FUZZY UNEQUAL CLUSTERING IN WIRELESS SENSOR NETWORKS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

HAKAN BAĞCI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JANUARY 2010

Approval of the thesis:

FUZZY UNEQUAL CLUSTERING IN WIRELESS SENSOR NETWORKS

submitted by **HAKAN BAĞCI** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen Dean, Graduate School of **Natural and Applied Sciences**	________________
Prof. Dr. Müslim Bozyiğit Head of Department, **Computer Engineering**	________________
Prof. Dr. Adnan Yazıcı Supervisor, **Computer Engineering Dept., METU**	________________

Examining Committee Members:

Prof. Dr. Müslim Bozyiğit Computer Engineering Dept., METU	________________
Prof. Dr. Adnan Yazıcı Computer Engineering Dept., METU	________________
Assoc. Prof. Dr. Ahmet Coşar Computer Engineering Dept., METU	________________
Assoc. Prof. Dr. Halit Oğuztüzün Computer Engineering Dept., METU	________________
Asst. Prof. Dr. İbrahim Körpeoğlu Computer Engineering Dept., Bilkent University	________________

Date: **14.01.2010**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: HAKAN BAĞCI

Signature :

ABSTRACT

FUZZY UNEQUAL CLUSTERING IN WIRELESS SENSOR NETWORKS

Bağcı, Hakan
M.S., Department of Computer Engineering
Supervisor : Prof. Dr. Adnan Yazıcı

January 2010, 64 pages

In order to gather information more efficiently, wireless sensor networks are partitioned into clusters. The most of the proposed clustering algorithms do not consider the location of the base station. This situation causes hot spots problem in multi-hop wireless sensor networks. Unequal clustering mechanisms, which are designed by considering the base station location, solve this problem. In this thesis, we propose a fuzzy unequal clustering algorithm (EAUCF) which aims to prolong the lifetime of wireless sensor networks. EAUCF adjusts the cluster-head radius considering the residual energy and the distance to the base station parameters of the sensor nodes. This helps decreasing the intra-cluster work of the sensor nodes which are closer to the base station or have lower battery level. We utilize fuzzy logic for handling the uncertainties in cluster-head radius estimation. We compare our algorithm with some popular algorithms in literature, namely LEACH, CHEF and EEUC, according to First Node Dies (FND), Half of the Nodes Alive (HNA) and energy-efficiency metrics. Our simulation results show that EAUCF performs better than other algorithms in most of the cases considering FND, HNA and energy-efficiency. Therefore, our proposed algorithm is a stable and energy-efficient clustering algorithm.

Keywords: Wireless Sensor Networks, Fuzzy Logic, Fuzzy Clustering, Unequal Clustering, Probabilistic Clustering

ÖZ

KABLOSUZ ALGILAYICI AĞLARDA BULANIK DEĞİŞKEN YARIÇAPLI KÜMELEME

Bağcı, Hakan
Yüksek Lisans, Bilgisayar Mühendisliği
Tez Yöneticisi : Prof. Dr. Adnan Yazıcı

Ocak 2010, 64 sayfa

Kablosuz algılayıcı ağlar daha verimli bir şekilde veri toplanmasını sağlamak için kümelere ayrılırlar. Şu ana kadar tasarlanan kümeleme algoritmalarının birçoğu baz istasyonun konumunu hesaba katmamıştır. Bu durum çoklu atlamalı kablosuz algılayıcı ağlarda sorunlu bölge problemine sebep olmaktadır. Baz istasyon konumunu dikkate alarak tasarlanan değişken yarıçaplı kümeleme algoritmaları bu problemi çözmektedir. Bu tezde kablosuz algılayıcı ağların ömrünü uzatmayı amaçlayan bulanık değişken yarıçaplı bir kümeleme algoritması (EAUCF) önerilmektedir. EAUCF, algılayıcıların enerji seviyelerini ve baz istasyona olan uzaklıklarını dikkate alarak kümelerin yarıçapını belirler. Bu durum, baz istasyona daha yakın olan veya pil seviyesi daha düşük olan küme liderlerine daha az küme içi iş verilmesini sağlar. Kümeleme yarıçapı hesaplamada ortaya çıkan belirsizliklerle başa çıkmak için bulanık mantık kullanılmıştır. Tasarladığımız algoritmayı literatürde popüler olan LEACH, CHEF ve EEUC kümeleme algoritmları ile FND, HNA ve enerji verimliliği ölçütlerini kullanarak karşılaştırdık. Simulasyon sonuçlarımıza göre çoğu durumda EAUCF algoritması diğer algoritmalardan FND, HNA ve enerji verimliliği açısından daha iyi performans göstermiştir. Bundan dolayı tasarladığımız algoritma kararlı ve enerji verimli bir kümeleme algoritmasıdır.

