



An Energy-aware Distributed Clustering Protocol in Wireless Sensor Networks

Venkateswarulu Naik. B

JJTU Scholar, Jhunjhunu, Rajasthan, India

Dr.S.Rama Krishna

Principal, SVIT, Secunderabad, India

ABSTRACT

Clustering is an effective approach for organizing a network into a connected hierarchy, load balancing, and prolonging the network lifetime. This paper proposes an energy-aware distributed dynamic clustering protocol (ECPF) which applies three techniques:

- 1) Non-probabilistic Cluster Head (CH) elections.
- 2) On demand clustering. The remaining energy of the nodes is the primary parameter for electing tentative CHs via a non-probabilistic fashion. Anon-probabilistic Selection is implemented by introducing a delay inversely proportional to the residual energy of each node. Therefore, tentative CHs are selected based on their remaining energy. Besides, in ECPF, CH elections are performed sporadically (in contrast to performing it every round). Simulation results demonstrate that our approach performs better than well known protocols (LEACH, HEED, and CHEF) in terms of extending network lifetime and saving energy.

Keywords: *Sensor Networks, Clustering, Network Lifetime, Energy Efficient Protocols, Distributed Algorithms*

1. INTRODUCTION

Wireless sensor networks (WSNs) provide reliable monitoring from very long distances. These networks are basically data gathering networks in which data are highly correlated and the end user needs a high level description of the environment ensued by the nodes [1]. The requirements of these networks are

ease of deployment, long system lifetime, and low-latency data transfers. The main task of a sensor node in a sensor field is to detect events, perform quick local data processing, and then to transmit the data [2]. As mentioned in [3] and [4], nodes have typically low mobility and are limited in capabilities, energy supply and bandwidth. The sensor network should perform for as long as possible. On the other hand, battery recharging may be inconvenient or impossible. Therefore, all aspects of the sensor node, from the hardware to the protocols, must be designed to be extremely energy efficient [5]. In a sensor node, energy consumption can be “useful” or “wasteful”. Useful energy consumption can be due to one or more of the following causes: x transmitting/receiving data x processing query requests x forwarding queries/data to neighboring nodes Wasteful energy consumption can be due to:

- Idle listening to the media x retransmitting due to packet collisions x overhearing
- Generating/handling control packets [6]

In direct communication WSN, the sensor nodes directly transmit their sensing data to the Base Station (BS) without any coordination between the two. However, in Cluster-based WSNs, the network is divided into clusters. Each sensor node exchanges its information only with its cluster head (CH), which transmits the aggregated information to the BS. Aggregation and fusion of sensor node data at the CHs cause a significant reduction in the amount of data sent to the BS and so results in saving both energy and bandwidth resources. Once the clusters are

constructed, each sensor node will be given an exclusive time slot; therefore, each sensor node knows when to transmit. Consequently, a node does not require being awake during the complete Time Division Multiple Access (TDMA) frame, but only during its specific time slot [7]. To sum up, by means of a common schedule, clustering coordinates the transmissions of sensor nodes during the steady state phase and so eliminates collisions, idle listening, and overhearing. In this way, clustering achieves a significant improvement in terms of energy consumption. Besides, it is particularly crucial for scaling the network to hundreds or thousands of nodes [8]. In many applications, cluster organization is a natural way to group spatially close sensor nodes in order to exploit the correlation and Page | 4 eliminate the redundancy that often shows up in the sensor readings [9]. However, these benefits, compared to those of the direct communication WSN, result in extra overhead due to the cluster formation's message exchanges. Research on clustering in WSNs has focused on developing centralized and distributed protocols to compute sets of CHs and to form clusters. Centralized approaches (e.g. [10, 11, 12, and 13]) are rather inefficient in the case of large scale networks since collecting the entire amount of necessary information at the central control (BS) is both time and energy consuming. Distributed approaches are more efficient for large scale networks. In these approaches, a node decides to become a CH or to join a cluster based on the information obtained solely from neighbors within its proximity. Several distributed clustering protocols have been proposed in literature (e.g. [5, 6, 14, 15, and 16]). As mentioned in [17], most of these protocols are in either case of iterative or probabilistic. In probabilistic protocols (e.g. [14, 15, and 18]), the decision to become CH is reached probabilistically. On the other hand, in iterative protocols (e.g. [6]), the nodes perform an iterative process to decide when there to become a CH or not. From another point of view, clustering protocols are considered as being static and dynamic. In static clustering, the clusters are permanently formed (e.g. [8, 10]), while in dynamic clustering (like [5, 6, 9, 14, and 15]), protocol operation is divided into rounds; clusters are formed for a round and then should be formed again for the next round. In doing so, extra overhead is imposed on the system. On the other hand, some protocols (e.g. [11, 16, 12, and 13]) take advantage of fuzzy logic. Fuzzy Logic [19, 20] is useful for making real-time decisions without needing complete information about the

