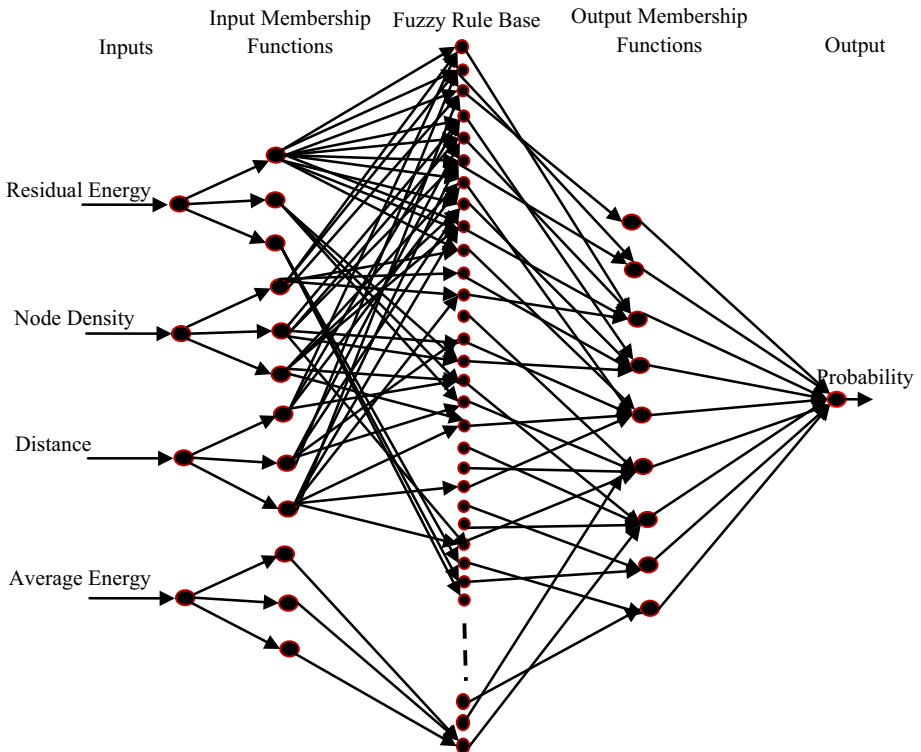


Fig. 2 Layered fuzzy scheme

cluster head. These parameters include the residual energy, node density, distance, and average energy of the node. The residual energy has linguistic values as low, medium, and high; the node density has sparsely, medium, and densely; the distance has near, medium, and far; and the average energy has less, average and elevate. The output variable has nine values—very weak, weak, little weak, lower medium, medium, higher medium, little strong, strong, very strong. The output variable helps deciding the node as a cluster head. The layered fuzzy scheme as shown in Fig. 2 consists of four input linguistic variables: residual energy, node density, distance and average energy at layer 1. Layer 2 contains the input membership functions and their values and at layer 3, fuzzy rules and knowledge base are there. The layer 4 consists of output membership functions and the layer 5 signifies output linguistic value to determine the probability for deciding the node as a cluster head.

The input membership functions for the residual energy, node density, distance, and average energy consist of two half trapezoidal and one triangular, and for output probability, two half trapezoidal and seven triangular, respectively, as shown in Fig. 3a–e.

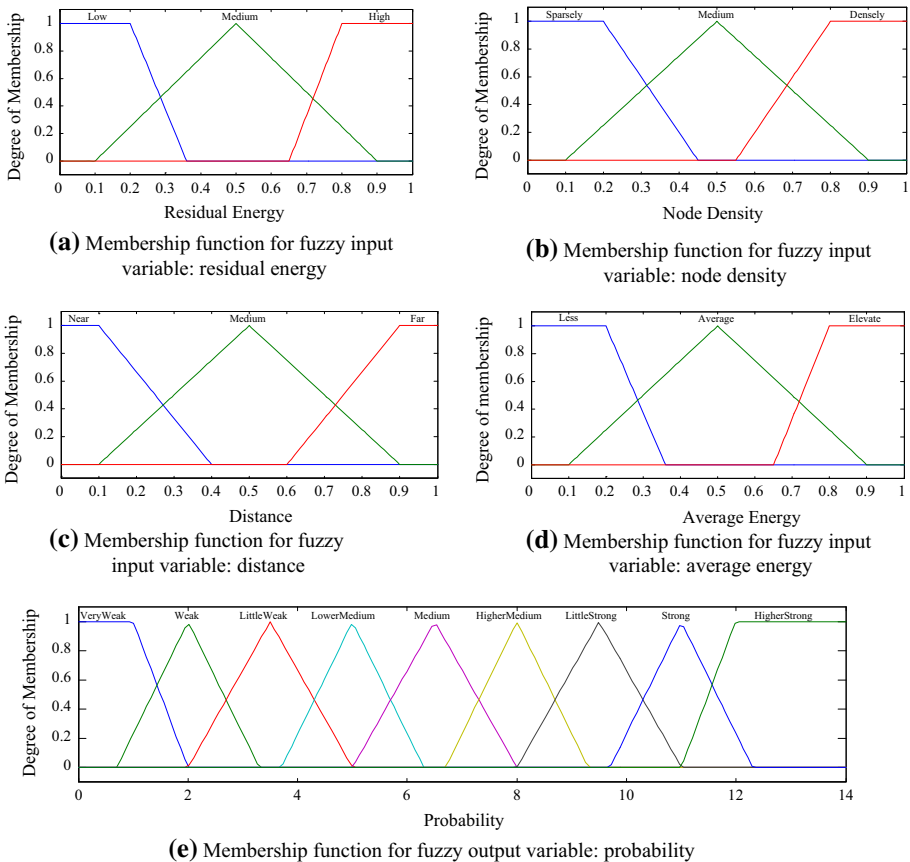


Fig. 3 Membership functions corresponding to fuzzy inputs and output parameters. **a** Membership function for fuzzy input variable: residual energy, **b** Membership function for fuzzy input variable: node density, **c** Membership function for fuzzy input variable: distance, **d** Membership function for fuzzy input variable: average energy, **e** Membership function for fuzzy output variable: probability

After defining the layered fuzzy scheme and membership functions, we discuss our proposed multi-level heterogeneous network model.