Anahtar Kelimeler: Kablosuz Algılayıcı Ağlar, Bulanık Mantık, Bulanık Kümeleme, Değişken Yarıçaplı Kümeleme, Olasılıklı Kümeleme

To my family...

ACKNOWLEDGMENTS

I am heartily thankful to my supervisor, Prof.Dr. Adnan Yazıcı, for his professional guidance and suggestions throughout my thesis studies.

I would like to show my gratitude to jury members Prof. Dr. Müslim Bozyiğit, Assoc. Prof. Dr. Ahmet Coşar, Assoc. Prof. Dr. Halit Oğuztüzün and Asst. Prof. Dr. İbrahim Körpeoğlu for reviewing and evaluating my thesis.

I would also like to thank TÜBİTAK UEKAE / İLTAREN for deeply supporting my graduate studies.

I thank to my wife and my son for their patience throughout my thesis studies.

Finally, I would like to thank my parents and my brother for their love, and endless support and guidance during my educational life.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

ALGORITHMS

LIST OF ABBREVATIONS

WSN	Wireless Sensor Network
MEMS	Micro-Electro-Mechanical Systems
EEUC	Energy-Efficient Unequal Clustering
LEACH	Low-Energy Adaptive Clustering Hierarchy
CHEF	Cluster Head Election Mechanism Using Fuzzy Logic
EAUCF	Energy-Aware Unequal Clustering with Fuzzy Logic
HEED	Hybrid Energy-Efficient Disributed Clustering
COA	Center of Area
GUI	Graphical User Interface
FND	First Node Dies
HNA	Half of the Node Alive
LND	Last Node Dies

CHAPTER 1

INTRODUCTION

There have been recent advances in micro-electro-mechanical systems (MEMS) technology, wireless communications, and digital electronics. These advances have enabled the development of low-cost, low-power, multifunctional sensor nodes that are small in size and communicate with each other using radio frequencies [1]. A single sensor node has limited capability in sensing and is not sufficient for gathering useful information from a specific domain. This data gathering process can be accomplished by the collective work of a number of sensor nodes. In many applications the number of sensor nodes could be hundreds or thousands. These collaboratively working sensor nodes form a network which is called a wireless sensor network (WSN).

Wireless sensor networks have plenty of advantages. The deployment of WSNs are easier and faster than the wired sensor networks or any other wireless networks [10], because they do not need any fixed infrastructure [22]. Since sensor nodes are densely deployed in most of the cases, they are able to tolerate the network failures. Wireless sensor networks do not require a central organisation and they are self-configuring [10].

There are several types of wireless sensors such as seismic, low sampling rate magnetic, thermal, visual, infrared, acoustic and radar sensors [1]. These sensor nodes can monitor various environmental conditions. Some of these conditions are temperature, pressure, humidity, soil makeup, vehicular movement, noise levels, lighting conditions, the presence or absence of certain kinds of objects and mechanical stress levels on attached objects [3].

Wireless sensor networks have various types of applications. Foremost WSN applications are military applications, environmental applications, health applications, home applications and other commercial applications [1]. Brief descriptions of some of these WSN applications are listed below:

- Military Applications: Properties such as fault tolerability, rapid deployment and self organization make WSNs useful in military applications. They may be used in monitoring and tracking friendly forces, battlefield surveillance, nuclear, chemical and biological attack detection.
- Environmental Applications: WSNs can be used for monitoring and tracking animals, detecting forest fires and floods, large-scale earth monitoring and planetary exploration.
- Health Applications: Some of the health areas that WSN applications may be used are patient monitoring, telemonitoring of human physiological data, and drug administration.
- Home Applications: WSNs can be used in home automation systems. The domestic devices may interact with each other using wireless communication.
- Other Commercial Applications: WSN applications can be developed for monitoring material fatigue, managing inventory, monitoring product quality, detecting and monitoring car thefts, factory process control and automation.