environment. Merging different environmental parameters according to predefined rules and then making a decision based on the result is another important application of fuzzy logic. Typically, fuzzy clustering algorithms in WSNs use fuzzy logic for merging different clustering parameters to elect CHs. Besides, as mentioned in [11] and [16] the overhead of cluster head election may be highly reduced by using fuzzy logic. This paper proposes an Energy-aware distributed dynamic Clustering Protocol using Fuzzy logic named ECPF. The proposed clustering approach does not make any assumptions regarding the distribution of the nodes or node capabilities, e.g., location-awareness. The protocol only assumes that sensor nodes can vary their transmission power. In this protocol, each node employs a process to decide its status. For each node, this process finishes when the node either elects itself as a CH or finds a CH to join. Notable features of ECPF are: x Distributed CH election (based on local information) that avoids extra communication with the BS.

Non-probabilistic choice of CH: a node waits for a certain delay (which is inversely proportional to the remaining energy of that node), before it tries to proclaim itself as a CH or join a cluster. The nodes whose delays expire first among the neighboring nodes will become a tentative CH, and in the next round a cost-based choice is made to choose a final CH from the set of neighboring tentative ones. Clustering is performed sporadically (on demand rather than each round) when some CH depletes a given fraction of its energy resources. We compare our solution to LEACH, HEED, and CHEF protocols. Simulation results in Mat lab software show that ECPF provides superior network lifetime and energy savings. The rest of the paper is organized as follows: Section 2 gives a short survey of cluster based protocols for WSNs. Section 3 describes the network model and clustering problem. A new energy efficient clustering scheme is outlined in Section 4. Section 5 presents the simulation results by comparing energy consumption, network lifetime, and the number of CH elections with other well-known algorithms. Finally, the conclusion is presented.

2. RELATED WORKS

The following presents a review of some famous clustering protocols. LEACH [5] minimizes energy dissipation in sensor networks due to its constructing of clusters. LEACH operation is performed in two

phases: a setup phase and a steady state phase. In the setup phase, a sensor node selects a random number between 0 and 1. If this number is less than the threshold $T(n)$, the node becomes a CH. $T(n)$ is computed as:

Where r is the current round; p , the desired percentage for becoming CH; and G is the collection of nodes not elected as a CH in the last $1/p$ rounds. After being elected, every CH announces to all of the network's sensor nodes that it is the new CH. When each node receives this announcement, it chooses a cluster to join based on the signal strength of the announcement. The sensor nodes then inform their appropriate CH to join them. Afterwards, the CHs, according to a TDMA approach, assign a time slot to each node so that its data can be sent to its CH during this period. During the steady state phase, the sensor nodes can perform sensing and transmit data to the CHs. The CHs also aggregate the data received from the nodes in their cluster before sending these data on to the BS. After a certain period of time has elapsed in the steady state phase, the network goes into the setup phase again and enters the next round. The advantages of the LEACH protocol over previous research are as follows: In this probabilistic approach, the nodes die randomly and at the same rate's The dynamic clustering of LEACH prolongs the network lifetime. x LEACH is fully distributed and does not require global knowledge of the network. The limitations of the LEACH protocol are as follows: x although energy consumption is a critical problem in WSNs, LEACH does not consider the remaining energy of nodes when selecting CHs. x Since CH election is probabilistic, a node with very low energy has a good chance of becoming a CH. When this node dies, the entire cluster is rendered dysfunctional. x It is possible that some CHs are located within close proximity of each other. This indicates that CHs are not well distributed in the network. x It is assumed that CHs have a long communication range enabling them to send data directly to the BS. This assumption is not always realistic since, due to signal propagation problems, such as the presence of obstacles, the BS is often directly unreachable to all nodes. On the other hand, the CHs have the capabilities of regular sensor nodes. Consequently, LEACH is not applicable to networks deployed in large areas.