4 Multilevel Heterogeneous Network Model

We consider the following assumptions for a WSN in our work:

- Initially all nodes have the same capabilities but different in terms of energies.
- Nodes are stationary after deployment; each is identified by a unique ID.
- Nodes in network can be homogeneous or heterogeneous, but not chargeable.
- Nodes are location-aware, i.e. they are equipped with GPS-capable antenna.
- Radio link is symmetric.
- There is only one stationary sink located at the centre in network that has a constant power supply; thus has no energy, memory and computation constraints.
- Distance between nodes can be computed based on the received signal strength.
- Nodes have the capability of controlling the transmission power according to the distance of receiving nodes.
- Nodes have ability to aggregate data and the node failure can be due to energy depletion.

We discuss our multilevel heterogeneous network model. In literature, most of the existing models [16–18, 21–25] consider either two-level or three-level heterogeneity as multilevel heterogeneity. We here consider five-level heterogeneity in our model. The model is general enough to describe 0-level, 1-level, 2-level, 3-level, and 4-level heterogeneity depending on the value of the model parameter. We consider N as the total number of nodes in our network. Depending on the level of heterogeneity, N is divided into different numbers. The general form of the energy model of network is defined as follows:

$$N * \left[E_0 * \sin\left(\frac{i}{3}\right) + E_1 * \sin\left(\frac{i}{4}\right) + E_2 * \sin\left(\frac{i}{2}\right) + E_3 * \sin(3i) + E_4 * \sin(2i) \right] \quad (3)$$

Here $E_0, E_1, E_2, E_3,$ and E_4 denote the initial energies of type-0, type-1, type-2, type-3 and type-4 nodes, respectively. The model parameter is ' i ' that determines the heterogeneity level of the network. For $i = 8\pi$, there is only one non-zero term, i.e., $E_0 * \sin\left(\frac{i}{3}\right)$. Thus, there are only one type of nodes in the network and the model describes zero-level heterogeneity i.e., homogenous network. The number of nodes in zero-level heterogeneous network, denoted by N_0 , is determined by

$$N_0 = \frac{N * \sin\left(\frac{i}{3}\right)}{\sin\left(\frac{i}{4}\right) + \sin\left(\frac{i}{3}\right) + \sin\left(\frac{i}{2}\right) + \sin(2i) + \sin(3i)} \quad (4)$$

For $i = 2\pi$, there are only two non-zero terms, i.e., $E_0 * \sin\left(\frac{i}{3}\right)$ and $E_1 * \sin\left(\frac{i}{4}\right)$. Thus, there are only two types of nodes in the network and the model describes one-level heterogeneous network. The numbers of nodes in one-level heterogeneous network are denoted by N_0 and N_1 . The value of N_0 is defined in (4) and that of N_1 are given by

$$N_1 = \frac{N * \sin\left(\frac{i}{4}\right)}{\sin\left(\frac{i}{4}\right) + \sin\left(\frac{i}{3}\right) + \sin\left(\frac{i}{2}\right) + \sin(2i) + \sin(3i)} \quad (5)$$

For $i = \pi$, there is only three non-zero terms, i.e., $E_0 * \sin(\frac{i}{3})$, $E_1 * \sin(\frac{i}{4})$ and $E_2 * \sin(\frac{i}{2})$. Thus, there are only three types of nodes in the network and the model describes two-level heterogeneous network. The numbers of zero-level and one-level nodes of the network are defined in (4) and (5), respectively. The number of level-2 nodes, denoted by N_2 , is given by

$$N_2 = \frac{N * \sin(\frac{i}{2})}{\sin(\frac{i}{4}) + \sin(\frac{i}{3}) + \sin(\frac{i}{2}) + \sin(2i) + \sin(3i)} \quad (6)$$

For $i = 3\pi/2$, there are four non-zero terms, i.e., $E_0 * \sin(\frac{i}{3})$, $E_1 * \sin(\frac{i}{4})$, $E_2 * \sin(\frac{i}{2})$ and $E_3 * \sin(3i)$. Thus, there are four types of nodes in the network and the model describes three-level heterogeneous network. The numbers of zero-level, one-level, and two-level nodes of the network are defined in (4)–(6) respectively. The number of level-3 nodes, denoted by N_3 , is given by

$$N_3 = \frac{N * \sin(3i)}{\sin(\frac{i}{4}) + \sin(\frac{i}{3}) + \sin(\frac{i}{2}) + \sin(2i) + \sin(3i)} \quad (7)$$

For $i = \pi/4$, there are all five terms non-zero. Thus, the model describes four-level heterogeneous network. The numbers of zero-level, one-level, two-level and three-level nodes of the network are defined in (4)–(7) respectively. The number of level-4 nodes, denoted by N_4 , is given by

$$N_4 = \frac{N * \sin(2i)}{\sin(\frac{i}{4}) + \sin(\frac{i}{3}) + \sin(\frac{i}{2}) + \sin(2i) + \sin(3i)} \quad (8)$$

The total energy of the network is given by

$$E_{total} = N * \left[E_0 * \sin\left(\frac{i}{3}\right) + E_1 * \sin\left(\frac{i}{4}\right) + E_2 * \sin\left(\frac{i}{2}\right) + E_3 * \sin(3i) + E_4 * \sin(2i) \right] \quad (9)$$

The energies of different types of nodes are related by the following:

$$E_j = E_0 * (1 + j * \beta)$$

This relation simply tells that the energy of a type j node is β times more than that of a type $j-1$ node, β is a constant. The energies must satisfy the inequalities $E_0 < E_1 < E_2 < E_3 < E_4$.

In this way, we have shown that the energy model in (9) can describe 0-level, 1-level, 2-level, 3-level and, 4-level heterogeneity in WSNs. Now we discuss the cluster formation and data transmission.

5 Cluster Formation and Data Transmission

Here, we generate energy efficient clusters for randomly deployed heterogeneous sensor nodes. In this section, formation of clusters, data collection, data aggregation, and its transmission to the sink are discussed. In our work, the sensor nodes are organized in groups, each having a designated node. These groups are called clusters and the designated

node is called the cluster head. We discuss the process of cluster formation and their head selection, energy spent in data collection and its transmission by cluster members and the cluster head. The cluster head receives the data from its cluster members and transmits it after aggregation to the sink.