In wireless sensor networks, each sensor node receives signal from a limited region. This signal is processed in that sensor node and sensed information is generally transmitted to the observers (e.g. base stations) [21]. Sensor nodes consume energy while receiving information, processing information and transmitting information. In most of the cases, these sensor nodes are equipped with batteries which are not rechargeable. Therefore, energy efficiency is a major design goal in wireless sensor networks [21].

Nodes can be partitioned into a number of small groups, called clusters, for aggregating data through efficient network organization [21]. In general, each cluster has a cluster-head which coordinates the data gathering and aggregation process in a particular cluster. Each cluster member forwards its data packets to the cluster-head. Clustering in wireless sensor networks guarantees basic performance achievement with a large number of sensor nodes [17] [2]. In other words, clustering improves the scalability of wireless sensor networks [14]. This is because clustering minimizes the need for central organization and promotes local decisions. The major benefits of clustering in wireless sensor networks are listed [22] below:

- Clustering provides the spatial reuse of resources to increase system capacity. For example, if the clusters are not neighbors, they can use the same frequency for wireless communication.

- Routing information of a cluster is shared with only other cluster-heads or cluster gateways. This restriction reduces the number of transmissions performed for distributing routing information. By using this advantage of clustering, more energy efficient routing protocols have been implemented.

- When cluster structure is used in a WSN, the local changes need not be reflected to entire network. This reduces the information processed by sensor nodes and data stored in sensor nodes.

There have been substantial amount of research on clustering protocols for WSNs. These clustering protocols are classified according to different criteria. The classification of clustering protocols according to their objectives is given [22] below:

- Dominating-set-based clustering: This type of clustering protocols try to find a weakly connected dominating set which is responsible for searching route and maintaining routing table. Thus, table-driven routing and on-demand routing can be applied easily.

- Low-maintenance clustering: This type of clustering protocols aim to provide a stable cluster structure to upper layer protocols. To achieve this goal, they try to limit re-clustering situations or reducing the control messages for clustering.

- Mobility-aware clustering: Mobility-aware clustering protocols take the mobility of sensor nodes into consideration. They try to group the mobile nodes that move with similar speed. The clusters that consist of mobile nodes moving with similar speed build a more stable cluster structure for wireless sensor networks.

- Energy-efficient clustering: Energy-efficient clustering protocols try to use the battery energy of the sensor nodes more wisely, because sensor nodes have limited battery energy, and they are generally not rechargeable. Energy consumption of sensor nodes can be reduced by eliminating redundant energy consumption and balancing the energy usage of sensor nodes over the network. The main goal of this type of clustering protocols is prolonging the network lifetime.

- Load-balancing clustering: This type of clustering protocols try to limit the number of sensor nodes in each cluster. This approach produces clusters with similar sizes. If the clusters are similar in size, loads can be more evenly distributed within each cluster.
- Combined-metrics-based clustering: As the name implies, this type of clustering protocols consider different metrics together. These metrics can be node degree, battery energy, cluster size, mobility speed, etc. These types of metrics are generally used in cluster-head election phase of clustering protocols.

Most of the clustering algorithms utilize two techniques which are selecting cluster-heads with more residual energy and rotating cluster-heads periodically to balance energy consumption of the sensor nodes over the network [13]. These clustering algorithms do not take the location of the base station into consideration. This lack of consideration causes the hot spots problem in multi-hop wireless sensor networks. The cluster-heads near the base station die earlier, because they will be in a heavier relay traffic than the cluster-heads which are relatively far from the base station. In order to avoid this problem, some unequal clustering algorithms are proposed in literature [18][13]. In unequal clustering, the network is partitioned into clusters with different sizes. The clusters close to the base station are smaller than the clusters that are far from the base station. EEUC (Energy-Efficient Unequal Clustering) mechanism for periodical data gathering partitions the sensor nodes into clusters of unequal size, and clusters closer to the base station have smaller size.