The authors in [14] extended the LEACH's probabilistic CH selection algorithm. They adjusted the threshold $T(n)$ denoted in (1), relative to the

node's residual energy. Through applying this threshold each node decides whether to become a CH in a round or not lifetime can be efficiently increased. There are 27 fuzzy if-then rules which are defined at the BS. The BS elects the CHs according to these fuzzy rules. This centralized approach is not suitable for scalable networks because BS must collect information about the status and location of all nodes. LEACH-FL [2] is an improvement on LEACH protocol which employs a similar approach to [3]. This method uses three descriptors (node residual energy, node degree and distance from BS) for computing the chance. The BS selects nodes with higher chance as CHs, using 27 fuzzy if-then rules. Although this method has the same drawback of Gupta's method, it presents a better result than LEACH protocol. CHEF [4] is a fuzzy approach which performs CH election in a distributed manner. In every round, each node generate s a random number between 0 and 1. If the random number is smaller than the predefined threshold, then that node becomes a tentative CH. There are two fuzzy descriptors that are used in CH election: residual energy of each node and local distance. The local distance is the sum of distances that a node has with other nodes in radius r . There are 9 fuzzy if -then rules that are defined in all sensor nodes. Tentative CHs calculate their chances to be an actual CH using these fuzzy rules. If the chance of a tentative CH is greater than the other tentative CHs' chances in radius r , then that tentative CH becomes an actual CH. Then, it sends a CH advertisement message to the nodes in its proximity. The nodes that are not elected as CH join the closes t CH. This method applies a probabilistic model for CH elections, too. Therefore, it is possible that CHs are not well distributed in the field. Consequently, some nodes find themselves uncovered (orphan nodes), and have to send their sensed data directly to the BS.

Bandyopadhyay and Coyle [15] proposed another extension of the LEACH protocol where the multi-hop routing is applied. Similar to LEACH, every CH advertises itself to the neighboring sensor nodes, which relay the advertisement in a multi-hop fashion. The advertisement is forwarded to sensor nodes in at most h hops away. Cluster Members (CMs) that receive multiple CH announcements, elect the closest CH in terms of hop count. On the other hand, a sensor node which is neither a CH nor receives any CH announcement becomes a forced CH. WSN operation in a multi-hop fashion has more energy conservation in communications in comparison with single hop

transmissions, especially in large scale networks. This gain achieves in the cost of additional complexity, e.g., the one hop CMs require data collection from the two hop CMs. In addition, the overhead in the setup phase increases considerably, because CH messages have to be forwarded via multiple hops. Youngish and Filmy [6] proposed an iterative clustering protocol, named HEED. HEED is different from LEACH in the manner in which CHs are elected; however, it employs probabilistic fashion. Both electing CHs and joining clusters are performed based on the hybrid combination of two parameters. The primary parameter depends on the node's residual energy. The alternative parameter is the intra-cluster communication cost. This technique is utilized in ECPF with a fuzzy cost. In HEED, each node computes a communication cost depending on whether variable power levels, applied for intra-cluster communication, are permissible or not. If the power level is fixed for all of the nodes, then the communication cost can be proportional to (i) node degree, if load distribution between CHs is required, or (ii) $1/\text{node degree}$, if producing dense clusters is required. The authors defined AMRP the average of the minimum power levels needed by all M nodes within the cluster range to access the CH u , i.e. $MiMinPwruAMRPMi1)()$. If variable power levels are admissible, AMRP is used as the cost function. In this approach, every regular node elects the least communication cost CH in order to join it. On the other hand, the CHs send the aggregated data to the BS in a multi-hop fashion. The advantages of the HEED protocol are as follows:

- It is a fully distributed clustering approach that benefits from the use of two parameters for CH election.
- The probability of two nodes within each other's transmission range becoming CHs is negligible. Therefore, in contrast with LEACH, CHs are well distributed in the network.
- Energy consumption is not required to be uniform for all the nodes.
- Communications in a multi-hop fashion between CHs and the BS promote more energy conservation and scalability in contrast with the single-hop fashion in the LEACH protocol.