We have used the wireless radio model [10, 12] for the radio hardware energy dissipation. In this model, the radio dissipates ϵ_{elec} joules of energy (energy consumed in the electronics circuit) to run the transmitter or receiver circuitry. The amount of energy (in joules) per bit per meter² dissipated in the transmitter amplifier for long and short distances are ϵ_{mp} and ϵ_{fs} , respectively. The energy consumed (E_{TXL}) to transmit an L-bit message for a long distance $d(d > d_0)$ is given by (10) and the energy consumed (E_{TXS}) for a short distance $d(d \leq d_0)$ is given by (9).

$$E_{TXL} = L * \epsilon_{elec} + L * \epsilon_{mp} * d^4 \quad \text{if } d > d_0 \tag{10}$$

$$E_{TXS} = L * \epsilon_{elec} + L * \epsilon_{fs} * d^2 \quad \text{if } d \leq d_0 \tag{11}$$

The distance d_0 , called as *threshold*, is given by [10, 12]

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} = \sqrt{\frac{10 * 10^{-12}}{0.0013 * 10^{-12}}} = 87.70 \tag{12}$$

This value of d_0 i.e., 87.70 is the maximum and in literature different values such as 70, 85, etc. have been used.

We now discuss how the energy is consumed in a network. Consider a network having total n nodes from which k nodes have been designated as cluster heads. The cluster members spend their energy in sensing the environment and sending the sensed information to their cluster heads. The cluster heads spend their energy in aggregating the received information from their cluster members and sending the aggregated data to the sink. The process of data collection by sensors, its transmission to their cluster heads, data aggregation by cluster heads, and its transmission to the sink comprises as one iteration. In each iteration, new nodes among the nodes that have not been cluster heads in past iterations are considered for cluster heads. The data sent to the sink in an iteration is referred to one frame. A cluster member spends its energy in collecting L bit data in each iteration and it is given by

$$E_{cm} = L * \epsilon_{elec} + L * \epsilon_{fs} * d^2 \tag{13}$$

A cluster head spends its energy in each iteration in aggregating the data received from its cluster members and sending it to the sink that is given by

$$E_{ch} = L * \epsilon_{elec} \left(\frac{n}{k} - 1\right) + L * \epsilon_{DA} * \left(\frac{n}{k} - 1\right) + L * \epsilon_{elec} + L * \epsilon_{mp} * d^4 \tag{14}$$

Each cluster has n/k sensor nodes including the cluster head. Thus, the first term in (14) gives the energy consumed in receiving the data from $\left(\frac{n}{k} - 1\right)$ cluster members and the second term refers to the energy consumed in data aggregation. The last two terms refer to the energy consumed in transmission of the message to the sink. The number of iterations when all sensors have become cluster heads at some point of time constitutes a round. An iteration is defined as the process of collecting the data from cluster members and sending it to the sink by a cluster head. It consists of election and data transfer phases that are described below.

5.1 Cluster Head Election Mechanism

Clustering mechanism helps reducing the energy consumption that in turn increases the network lifetime. We discuss cluster head selection for fuzzy and non-fuzzy implementation. Initially some percent of nodes, say 5 %, are selected as cluster heads in such a way that none should lie in the sensing range of another. This condition ensures that the cluster heads are well distributed [5]. These nodes broadcast an advertisement message. The non-head nodes join one of the cluster heads by reciprocating their desire to their intended cluster heads depending on the received signal energy. If a node receives the same amount of energy from two cluster heads, then this node will join the cluster head that has lower node density. In this way, the cluster heads form their clusters. If the entire area is not covered by the clusters, then some nodes are randomly selected as cluster heads from the uncovered area and they form their clusters. In that way, the entire area is covered by the clusters.

5.1.1 Election Phase for Non-Fuzzy Implementation

For deciding the cluster heads in case of non-fuzzy implementation, we estimate the residual energy of all nodes and their node density using the same approach as discussed in HEED protocol. In actual we need not to estimate the residual energy of a node, it can be obtained from its energy by subtracting the energy spent in the activity entrusted upon it. The node density of a node means the number of nodes in its sensing range. A node joins the cluster head that has minimum degree in order to distribute load among the cluster heads.

5.1.2 Election Phase for Fuzzy Implementation

In this phase, the decision of cluster heads selection is based on the residual energy of all nodes, node density, average energy, and distance of nodes from the sink. The selection of a cluster head (or rotating cluster head) amongst the nodes is based on the following probability:

$$Probability = \frac{a * L_{re} + b * L_{nd} + c * L_{ae} + d * (M_d - L_d)}{a * M_{re} + b * M_{nd} + c * M_{ae} + d * M_d} \quad (15)$$

where a, b, c and d are weights of residual energy, node density, average energy, and distance, respectively. L_{re} , L_{nd} , L_{ae} , & L_d , and M_{re} , M_{nd} , M_{ae} , & M_d denote the level values and maximum value of their levels for residual energy, node density, average energy, and distance, respectively. In our work, the residual energy has level values as 0 (low), 1 (medium), and 2 (high). The node density has level values as 0 (sparsely), 1 (medium), and 2 (densely). The average energy has level values as 0 (less), 1 (average), and 2 (elevated). The distance has level values as 0 (near), 1 (medium), and 2 (far). In literature, the residual energy has been given more weightage; therefore, the values of a, b, c, and d have been taken as 2, 1, 1, and 1, respectively. The prespecified number of nodes that have maximum probabilities are selected as cluster heads. Once the cluster heads are decided, each cluster head broadcasts a short range advertisement and the sensor nodes may receive advertisements from one or more cluster heads. Each sensor node chooses its cluster head on the basis of the received signal strength of the advertisements. The sensor nodes transmit short range acknowledgments to inform their cluster heads about their decision. At this stage the

clusters for the current round are determined. Each cluster head creates a TDMA schedule and broadcasts it to its members. At the end of election phase, each cluster member checks if it has sufficient energy for next round. If the energy of any cluster member goes to zero, it is removed and the remaining cluster members update their schedules accordingly. Figure 4 shows an instance of clusters formed in multilevel heterogeneity for non-fuzzy implementation. In this figure, the type-0, type-1, type-2, type-3 and type-4 nodes have been denoted by circular (O), plus (+), star (*), dollar (\$) and at the rate (@) signs, respectively. The sink has been marked as X, which is situated at the center of the area. The members of a cluster including cluster head are shown by the same color. In case of fuzzy implementation, similar types of clusters are formed.