In order to balance energy consumption of cluster-heads, a periodically rotating cluster-head mechanism is firstly proposed by Heinzelman *et al.*, namely LEACH (Low-Energy Adaptive Clustering Hierarchy). LEACH is a clustering algorithm that utilizes randomized rotation to balance energy consumption of cluster-heads over the network [9]. Randomized periodical rotation property of LEACH is used in many clustering algorithms. Although periodical rotation is a vital property for clustering algorithms, it is not sufficient by itself. Most of the clustering algorithms, such as EEUC and CHEF, use periodical rotation as a base property and build their approach on top of it.

Various uncertainties may arise while partitioning wireless sensor networks into clusters. Fuzzy set theory is exclusively useful to model uncertainty. In general, it might also be the most appropriate way to model uncertainty for well-defined situations [23]. Some fuzzy clustering algorithms, such as CHEF [11] and the algorithm of Gupta *et al.* [5], are proposed for handling uncertainties in clustering. CHEF (Cluster Head-Election with Fuzzy) is a localized cluster-head election mechanism that uses fuzzy logic to maximize the lifetime of the WSN [11]. On the other hand, the algorithm that is proposed by Gupta *et al.* is a centralized cluster-head election mechanism. Thus, it requires gathering clustering information from sensor nodes to the base station.

In this thesis, a fuzzy energy-aware unequal clustering approach (EAUCF) is introduced to make a further improvement in maximizing the lifetime of the WSN. EAUCF is a distributed competitive aglorithm. It selects the cluster-heads via energy-based competition among the tentative cluster-heads which are selected using a probabilistic model. EAUCF mostly focuses on wisely assigning competition ranges to the tentative cluster-heads. In order to make wise decisions, it utilizes the residual energy and the distance to the base station parameters of the sensor nodes. In addition to this, EAUCF uses fuzzy logic to handle uncertainties in competition range estimation.

LEACH protocol rotates the cluster-heads periodically in order to balance energy consumption. It uses a pure probabilistic model to elect cluster-heads. CHEF, EEUC and EAUCF also utilize randomized periodical rotation. However, they do not elect the final cluster-heads by using a pure probabilistic model. They rotate the tentative cluster-heads periodically, but not the actual cluster-heads. CHEF, EEUC and EAUCF are competitive clustering algorithms. They elect the actual cluster-heads via competition among the tentative cluster-heads. The competition of tentative cluster-heads are based on the residual energy and local distance parameters. On the other hand, EEUC and EAUCF only employ the residual energy levels of the tentative cluster-heads in order to elect the actual cluster-head.

EAUCF, CHEF and the approach of Gupta *et al.* utilize fuzzy logic for handling uncertainties in clustering. CHEF and the approach of Gupta *et al.* assign chances to the sensor nodes using the results which are inferred from the predefined fuzzy if-then mapping rules. These chances are used in cluster-head competition. However, EAUCF employs fuzzy logic for wisely adjusting the competition ranges of the tentative cluster-heads. The approach of Gupta *et al.* is a centralized cluster-head election mechanism. Thus, in order to elect cluster-heads, the base station has to collect clustering information from all sensor nodes in WSN [11]. However, this is a very costly task for a WSN, and it is hard to repeat cluster-head election process periodically. LEACH, CHEF and EAUCF clustering algorithms are localized and distributed algorithms. Since they elect cluster-heads locally, they do not need to forward clustering information to the base station. Thus, they elect the cluster-heads with a lower energy cost. Morover, it is easy to repeat cluster-head election periodically in LEACH, CHEF and EAUCF.

EAUCF is an unequal clustering algorithm like EEUC. EEUC assigns unequal competition ranges to the tentative cluster-heads considering only the distance to the base station parameter. However, EAUCF utilizes both the residual energy and the distance to the base station parameters of the tentative cluster-heads for estimating competition ranges. EAUCF assigns greater competition ranges to the tentative cluster-heads which have higher residual energy levels, because they can serve to a larger region. This significant property of EAUCF puts it forward considering LEACH, CHEF and EEUC algorithms.

The rest of the thesis is organized as follows. In the next chapter, we give information about studies that are related to probabilistic clustering, fuzzy clustering and unequal clustering. In chapter 3, we describe LEACH, CHEF and EEUC algorithms in detail. In that chapter, we also introduce our clustering algorithm EAUCF. In chapter 4, we evaluate our proposed algorithm by comparing LEACH, CHEF and EEUC and provide the detailed evaluation results. Finally, we conclude the thesis and discuss some possible future works.