The limitations of the HEED protocol are as follows:

It uses a probabilistic model for CH elections.

Similar to LEACH, the performing of clustering in each round imposes significant overhead on the network. This overhead causes noticeable energy

dissipation which results in decreasing the network lifetime. Some protocols attempt to eliminate overhead due to setup phase. As an example in [8], Zhu et al presented a distributed static clustering protocol to prolong the network lifetime. This includes three parts. First, nodes by means of Hausdorff clustering algorithm organize themselves into multiple static clusters based on location of nodes, communication effectiveness, and network connectivity. Second, clusters are formed only once, and the CH role is scheduled between the CMs optimally. Third, after CH elections, CHs construct a backbone network to periodically collect, aggregate, and send data to the BS using minimum energy routing. They showed that this method considerably prolong the network lifetime in comparison with some other known methods because it eliminates the communication overhead due to setup phase. This approach suffers the problems due to the proximity of the CHs.

3. PRELIMINARIES

3.1 Network Model

The following properties are assumed in regard to the sensor network being studied:

- The nodes can use power control to change the amount of transmit power. Also, each node performs signal processing functions and has the computational power to support different MAC protocols.
- The nodes have ideal sensing capabilities. In other words, the quality of the node's sensing does not change within the cluster range regardless of the distance from the node.
- The sensor nodes are quasi-stationary. This is typical for sensor network applications. x Nodes are not equipped with GPS-capable antennae, meaning they are location-unaware. In addition to being of equal importance, the capabilities of nodes, such as processing and communicating, are similar.
- Nodes are energy constrained and are left unattended after deployment. Therefore, battery recharge is not possible.
- Because the energy consumed per bit for sensing, processing, and communicating is typically known, remaining energy can be estimated. As a result, measuring this remaining energy is not essential.

- Each node has an initial amount of energy, A_{max} , and the BS is not limited in terms of energy, memory, and computational power.
- Node failures are basically due to energy depletion.
- Distance can be measured based on the wireless radio signal power.
- Links are symmetric, i.e., two nodes v_1 and v_2 can communicate using the same transmission power. TCP and TNO are defined as follows:
- TCP (the period of the clustering process) is the time interval used by the clustering protocol to cluster the network.
- TNO (the network operation interval) is the time between the end of a TCP interval and the start of the subsequent TCP interval. In order to reduce overhead, it must be ensured that $TNO \geq TCP$.

Note that, in contrast with other dynamic clustering protocols that perform clustering in each round, ECPF clusters the nodes on demand rather than at each round. Therefore, it is possible that some rounds do not include TCP; instead TNO is extended during these rounds. As a result, the length of the TCP interval is fixed but the length of the TNO interval varies throughout the network lifetime. In ECPF, node readings are periodically reported to the BS. Therefore, a TDMA frame is created in each CH to remove interference within a cluster. The protocol uses special synchronization pulses to alert the sensor nodes that clustering will be triggered in the beginning of the next round. These pulses are propagated in a centralized multi-hop fashion (like the approach presented in [21]). The basis of this approach is the construction of a low-depth spanning tree T comprising the nodes in the network. In general, a new spanning tree is constructed each time the algorithm is performed. In order to synchronize nodes in the tree, pair-wise synchronizations are performed along the edges of T . In centralized multi-hop synchronization, the reference node (BS) initiates the synchronization through all its immediate (single-hop) children in T . Next, every child of the reference node, in turn, synchronizes with its children. This process continues until the leaf nodes of T are reached. The algorithm terminates when all the leaf nodes are synchronized. The running time of the algorithm is proportional to the depth of the tree which is $\log(n)$, where n is the number of nodes in the tree. Therefore, these pulses are quickly penetrated throughout the network.