5.2 Data Transfer Phase

During data transmission phase, the cluster member nodes collect the data and transmit it to their cluster heads. After receiving the data from all cluster members, the cluster heads aggregate the data and route the same to the sink. This forms an iteration in which the cluster members spend their energy in collecting the data and sending it to their cluster heads and the cluster heads spend their energy in aggregating the received data and sending it to the sink. In next iteration, the cluster heads are selected from the remaining nodes that have not been made cluster heads in past iterations of the current round. This process continues for several rounds till all the nodes have not depleted their energy.

6 Results and Discussions

In this section, we discuss the simulation results of the HEED protocol implementation for our proposed multilevel heterogeneity network model and call this implementation as HEEDML. We have shown in Sect. 3 that our network model can define 0-level (homogenous), 1-level, 2-level, 3-level, and 4-level heterogeneity of the WSNs and

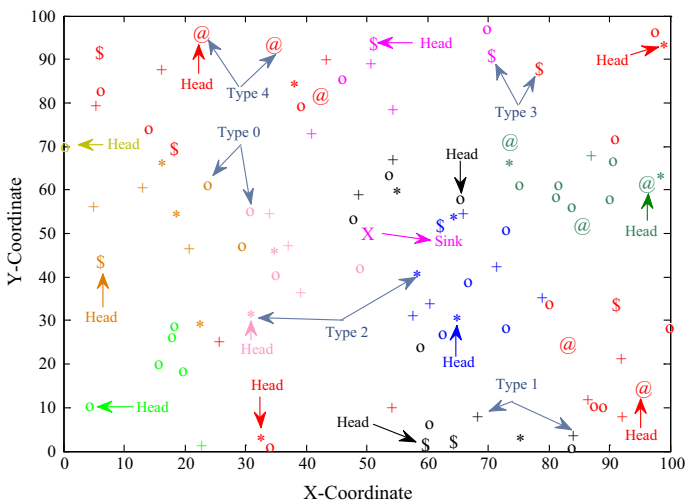


Fig. 4 Clusters with their cluster heads shown in different colors

accordingly we denote them as HEEDML-0, HEEDML-1, HEEDML-2, HEEDML-3, and HEEDML-4 heterogeneity, respectively. In original HEED protocol, the probability for selecting a cluster head has been calculated based on the residual energy and node density. We use same parameters in HEEDML for cluster head selection. We also discuss the fuzzy implementation of the HEEDML and use two additional parameters—distance and average energy to calculate the probability for cluster head selection. The fuzzy implementations of HEEDML are denoted as HEEDML-FL-0, HEEDML-FL-1, HEEDML-FL-2, HEEDML-FL-3, and HEEDML-FL-4, respectively. We compute the number of nodes alive, number of packets delivered to sink, energy consumed, energy consumption, throughput, traffic load, and aggregate delay to evaluate the performance of the proposed protocol. The number of nodes alive refers to the instantaneous measure of the total number of nodes that have not yet expended all of their energy. The data packet delivery refers to the total number of data packets delivered to the sink. This is also a measure of amount of information sent to the sink from the sensor field. The energy consumption is the measure of the instantaneous amount of energy consumed in the network in a round. This is simply the energy difference from the beginning till the end of the round. The throughput is defined as the number of bits transmitted per unit time. The traffic load is defined as total number bits transmitted/sent to the sink and the aggregate delay is defined as the ratio of total delay to the total packets transmitted.

In our simulations, we have used MATLAB by considering random deployment of 100 nodes in a square field of area $100 \text{ M} \times 100 \text{ M}$. Our network model is characterized by the parameter i (the argument of sin functions in (9)) that helps in defining different levels of heterogeneity. For $i = \pi/4, 3\pi/2, \pi, 2\pi$, and 8π , the model (9) defines the four-level, three-level, two-level, one-level, and zero-level heterogeneity, respectively, and the corresponding HEED implementation is referred as HEEDML-4, HEEDML-3, HEEDML-2, HEEDML-1 and HEEDML-0. The HEEDML with fuzzy logic, i.e., HEEDML-FL-0,

Table 1 Simulation parameters

Description	Symbol	Value
No of sensors	N	100
Sink position	S_p	(50, 50)
Threshold distance	d_0	70 m
Cluster radius	C_r	25 m
Energy consumed by the amplifier to transmit at a shorter distance	ϵ_{fs}	10 nJ/bit/m ²
Energy consumed by the amplifier to transmit at a longer distance	ϵ_{mp}	0.0013 pJ/bit/m ⁴
Energy consumed in the electronics circuit to transmit or receive the signal	E_{elec}	50 nJ/bit
Energy for data aggregation	ϵ_{DA}	5 nJ/bit/signal
Message size	L	4000 bits
Initial energy	E_0	0.5 J
Energy constant	β	0.2
Simulation time	S_t	900 s
Packet size	P_s	512 bits
Bandwidth	bd	1 Mbps
Model parameter	i	$8\pi, 2\pi, \pi, 3\pi/2, \& \pi/4$

Table 2 Distribution of sensor nodes in multilevel heterogeneity for both fuzzy and non-fuzzy implementation of HEED

Protocols	HEEDML-0	HEEDML-1	HEEDML-2	HEEDML-3	HEEDML-4	HEEDML-FL-0	HEEDML-FL-1	HEEDML-FL-2	HEEDML-FL-3	HEEDML-FL-4
Total no of sensors	100	100	100	100	100	100	100	100	100	100
Type 0 nodes	100	54	40	39	39	100	54	40	39	39
Type 1 nodes	NA	46	33	27	28	NA	46	33	27	28
Type 2 nodes	NA	NA	27	20	15	NA	NA	27	20	15
Type 3 nodes	NA	NA	NA	14	10	NA	NA	NA	14	10
Type 4 nodes	NA	NA	NA	NA	8	NA	NA	NA	NA	8

NA: not applicable

Table 3 Categorization of energies in multilevel heterogeneity for both fuzzy and non-fuzzy implementation of heed