CHAPTER 2

RELATED WORK

There are several proposed clustering algorithms for WSNs in recent years. In this section, we review probabilistic clustering algorithms, fuzzy clustering algorithms and unequal clustering algorithms.

2.1 Probabilistic Clustering Algorithms

In probabilistic clustering approaches, each node in the wireless sensor network decides its role by itself. This type of clustering algorithms aim to minimize the communication between sensor nodes. Probabilistic clustering algorithms guarantee rapid convergence and provide balanced cluster sizes [21]. Basically, each node assigns itself a probability which is a number between 0 and 1. If this probability is less than a predefined threshold, then that node becomes a cluster-head. Based on this principle, various probabilistic clustering algorithms are proposed. Here we overview LEACH [9], HEED (Hybrid Energy-Efficient Distributed Clustering) [20] and the algorithm proposed by Kuhn *et al.* [12].

The objective of LEACH protocol is to minimize energy dissipation in sensor networks. LEACH has distributed coordination and control mechanisms for cluster set-up and operation processes [9]. Static clustering algorithms select cluster-heads for WSNs only once, and these cluster-heads operate as cluster-head until they die. Since cluster-heads consume much more energy than ordinary sensor nodes, energy consumption over the network cannot be distributed evenly by using static clustering. Therefore, WSN can quickly move to a useless state, because the number of cluster-heads decreases drastically. In LEACH protocol, cluster-heads are rotated in randomized manner, and cluster-head election is done periodically. The interval between two consecutive cluster formation process is called as *round*. A single round consits of two phases which are set-up and steady-state phases [7]. The cluster-head election and cluster formation are done during set-up phase. In steady-state phase, the data, which is gathered from cluster member nodes, is aggregated at local cluster-head and transmitted to the base station. We compare our approach EAUCF with this well-known probabilistic clustering protocol, because EAUCF uses randomization for selecting tentative cluster-heads in each round.

In HEED protocol, residual energy of each sensor node is the primary parameter for probabilistic election of cluster-heads [21]. As stated in [20], there are four primary goals of HEED. These are listed below:

- Prolonging the lifetime of the wireless sensor network by evenly distributing energy consumption
- Selecting cluster-heads in a constant number of iterations
- Minimization of control overhead
- Formation of well-distributed cluster-heads and compact clusters

In case of a tie in cluster-head election, node degree or average distance to neighbors parameters are used to determine the clusted-head. HEED protocol is implemented in TinyOS, which is an operating system developed for Berkeley motes. Experimentations that are employed for evaluating HEED protocol show that clustering and data aggregation at least double the lifetime of the wireless sensor network [21].

Kuhn *et al.* studied initializing newly deployed ad hoc and sensor networks, and proposed a probabilistic cluster-head election algorithm. In this approach, the probability of each node depends on the node degree [21]. Kuhn *et al.* showed that their proposed clustering approach computes an asymptotically optimal clustering in polylogarithmic time [12]. This algorithm tries to find a dominating set of nodes which will be assigned as cluster-heads. Sensor nodes compete to become dominators by exponentially incrementing their sending probability on a specified channel. Three different channels are used in this algorithm. Remaining two channels are used to keep the number of dominators small in a vicinity of an emerging dominator [12].

Pure probabilistic clustering algorithms (e.g. LEACH) have some disadvantages, which are listed [11] below:

- Since pure probabilistic clustering algorithms only depend on probability, they can produce cluster-heads closer to each other.
- They do not consider the residual energy of the sensor nodes. Therefore, the nodes that have lower energy levels may become cluster-heads.
- These algorithms may randomly elect cluster-heads in vicinities that have low node density.

Pure probabilistic clustering approaches are useful for cluster-head election, but they are not sufficient. In order to make a more accurate cluster-head election, some additonal parameters such as node degree, residual energy and local distance should be taken into consideration.

2.2 Fuzzy Clustering Algorithms

Fuzzy logic is useful for making real time decisions without needing complete information about the environment. On the other hand, conventional control mechanisms generally need accurate and complete information about the environment [5]. Fuzzy logic can also be utilized for making a decision based on different environmental parameters by blending them according to predefined rules.