3.2 The Clustering Problem

Suppose the above assumptions hold and that n nodes

are distributed in a field. In consideration of energy saving issues, the goal is to identify a collection of CHs which cover the entire area. Each node v_i , where $1 \leq i \leq n$, must be mapped to exactly one cluster c_j , where $1 \leq j \leq n_c$, and n_c is the number of clusters ($n_c \leq n$). L_i shall denote the lifetime of node i . The network lifetime will be defined as follows: x_F is the time elapsed until the First Node Dies (FND). Therefore, $F = \min(L_1, L_2, \dots, L_n)$. x_H is the time elapsed until only one Half of Nodes remain Alive (HNA). In other words, $H = \text{median}(L_1, L_2, \dots, L_n)$. x_L is the time elapsed until the Last Node Dies (LND) or, $L = \max(L_1, L_2, \dots, L_n)$. The major purpose here is to maximize F , H , and L , which requires using the energy of all nodes uniformly. A node must have the ability to directly communicate with its CH and by a single-hop fashion. A CH has two critical responsibilities: (1) intra-cluster coordination and (2) inter-cluster communication. Multi-hop routing is used for inter-cluster communication. CHs can utilize a routing protocol to compute inter-cluster paths for communicating in a multi-hop fashion with the BS, e.g. the power-aware routing protocol in [22].

The following requirements are recommended:

- Clustering is fully distributed. Each node decides independently based on local information's. Clustering finishes within a fixed number of iterations (regardless of network diameter).
- At the end of each TCP, each node is either a CH or a regular node that belongs to exactly one cluster.
- In terms of processing complexity and message exchange, clustering should be efficiently performed.
- CHs are well distributed over the sensor field. Note that, in the clustering process, every iteration takes time, t_c . Period t_c should be long enough to receive messages from any neighbor within the cluster radius. Because the nodes are quasi-stationary, neighbor discovery is not required every time clustering is performed. Therefore, the neighbor set of every node does not vary very frequently. In multi-hop networks, the nodes automatically update their neighbor sets by periodically sending and receiving heartbeat messages.

3. THE PROTOCOL

In this section, the ECPF and its pseudo code are illustrated. The operation of ECPF is divided into rounds and each round is comprised of two phases:

1. **The setup phase**, which includes CH election and consequently cluster formation. In addition, in this phase, every CH coordinates with its members to send sensing data during the following phase.
2. **The steady state phase**, which is broken up into TDMA frames. During each frame, every regular node, at the time of its respective time slot, sends sensing data to its CH (similar to [5]). At the end of each TDMA frame, every CH forwards the aggregated data to the BS through the CHs.

This protocol has the following characteristics which resemble [6]'s:

- The steady state phase is similar.
- A chosen CH advertises only to its neighbors.
- Each node can directly communicate with its CH.
- During the clustering process, a node can be either a tentative_CH or a final_CH or it can be covered. At the end of the setup phase, CHs form a network backbone, such that packets are routed from the CHs to the BS in a multi-hop fashion over CHs.

4.1 On Demand Clustering

A novelty of ECPF is that it decreases overhead by performing the setup phase on demand instead of in each round. To do so, when the clustering process finishes (at the end of each setup phase), every CH saves its residual energy in its memory, for example in its ECH variable. During the steady state phase, whenever a CH finds that its Residual falls below α (where α is a constant number and 10), it sets a prespecified bit in a data packet which is ready to be sent to the BS in the current TDMA frame. Upon receiving the forwarded CH data packet (sent in a multi-hop fashion), the BS informs the sensors to hold the setup phase at the beginning of the upcoming round. This could be achieved by having the BS send out, in a multi-hop fashion, specific synchronization pulses to nodes. These pulses are quickly dispersed throughout the network. When every node receives a pulse, it prepares itself to perform clustering. Therefore, CH election and consequently cluster formation are performed on demand. As a result, the overhead created by consecutive setup phases is tremendously reduced. Consequently, there is a decrease in the energy dissipation of nodes and an increase in network lifetime.

4.2 Fuzzy Cost

Fuzzy Logic (FL) is used to model human experience and human decision making behavior. In FL the

input-output relationship is expressed by using a set of linguistic rules or relational expressions. As shown in Fig. 2 a FL basically consists of four important parts including a fuser, a defuzzifier, an inference engine, and a rule base. As in many fuzzy applications, the input data are usually crisp, so a fuzzification is necessary to convert the crisp input data into a suitable set of linguistic value which is needed by the inference engine. In the rule base of an FL, a set of fuzzy rules, which characterize the dynamic behavior of the system, are defined. The inference engine is used to form inferences and draw conclusions from the fuzzy rules. The output of the inference engine is sent to the defuzzification unit. Defuzzification is a mapping from a space of fuzzy actions into a space of crisp actions. We have employed the most commonly used fuzzy inference technique, called the Madman [23] method, because of its simplicity. To obtain a cost, ECPF uses two fuzzy sets and the fuzzy if-then rules. We adjusted the input variables, used in the fuzzy if-then rules, between 0 and 1 such that the fuzzy sets will be applicable for any size of networks. The fuzzy system input variables are defined as follows:

- **Node degree:** the number of neighbors a node has which is divided by total number of nodes in the network. In other words, $node_degree_i = (|S_{narl}(i)| / \#nodes)$ where, $S_{narl}(i) = \{v: v \text{ lies within node } i\text{'s cluster range}\}$.
- **Node centrality:** a value that shows how central the node is among its neighbors proportional to network dimension, or $node_centrality_i = Dimension$

$$Network_i = \frac{S_{jdistn}(i)}{S_{jnbr}_i} \cdot \frac{1}{(2 + \dots)}$$

Since, typically transmissions energy is proportional to the squared distance, the lower value of the node centrality results in a lower amount of energy required by other nodes to send data to the node assuming the role of a Chute fuzzy sets of input variables and output variable (cost) are described in Fig. 3. ECPF calculates a cost using fuzzy if-then rules. A smaller cost means that the node has a higher priority of being elected a CH. Based on the two fuzzy variables, fuzzy if-then rules can be defined which are similar to those presented in Table I. After aggregating the results achieved from each rule, a defuzzification method is required to obtain the crisp value. Defuzzification is performed using the Coal method, which returns the Center of Area under the fuzzy set achieved aggregating conclusions. demonstrates the effect of nodes' attributes on the chance of the nodes becoming

CHs. This plot was generated by the rules that accounted for both node degree and node centrality factors. With the increase of node degree and decrease of node centrality, the cost of a node to be selected as a CH is decreased. A node with lower cost has more priority for becoming CH.

4.3 Clustering Process

The clustering process of ECPF is divided into three phases:

Initialization Phase:

In the beginning of this phase, neighbor information could be updated using CSMA/CA. Afterwards, each node may compute its cost independently. The cost is the output of fuzzy system described in the previous section. This cost will not be broadcasted to neighbors as it is exchangeable through Chums messages. As previously mentioned, note that updating the neighbor information and computing costs are not required every time clustering is triggered. Each sensor node sets its own delay time: $(1 - \frac{E_{residual}}{E_{max}}) \times \text{delay_residual}$. In the above, Residual is the current energy of the sensor node and Exam is the maximum energy corresponding to a fully charged battery. Therefore, the main constraint in sensor nodes, i.e., residual energy, is taken into account in this delay time. The value of dams limits the period of time should be elapsed in the clustering process. See the Initialization phase in x Main processing Phase: In this phase, every node must wait until the expiration of its delay time. If a node does not hear the Chums message from any other sensor node during the delay time, upon delay time expiration it shall declare itself to be a tentative CH. The node announces its status by sending Chums(Nodded, tentative_CH, cost), to all the nodes within the cluster range. Note that when a node has higher energy, its delay time is less than that of nodes with a lower amount of energy. As a result, because its delay time has expired sooner, the node has a higher priority of becoming selected as a tentative_CH. In this way, the non-probabilistic method of selecting CHs is employed. In the next iteration, if this particular node has the least cost among the tentative_CHs in its proximity, it will become a final_CH and shall broadcast a final_CH message within its cluster range. On the other hand, if a node receives a final_CH message, it can no longer be elected as a CH. Therefore, in the following phase, it must choose to connect to one of the final_CHs in its cluster radius, based on the cost of that final_CH. Observe that, in this phase, each tentative_CH (or

final_CH) node can send a Chums only once. It is possible that in the beginning of this phase some neighboring nodes whose delays expire at the same time will become tentative_CHs, however in the next iteration only one of them declares itself as a final_CH. Therefore, final_CHs do not locate in each other's cluster range. See the Main processing phase x Finalization Phase: During this phase, each sensor node makes a final decision about its status. If the node is not a final_CH and has received at least one final_CH message, it will elect the final_CH with the least cost to join it. If a node completes the clustering process and has not yet received any final_CH message, it will find itself uncovered and so shall introduce itself as the final_CH. See the Finalization phase. All distributed protocols face the convergence issue in the cluster head election (e.g. when two nodes receive tentative CHs messages with same cost). This condition rarely happens; both of the nodes should have equal amount of energy to have equal delay time, on the other hand, they should not be in the cluster range of any final_CH. Besides, they should execute the algorithm at the same time. However, the least cost function can select the node with the lower ID in this case.