Protocols	HEEDML-0	HEEDML-1	HEEDML-2	HEEDML-3	HEEDML-4	HEEDML-FL-0	HEEDML-FL-1	HEEDML-FL-2	HEEDML-FL-3	HEEDML-FL-4
Energy of type 0 nodes	0.5 J	0.5 J	0.5 J	0.5 J	0.5 J	0.5 J	0.5 J	0.5 J	0.5 J	0.5 J
Energy of type 1 nodes	NA	0.6 J	0.6 J	0.6 J	0.6 J	NA	0.6 J	0.6 J	0.6 J	0.6 J
Energy of type 2 nodes	NA	NA	0.7 J	0.7 J	0.7 J	NA	NA	0.7 J	0.7 J	0.7 J
Energy of type 3 nodes	NA	NA	NA	0.8 J	0.8 J	NA	NA	NA	0.8 J	0.8 J
Energy of type 4 nodes	NA	NA	NA	NA	0.9 J	NA	NA	NA	NA	0.9 J

HEEDML-FL-1, HEEDML-FL-2, HEEDML-FL-3, and HEEDML-FL-4 use the same simulation environment as that used for HEEDML. The nodes in the network, which are 100, are assumed to be of type-0 for HEEDML-0 and HEEDML-FL-0. These numbers have been obtained from (4) to (8). For HEEDML-1 and HEEDML-FL-1, the numbers of type-0 and type-1 nodes are 54 and 46, respectively. For HEEDML-2 and HEEDML-FL-2, the number of type-0, type-1, and type-2 nodes are 40, 33, and 27, respectively. For HEEDML-3 and HEEDML-FL-3, the number of type-0, type-1, type-2, and type-3 nodes are 39, 27, 20, and 14, respectively. For HEEDML-4 and HEEDML-FL-4, the number of type-0, type-1, type-2, type-3, and type-4 nodes are 39, 28, 15, 10 and 8, respectively. Our model indeed defines all types of nodes in all cases, but depending upon the value of model parameter, nodes of some levels are non-zero and the remaining ones are zero. For example, in case of level-0 heterogeneity, the numbers of all type nodes except type -0 nodes are zero and thus the numbers of type-0 nodes are 100. The energy levels for type-0, type-1, type-2, type-3, and type-4 nodes are taken as 0.5 J, 0.6 J, 0.7 J, 0.8 J, and 0.9 J, respectively, in both fuzzy and without fuzzy implementation of HEEDML. The input parameters used in our simulations are summarized in Table 1. The categorization of number of sensor nodes and their energies used are summarized in Tables 2 and 3, respectively.

We have computed the number of alive nodes with respect to the number of rounds for different levels of heterogeneity using fuzzy and without fuzzy logic in terms of the first node dead and the last node dead, as shown in Fig. 5. As the level of heterogeneity increases, the network lifetime in terms of alive nodes increases. Same is the case for fuzzy implementation as shown in Fig. 5. It may however be noted that the fuzzy implementation has better performance as compared to the non-fuzzy implementation, for example, HEEDML-FL-0 performs better than the HEEDML-3 as evident from Fig. 5. The performance of the HEEDML-FL-1 is better than that of all variants of the HEEDML in spite

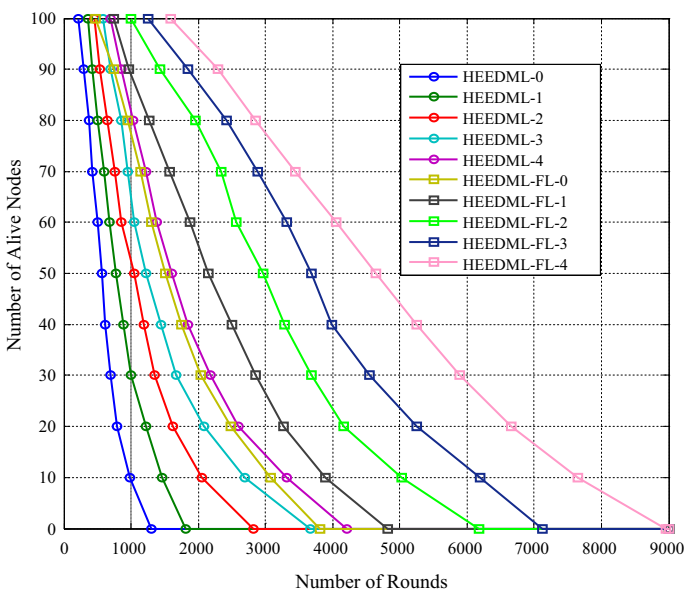


Fig. 5 Lifetime of the network in terms of number of nodes alive versus number of rounds

of the fact that the related network energy is higher than that of the HEEDML-FL-1. The network energy for both HEEDML-1 and HEEDML-FL-1 are same, but the HEEDML-FL-1 provides much longer network lifetime. Among all these, the HEEDML-FL-4 performs the best as far as the number of alive nodes is concerned.

We have also obtained the network lifetime in terms of ‘first node dead’ and ‘last node dead’ as shown in Table 4. It is evident from Table 4 that increasing the level of heterogeneity increases the network lifetime in all cases. For example, in HEEDML-0 and HEEDML-FL-0, the first node dead occurs in 213th and 476th round and the last node dead occurs in 1300th and 3820th round, respectively; in HEEDML-4 and HEEDML-FL-4, the first node dead occurs in 698th and 1595th round and the last node dead occurs in 4209th and 8958th round, respectively.

Table 5 shows the total network energy and the network lifetime with respect to the original HEED or HEEDML-0 for HEEDML-1, HEEDML-2, HEEDML-3, HEEDML-4, HEEDML-FL-0, HEEDML-FL-1, HEEDML-FL-2, HEEDML-FL-3, and, HEEDML-FL-4. This table also has the percentage increase in the network energy and the percentage increase in the network lifetime for all variants of HEEDML. The 18.10 % increase in energy gives 140.86 and 376.43 % increase in the lifetime for non-fuzzy and fuzzy implementation, respectively.

We have computed the number of packets delivered to the sink with respect to the number of rounds for different levels of heterogeneity using fuzzy and without fuzzy logic as shown in Fig. 6. It is clear from figure that the HEEDML-FL-4 has more numbers of packet delivered at the sink in comparison to all other variants. The number of packets transferred to the sink in HEEDML-0, HEEDML-1, HEEDML-2, HEEDML-3, HEEDML-4, HEEDML-FL-0, HEEDML-FL-1, HEEDML-FL-2, HEEDML-FL-3, and HEEDML-FL-4 have been obtained as 1.2×10^4 , 1.8×10^4 , 2.3×10^4 , 2.8×10^4 , 3.7×10^4 , 3.2×10^4 , 4.1×10^4 , 5.4×10^4 , 6.1×10^4 and 7.3×10^4 , respectively, with respect to the number of rounds. We observe that the HEEDML-FL has more number of data packets delivered to the sink than that of the HEEDML using the same amount of energy and the same heterogeneity level.