Some of the clustering algorithms employ fuzzy logic to handle uncertainties in the wireless sensor networks. Basically, fuzzy clustering algorithms use fuzzy logic for blending different clustering parameters to elect cluster-heads. They assign chances to tentative cluster-heads according to the defuzzified output of fuzzy if-then rules. The tentative cluster-head becomes a cluster-head if it has the greatest chance in its vicinity. There are distributed and centralized fuzzy logic clustering approaches. Here we are going to overview the centralized approach of Gupta *et al*. [5] and the distributed approach of Kim *et al*. [11] which is abbreviated as CHEF.

In the fuzzy clustering approach proposed by Gupta *et al*., the cluster-heads are elected at the base station. In every round, each sensor node forwards its clustering information to the base station. There are three fuzzy descriptors which are considered by the base station during cluster-head election. These fuzzy descriptors are node concentration, residual energy in each node and node centrality [5]. The definitions of these fuzzy descriptors are given [5] below:

- Node Concentration: Number of the nodes in the vicinity
- Residual Energy: Remaining battery energy of each sensor node
- Node Centrality: A parameter that indicates how central the node is to the cluster

There are 27 fuzzy if-then rules which are defined at the base station. The base station elects the cluster-heads according to these fuzzy rules. After the base station elects the cluster-head, it forwards the election results to entire network. This algorithm is a centralized clustering algorithm, because all clustering decisions are made at the base station. Gupta *et al*. claims that a centralized clustering approach will produce more accurate cluster-heads, because the base station has all clustering information about the network and base stations are more powerful than ordinary nodes [5]. However, this centralized approach have some disadvantages [11]:

- The base station must collect all clustering information from the network. Repeating this process in every round brings a high overhead to sensor nodes. Thus, the battery levels of the sensor nodes may run low quickly.
- In this approach the simulation is done for electing only one cluster-head per round. Therefore, this simulation is not a realistic one.

CHEF is a similar approach to that of Gupta *et al*. [5], but it performs cluster-head election in a distributed manner. Cluster-head election is done locally. Thus, the base station does not need to collect clustering information from all sensor nodes [11]. In every round, each node generates a random number between 0 and 1. If the random number is smaller than the predefined threshold, then that node becomes a tentative cluster-head. There are two fuzzy descriptors that are used in cluster-head election. These are residual energy of each node and local distance. Local distance is the total distance between the tentative cluster-head and the nodes within predefined constant radius r. There are 9 fuzzy if-then rules that are defined in all sensor nodes. Tentative cluster-heads calculate their chances to be a cluster-head using these fuzzy rules. If the chance of a tentative cluster-head is greater than the other tentative cluster-heads' chances in radius r, then that tentative cluster-head becomes an actual cluster-head.

Afterwards, it sends a cluster-head advertisement message to the nodes in the vicinity. The nodes that are not elected as cluster-head join to the closest cluster by sending a message to that cluster-head. CHEF guarantees that any two cluster-heads cannot exist within *r* distance [11]. We compare our approach EAUCF with CHEF, because this approach is a well-known distributed fuzzy approach.

2.3 Unequal Clustering Algorithms

The sensor nodes closer to the base station consume more energy, because the network traffic increases as we get close to the base station [21]. Therefore, the nodes closer to the base station quickly run out of battery. In order to balance energy consumption over the network, unequal clustering approach is introduced. This approach is based on the idea of decreasing the cluster sizes as we get close to the base station. If a cluster-head closer to the base station has less intra-cluster work, then it can contribute to inter-cluster data forwarding more. Unequal clustering is meaningful even in the cases where each cluster-head forwards its aggregated data to the base station directly. Here, we overview two unequal clustering approaches. These are the approaches that are proposed by Shu *et al*. [18] and Li *et al*. [11] which is abbreviated as EEUC.

If a cluster-head is closer to the base station, it has to relay more data forwarding traffic than the sensor nodes which are far from the base station [18]. Each sensor node in the network tries to send its data to the base station. Therefore, as we get close to the base station, the data forwarding traffic increases. Shu *et al*. proposed an approach that aims to achieve optimal power allocation over the sensor network. This approach assigns larger cluster sizes to cluster-heads that take less role in data forwarding process. This approach is illustrated in Figure 2.1. The proposed network model in this approach assumes a circular sensing region. However, generally sensor nodes are deployed randomly by throwing them to the target region. Therefore, this approach is not a practical one for real environments in most of the cases. This model should be improved to handle non-circular regions.