5. SIMULATION RESULTS

In this section, a comparison between the simulation results in ECPF, LEACH, HEED and CHEF protocols is performed via Mat lab software. The following assumptions and system parameters (similar to [5]) are used:

- The nodes always have data to send to the end user and the nodes situated in close proximity to others have correlated data.
- The energy required for data aggregation is set as $E_{DA} = 5nJ/bit/signal$ and CHs perform ideal data aggregation (i.e. all the messages received from cluster members can be aggregated into a single message).
- A simple model for the energy dissipation of radio hardware is assumed, in which the receiver dissipates energy to run the radio electronics and the transmitter dissipates energy to run the power amplifier and radio electronics, as shown in. Thus, for transmitting a k-bit message over distance d, the radio expends: $E_{Tx}(kid) = E_{Tx_elec}(k) + E_{Tx_amp}(kid)$

5.1. Setting α Variable

As mentioned before, CH election (clustering execution) is performed on demand. When a CH consumes a prespecified part of its energy (i.e. $CH_{residual}EEDd$), it indirectly informs the other nodes that clustering must be performed for the upcoming round. In order to obtain the proper α , ECPF was run for the two scenarios described above. α differs from 0 to 1 and each plot demonstrates the average of three executions for 100, 200, 300, and 400 number of nodes, until the first node dies (FND). Also plotted is an average of the four mentioned plots. When α is equal to zero, no clustering is performed during the network lifetime (i.e. the static clustering approach that considers fixed CHs). In homogenous networks in which nodes have similar capabilities and the same amount of energy, CHs quickly deplete respective energies. Therefore, when a CH dies, the respective cluster becomes dysfunctional. When α equals one, this signifies that clustering is performed in each round, similar to the LEACH, HEED and CHEF protocols. Considering the plot, which is the average of the four different numbers of nodes, figures 8-14 are plotted using 8.0 Das it approximately results in better network lifetime.

5.2 Energy Consumption Comparisons

In this subsection, the energy dissipation to cluster the WSN and the energy consumption to transmit the sensed data to the BS are evaluated. Note that in the figures belonging to this subsection, the vertical axis in Scenario 1 does not have an equal range of data in contrast to Scenario 2 because of the high energy dissipation of LEACH in Scenario 2. In Fig. 8, the average energy dissipation of protocols clustering the WSN per election is evaluated. The ECPF performs better because its clustering's message complexity is low and, similar to HEED and CHEF.

6. CONCLUSION

In this paper, we proposed an energy efficient, distributed clustering protocol for WSNs. Our approach can be useful for applications that require scalability, prolonged network lifetime and nodes are dispersed in a spacious field. We assumed quasi-stationary networks in which nodes are location-unaware and have equal significance. Based on this assumption, we presented the ECPF, where terminates CH election process with a constant number of iterations, and without any dependency of the network

diameter. Combining fuzzy logic, on demand clustering, non-probabilistic CH election, and consideration of nodes' energy, allows ECPF achieve longer lifetime when compared to existing clustering protocols. Many applications require the ability to provide information from each part of the monitored area at any moment in order to meet the application's quality of service (Quos) [9]. For future works, we would like to extend the protocol to meet Quos requirements of WSNs, such as coverage preservation, because complete coverage of the monitored area over long period of time is an outstanding issue.