Figure 7 shows the total remaining energy with respect to the number of rounds. Here the initial energies of the type-0, type-1, type-2, type-3, and type-4 are 50, 54.6, 58.7, 60.9 and 62.0 J for, respectively (for both fuzzy and non-fuzzy implementation of the HEEDML). The HEEDML-FL-0 performs better than the HEEDML-0, HEEDML-1, HEEDML-2, and HEEDML-3. The HEEDML-FL-1 performs better than all levels of the

Table 4 Number of rounds when first and last nodes are dead for all variants of HEEDML

Protocols	First node dead	Last node dead
HEEDML-0	213	1300
HEEDML-1	355	1815
HEEDML-2	458	2826
HEEDML-3	580	3675
HEEDML-4	698	4209
HEEDML-FL-0	476	3820
HEEDML-FL-1	746	4814
HEEDML-FL-2	1007	6186
HEEDML-FL-3	1258	7128
HEEDML-FL-4	1595	8958

Table 5 Percentage increase in network energy and corresponding increase in network lifetime for HEEDML and HEEDML with fuzzy logic of heed

Protocols	HEEDML-0	HEEDML-1	HEEDML-2	HEEDML-3	HEEDML-4	HEEDML-FL-0	HEEDML-FL-1	HEEDML-FL-2	HEEDML-FL-3	HEEDML-FL-4
Total network energy	50 J	54.6 J	58.7 J	60.9 J	62 J	50 J	54.6 J	58.7 J	60.9 J	62 J
Lifetime	1300	1815	2826	3675	4209	3820	4814	6186	7128	8958
% Increase in energy	NA	9.20 %	17.40 %	21.80 %	24 %	0 %	9.20 %	17.40 %	21.80 %	24 %
% Increase in lifetime	NA	39.61 %	117.38 %	182.69 %	223.77 %	193.84 %	270.31 %	375.84 %	448.30 %	589.07 %

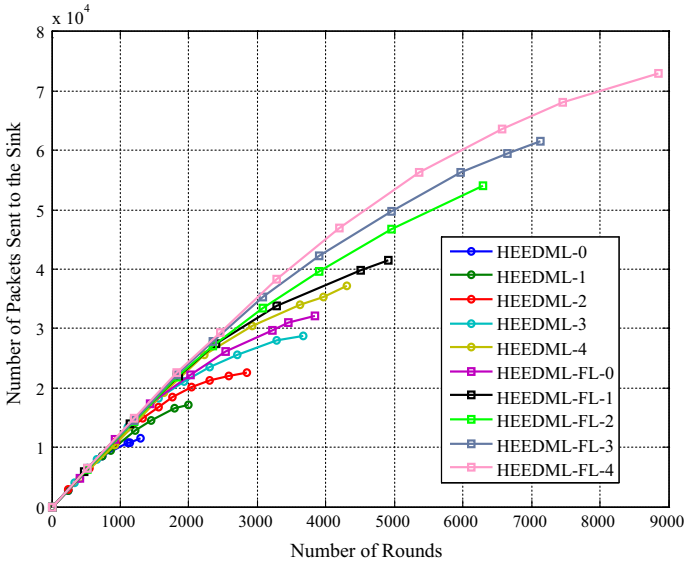


Fig. 6 Number of packets sent to the sink versus number of rounds

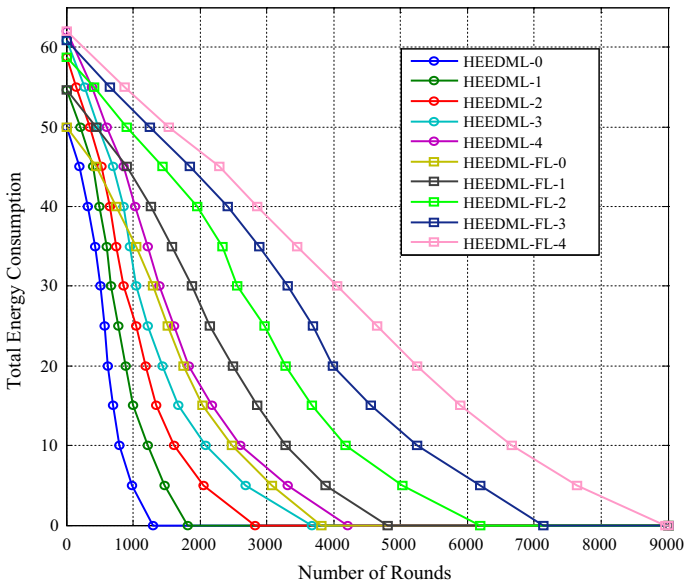


Fig. 7 Total energy consumption versus number of rounds

HEEDML in spite of the fact that the HEEDML-FL-1 has less network energy than that of the HEEDML-4. Thus, the rate of energy dissipation is much slower in case the HEEDML with fuzzy logic than the HEEDML for all levels of heterogeneity.