EEUC is a distributed competitive unequal clustering algorithm where cluster-heads are elected by local competition [11]. Every node has a preassigned competitive range. This range gets smaller as we get close to the base station. This makes EEUC an unuequal clustering algo-

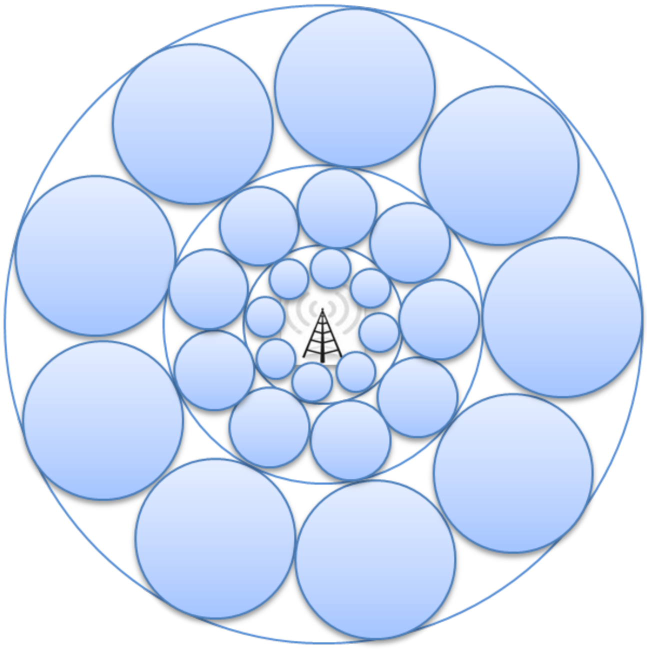

Figure 2.1: Cluster size distribution of Shu *et al*. approach

rithm. EEUC algorithm is also a probabilistic clustering algorithm, because in each cluster formation round, each node generates a random number between 0 and 1 to decide whether it is going to participate to the cluster-head election competition or not. If a sensor node has decided to participate to the competition, then it becomes a tentative cluster-head. Tentative cluster-heads in local regions compete in order to become an actual cluster-head. This competition is based on the residual energy of each tentative cluster-head. After cluster-head election is completed, the remaining sensor nodes join to the closest cluster. We compare our approach with EEUC, because EEUC is a recent and well-structured unequal clustering algorithm.

CHAPTER 3

CLUSTERING ALGORITHMS

In this chapter, we propose an algorithm for cluster-head election in wireless sensor networks. We also provide the details of three different clustering algorithms. We are going to compare our approach with these clustering algorithms. These algorithms are LEACH [9], CHEF [11] and EEUC [13].

3.1 Preliminaries

Before describing clustering algorithms in detail, we introduce the characteristics of the system model that we use in our implementations. First, we list the assumptions that we make about the network model:

- Sensor nodes are deployed randomly.
- All sensor nodes and the base station are stationary after deployment phase.
- Nodes have the capability of adjusting the transmission power according to the distance of the receiver nodes.
- The distance between nodes can be computed based on the received signal strength. Therefore, there is no need for sensor nodes to know their exact locations.
- All sensor nodes have the same amount of energy when they are initially deployed.
- Base station need not to be located far away from the sensing region.
- All sensor nodes are identical.

The first order radio model that is shown in [9] is used for energy dissipation model in simulations. Equation 3.1 represents the amount of energy consumed for transmitting l bits of data to d distance. E_{elec} is the energy consumption per bit in the transmitter and the receiver circuitry. ϵ_{amp} is the energy dissipated per bit in the RF amplifier.

$$E_{Tx}(l, d) = lE_{elec} + l\epsilon_{amp}d^2 \tag{3.1}$$

Equation 3.2 represents the amount of energy consumed for receiving l bits of data.

$$E_{Rx}(l) = lE_{elec} \tag{3.2}$$

3.2 LEACH Clustering Protocol

In this section, we describe LEACH (Low-Energy Adaptive Clustering Hierarchy) protocol proposed by Heinzelman *et al*. LEACH is a well-known cluster-head election approach that constitutes a basis for many other approaches [8] [7] [19] as stated in [4]. It is the first significant protocol that aims to minimize the overall energy used in data gathering operations in wireless sensor networks [4].