REFERENCES

- 1) W. B. Heinemann, "Application-specific protocol architectures for wireless networks," Ph.D. Thesis, Massachusetts Institute of Technology, (2000).
- 2) I. Akiyildiz, W. Su, Y. Sankarasubramaniam, and E. Cerci, "A survey on sensor networks," IEEE Communication Magazine, vol. 40, no. 8, pp. 102-114, (2002).
- 3) O. Youngish and S. Foamy, "Distributed clustering in ad-hoc sensor networks: A hybrid, energy-efficient approach," in Proceeding of IEEE INFOCOM, vol. 1, pp. 629-640, (2004).
- 4) K. Akaka, M. Youngish, "A survey on routing protocols for wireless sensor networks," Ad Hoc Networks, vol. 3, no. 3, pp. 325-349, (2005).
- 5) W. B. Heinemann, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless micro sensor networks," IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660-670, (2002).
- 6) O. Youngish, S. Filmy, "HEED: A Hybrid, Energy-Efficient, Distributed clustering approach for Ad Hoc sensor Networks," IEEE Transactions on Mobile Computing, vol. 3, no. 4, pp. 366-379, (2004).
- 7) N. Bouabdallah, M. E. Rivers-Angeles, and B. Seri cola, "Continuous Monitoring Using Event-Driven Reporting for Cluster-Based Wireless Sensor Networks," IEEE Transactions on Vehicular Technology., vol. 58, no. 7, pp. 3460-3479, (2009).
- 8) X. Zhu, L. Sheen, and T. P. Yum, "Hausdorff Clustering and Minimum Energy Routing for Wireless Sensor Networks," IEEE Transactions on

- Vehicular Technology, vol. 58, 2, pp. 990-997, (2009).
- 9) S. Sorrow, W. Heinemann, "Cluster head election techniques for coverage preservation in wireless sensor networks", *Ad Hoc Networks*, vol. 7, no. 5, pp. 955-972, (2009).
 - 10) T. Kauri, J. Beak, "A Strategic Deployment and Cluster-Header Selection for Wireless Sensor Networks," *IEEE Transactions on Consumer Electronics*, Vol. 55, 4, pp. 1890-1897, (2009).
 - 11) I. Gupta, D. Riordan, and S. Sample, "Cluster-head election using fuzzy logic for Wireless Sensor Networks," In *Communication Networks and Services Research Conference, Proceedings of the 3rd Annual*, pp. 255–260, 2005.
 - 12) G. Ran, H. Zhang, S. Gong, "Improving on LEACH protocol of Wireless Sensor Networks Using Fuzzy Logic," *Journal of Information & Computational Science*, pp. 767-775, 2010.
 - 13) N. Bazaars Torghabeh, M. R. Akbarzadeh Totochi, and M. H. Yaghmaee Moghaddam, "Cluster Head Selection using a Two-Level Fuzzy Logic in Wireless Sensor Networks," *International Conference on Computer Engineering and Technology*, pp. 357-361, 2010.
 - 14) M. C. M. Thin, T. Their, "An Energy Efficient Cluster-Head Selection for Wireless Sensor Networks," *International*
 - 15) S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proceeding of IEEE INFOCOM*, vol. 3, pp. 1713-1723, (2003).
 - 16) 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," *International Conference of Advanced Communication Technology*, pp. 654–659, 2008.
 - 17) O. Youngish, M. Runs, and S. Ramasubramanian, "Node Clustering in Wireless Sensor Networks: Recent Developments and Deployment Challenges," *IEEE Network*, vol. 20, 3, pp. 20-25, (2006).
 - 18) W. B. Heinemann, A. P. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Micro sensor Networks," *Proceeding of the Hawaii International Conference on System Sciences*, pp. 1-10, (2000).
 - 19) L. A. Zane, "Fuzzy Sets," *Information and Control*, vol. 8, 3, pp. 338-353, (1965).
 - 20) L. A. Sade, "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes," *IEEE Transactions on Systems, Man and Cybernetics*, SMC-3, 1, pp. 28-44, (1973).
 - 21) J. Greened, J. Rabies, "Lightweight Time Synchronization for Sensor Networks," *International Workshop on Wireless Sensor Networks and Applications*, (2003).
 - 22) S. Singh, Miwok, and C. Raghavendra, "Power-aware routing in mobile ad hoc networks," in *Proceeding of 4th Annual ACM/IEEE International Conference on Mobile Computing Networking (Modicum)*, (1998).
 - 23) E.H. Madman, and S. Assailant, "An experiment in linguistic synthesis with a fuzzy logic controller." *International Journal of Man-Machine Studies*, vol. 7, 1, pp. 1-13, (1975).
 - 24) R. Shah, J. Rafael, "Energy aware routing for low energy ad hoc Sensor Networks," *IEEE Wireless Communications and Networking Conference*, pp. 350-355, vol.1, (2002).