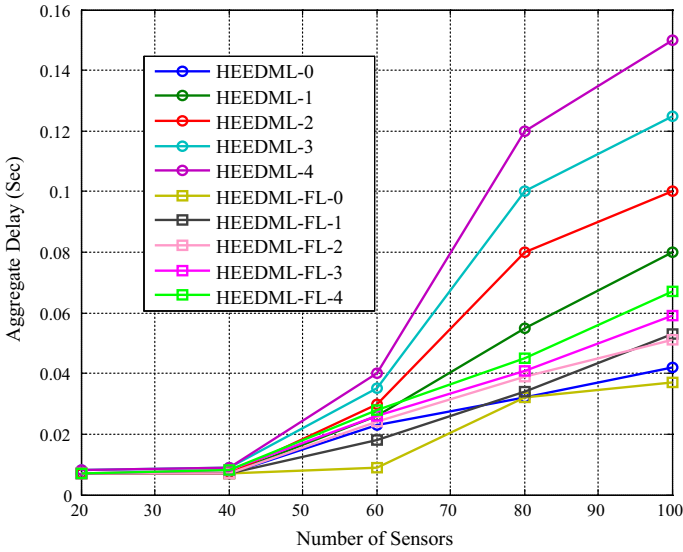


Fig. 8 Aggregate delay versus number of sensor nodes

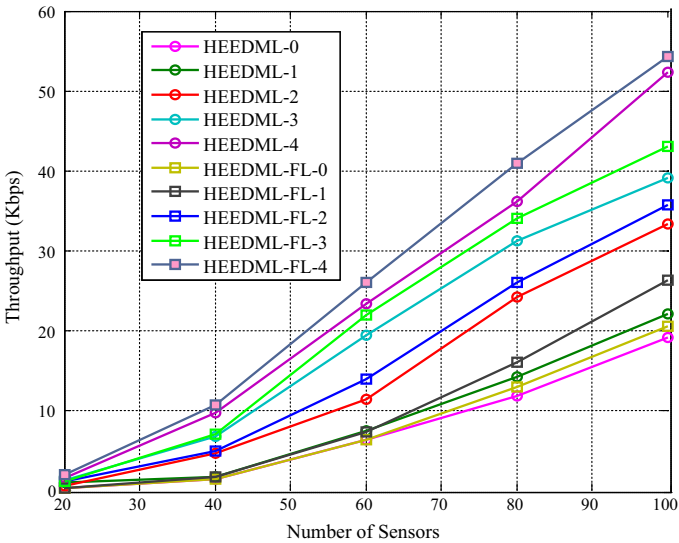


Fig. 9 Throughput versus number of sensor nodes

The aggregate delay, throughput, and traffic load are shown in Figs. 8, 9, and 10, respectively. These simulation results have been taken at an instance of a round. The aggregate delay lies in a very small range in case of HEEDML-FL as compared to the HEEDML (refer Fig. 8). We observe from Fig. 9 that increasing the number of sensors increases the throughput of the network. Furthermore, as the level of heterogeneity increases, the throughput also increases. We can see that the HEEDML with fuzzy logic

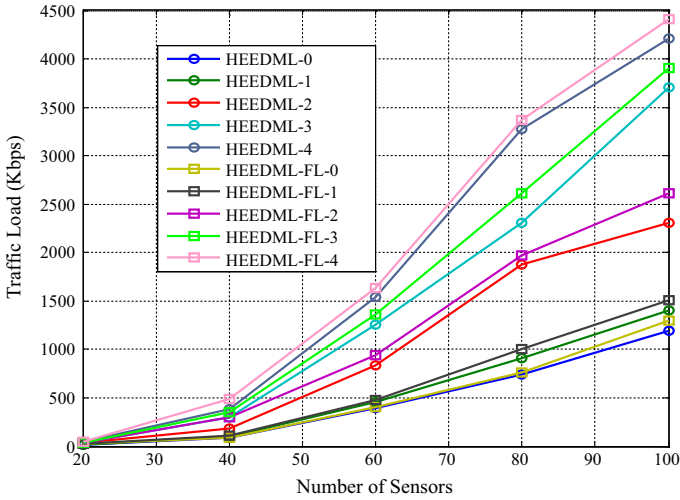


Fig. 10 Traffic load versus number of sensor nodes

gives better performance than the HEEDML for same value of i (level of heterogeneity) and the residual energy. The traffic load increases as the number of sensors increases as shown in Fig. 10. It has similar behavior in fuzzy and non-fuzzy implementations.

The performance of some of the protocols have been computed by using the heterogeneous networks and compared with our proposed protocols. As evident from Table 6, our protocols perform much better than other protocols by taking equal amount of energy in the case of fuzzy and non-fuzzy implementation

Table 6 Number of rounds for our proposed and existing heterogeneity models for given number of sensors with their network energies in WSNs

Protocols	No. of sensors	Total energy of the network	Type of nodes	No. of rounds (network lifetime)
DEEC [18], SEP [15]	100	54.60 J	Two types	1459
HEEDML-1	100	54.60 J	Two types	1815
HEEDML-FL-1	100	54.60 J	Two types	4814
EEHC [16]	100	58.70 J	Three-types	1621
DSCHE [23]	100	58.70 J	Three-types	1968
HEEDML-2	100	58.70 J	Three-types	2826
HEEDML-FL-2	100	58.70 J	Three-types	6186
BEENISH [25]	100	60.90 J	Four-types	2159
HEEDML-3	100	60.90 J	Four-types	3675
HEEDML-FL-3	100	60.90 J	Four-types	7128

7 Conclusion

In this paper, we have discussed a multilevel network model for heterogeneous wireless sensor networks. In this work, we have incorporated five levels of heterogeneity, namely: 0-level, 1-level, 2-level, 3-level, and 4-level heterogeneity in terms of the energy for both fuzzy and non-fuzzy implementations. In case of fuzzy implementation, we have considered two more parameters distance and average energy in addition to the residual energy and node density for selecting the cluster heads. Increasing the heterogeneity level increases the network lifetime in much proportion as compared to the increase in the network energy. In fact, using fuzzy logic for the original HEED, the network lifetime increases by 193.84 % without any increase in the network energy. In case of four-level heterogeneity, the network lifetime increases by 223.77 % and 589.07 by 24 % increases in the network energy using non-fuzzy and fuzzy implementation, respectively.

References

1. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). WSNs: A survey. *Computer Networks*, 38, 393–422.
2. Ullah, M., & Ahmad, W. (2009) Evaluation of routing protocols in WSNs. Master Thesis, Department of Interaction and System Design, School of Engineering at Blekinge Institute of Technology.
3. Lewis, F. L. (2004). *WSN*. New York: Technologies Protocols and Applications.
4. Khetrapal, A. (2006) Routing techniques for mobile ad hoc networks classification and qualitative/quantitative analysis. In *Proceedings international conference on wireless networks, (ICWN 2006)* (pp 251–257).
5. Younis, O., & Fahmy, S. (2004). HEED: A hybrid, energy efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 3(4), 366–379.
6. Frikha, M., & Slimane, J. B. (2006) Conception and simulation of energy efficient AODV protocol ad hoc networks. In *3rd International conferenc on mobile technology, applications and systems* (pp. 1–7).
7. Heinzelman, W. R., et al. (2000) Energy scalable algorithms and protocols for wireless microsensor networks. In *Proceedings of the international conference on acoustics, speech, and signal processing (ICASSP '00)*, Istanbul, Turkey.
8. Manjeshwar, A., & Agarwal, D. P. (2001) TEEN: A routing protocol for enhanced efficiency in WSNs. In *1st International workshop on parallel and distributed computing issues in wireless networks and mobile computing, April 2001*.
9. Manjeshwar, A., & Agarwal, D. P. (2001) APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in WSNs. In *Parallel and distributed processing symposium, IPDPS 2002* (pp. 195–202).
10. Akkaya, K., & Younis, M. (2005). A survey on routing protocols for WSNs. *Ad Hoc Networks*, 3, 325–349.
11. Heinzelman, W. R., Chandrakasan, A. P., & Balakrishnan, H. (2000) Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Hawaii international conference on system sciences (HICSS-33)*.
12. Heinzelman, W. R., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1(4), 660–670.
13. Lindsey, S., & Raghavendra, C. S. (2002) PEGASIS: Power efficient gathering in sensor information systems. In *Proceedings of the IEEE aerospace conference, Big Sky, Montana, March 2002*.
14. Lindsey, S., Raghavendra, C. S., & Sivalingam, K. (2001) Data gathering in sensor networks using the energy*delay metric. In *Proceedings of the IPDPS workshop on issues in wireless networks and mobile computing, San Francisco, CA*.
15. Smaragdakis, G., Matta, I., & Bestavros, A. (2004) SEP: A stable election protocol for clustered heterogeneous WSNs. In *Second international workshop on sensor and actor network protocols and applications (SANPA 2004)*.

16. Kumar, D., Aseri, T. C., & Patel, R. B. (2009). EEHC: Energy efficient heterogeneous clustered scheme for WSNs. *Computer Communications*, 32, 662–667.
17. Mao, Y., Liu, Z., Zhang, L., & Li, X. (2009). An effective data gathering scheme in heterogeneous energy wireless sensor networks. In *Proceedings of international conference on computational science and engineering* (pp. 338–343).
18. Qing, L., Zhu, Q., & Wang, M. (2006). Design of a distributed energy efficient clustering algorithm for heterogeneous WSNs. *Computer Communications*, 29, 2230–2237.
19. Haibo, Z., Yuanming, W., & Guangzhong, X. (2009). EDFM: A stable election protocol based on energy dissipation forecast method for clustered heterogeneous WSNs. In *Proceedings of the 5th international conference on wireless communications, networking and mobile computing* (pp. 3544–3547).
20. Elbhiri, B., Rachid, S., Zamora, A.-P., & Aboutajdine, D. (2009). Stochastic distributed energy-efficient clustering (SDEEC) for heterogeneous WSNs. *ICGST-CNIR Journal*, 9(2), 11–17.
21. Kumar, D., Patel, R., & Aseri, T. C. (2010). Prolonging network lifetime and data accumulation in heterogeneous sensor networks. *International Arab Journal of Information Technology*, 7(3), 302–309.
22. Kumar, D., Aseri, T. C., & Patel, R. B. (2011). EEEDA: Energy efficient clustering and data aggregation protocol for heterogeneous wireless sensor networks. *International Journal of Computers Communications & Control*, 6(1), 113–124.
23. Kumar, D. (2012). Distributed stable cluster head election (DSCHE) protocol for heterogeneous wireless sensor networks. *International Journal of Information Technology, Communications and Convergence*, 2(1), 90–103.
24. Mantri, D., Prasad, N. R., & Prasad, R. (2013). BHCDA: Bandwidth efficient heterogeneity aware cluster based data aggregation for wireless sensor network. In *International conference on advances in computing, communications and informatics (ICACCI)* (pp. 1064–1069).
25. Qureshi, T. N., et al. (2013). BEENISH: Balanced energy efficient network integrated super heterogeneous protocol for wireless sensor networks. In *International workshop on body area sensor networks (BASNet-2013)*, *Procedia Computer Science* (pp. 1–6).
26. Chand, S., Singh, S., & Kumar, B. (2014). Heterogeneous HEED protocol for wireless sensor networks. *Wireless Personal Communications*, 77(3), 2117–2139.
27. Singh, S., Chand, S., Kumar, R., & Kumar, B. (2013). A heterogeneous network model for prolonging lifetime in 3-D WSNs. *IEEE/IET*. In *4th International conference CONFLUENCE 2013: The next generation information technology summit*, 26th–27th Sept. 2013 (pp. 257–262).
28. Chand, S., Singh, S., & Kumar, B. (2013). hetADEEPS: ADEEPS for heterogeneous wireless sensor networks. *International Journal of Future Generation Communication and Networking*, 6(5), 21–32.
29. Chand, S., Singh, S., & Kumar, B. (2013). 3-Level heterogeneity model for wireless sensor networks. *International Journal of Computer Network and Information Security (IJCNIS)*, 5(4), 40–47.
30. Taheri, H., Neamatollahi, P., Younis, O. M., Naghibzadeh, S., & Yaghmaee, M. H. (2012). An energy-aware distributed clustering protocol in wireless sensor networks using fuzzy logic. *Ad Hoc Networks*, 10(7), 1469–1481.



Samayveer Singh received his B.Tech. in Information Technology from Uttar Pradesh Technical University, Lucknow, India in 2007 and his M.Tech. in Computer Science & Engineering from National Institute of Technology, Jalandhar, India, in 2010. He is pursuing his PhD in the Department of Computer Engineering, Netaji Subhas Institute of Technology, New Delhi, India. His research interest includes wireless sensor networks.



Satish Chand did his M.Sc. in Mathematics from Indian Institute of Technology, Kanpur, India and M.Tech. in Computer Science from Indian Institute of Technology, Kharagpur, India and Ph.D. from Jawaharlal Nehru University, New Delhi, India. Presently he is working as a Professor in Computer Engineering Division, Netaji Subhas Institute of Technology, Delhi, India. Areas of his research interest are Multimedia Broadcasting, Networking, Video-on-Demand, Cryptography, and Image processing.



Bijendra Kumar did his Bachelor of Engineering from H.B.T.I. Kanpur, India. He has done his Ph.D. from Delhi University, Delhi, India in 2011. Presently he is working as an Assistant Professor in Computer Engineering Division, Netaji Subhas Institute of Technology, Delhi, India. His areas of research interests are Video applications, watermarking, and Design of algorithms.