LEACH is a distributed algorithm which makes local decisions to elect cluster-heads. If the cluster-heads are selected for once and do not change throughout the network lifetime, then it is obvious that these static cluster-heads die earlier than the ordinary nodes. Therefore, LEACH includes randomized rotation of cluster-head locations to evenly distribute the energy dissipation over the network [9]. LEACH also performs local data compression in cluster-heads to decrease the amount of data that is forwarded to the base station.

In LEACH, cluster-head election is done periodically to enable randomized rotation of cluster-heads. Every round consists of two phases, namely set-up phase and steady-state phase. In set-up phase, cluster-heads are elected and clusters are formed. In steady-state phase, data transfers to the base station are performed through the clustered network. A particular sensor node decides whether it is going to become a cluster-head or not by generating a random number between 0 and 1. If this number is less than the predefined threshold $T(n)$, then the sensor node becomes a cluster-head. G represents the set of sensor nodes that have not been cluster-heads in the last $\frac{1}{P}$ rounds where P is the desired percentage of cluster-heads. r represents the current round number. Using these parameters, $T(n)$ is formulated as follows:

If the sensor node n belongs to G:

$$T(n) = \frac{P}{1 - P * (r \mod \frac{1}{P})} \tag{3.3}$$

If the sensor node n does not belong to G, then the $T(n)$ is set to 0. Thus, n cannot become a cluster-head. At round 0, the propability of becoming a cluster-head for each node is equal to P. However, this situation changes in the following rounds. The cluster-heads of round 0 cannot become cluster-heads during the following $\frac{1}{P}$ rounds. This restriction prevents a particular node to become a cluster-head frequently. However, this restriction brings a drawback. It causes rapid decrease in the number of cluster-heads. To handle this drawback, as r increases, the chance of the remaining sensor nodes to be a cluster-head is also increased by adjusting the threshold $T(n)$ for the remaining sensor nodes. This critical balance is a significant property of LEACH.

After cluster-heads are elected for a particular round, each cluster-head broadcasts an advertisement message to the remaining sensor nodes. As each non-cluster-head node receives these advertisement messages, they decide the cluster to which they belong. Each non-cluster-head joins to the cluster from which it has received the largest signal strength. In order to join to the selected cluster, it transmits a $JoinClusterHeadMessage$ to that cluster. Once all the cluster-heads are selected and the clusters are formed, data transmission continues up to the next round. Pseudo-code for cluster-head election and cluster formation procedure for a single sensor node is given in Algorithm 3.1.

The simulations in [9] showed that LEACH reduces communication energy as much as 8 times as compared to direct transmission. In other words, the first node death in LEACH occurs 8 times later than the first node death in direct transmission. Since we compare our proposed algorithm with LEACH, we have developed a LEACH simulation. Figure 3.1 and Figure 3.2 shows two different cluster-head distribution examples over the network for two different particular rounds. In both of the examples, the number of deployed sensor nodes is 200. The desired percentages of cluster-heads are 0.05 and 0.1, respectively for the examples in Figure 3.1 and Figure 3.2.

```
P ← desired percentage of cluster-heads
currentRound ← currentRound + 1
nodeState ← CLUSTERMEMBER
clusterMembers ← empty
myClusterHead ← this
if notClusterHeadCount < 1/P then
    notClusterHeadCount ← notClusterHeadCount + 1
else
    T ← threshold for current round that is calculated by Equation 3.3
    μ ← rand(0,1)
    if μ < T then
        nodeState ← CLUSTERHEAD
        notClusterHeadCount ← 0
        Advertise ClusterHeadMessage(ID)
    end if
end if
On receiving all ClusterHeadMessages
if nodeState = CLUSTERMEMBER then
    myClusterHead ← the closest cluster-head
    Send JoinClusterHeadMessage(ID) to the closest cluster-head
end if
On receiving JoinClusterHeadMessage from node N
if nodeState = CLUSTERHEAD then
    add node N to the clusterMembers list
end if
```

Algorithm 3.1: Clustering Algorithm of LEACH protocol

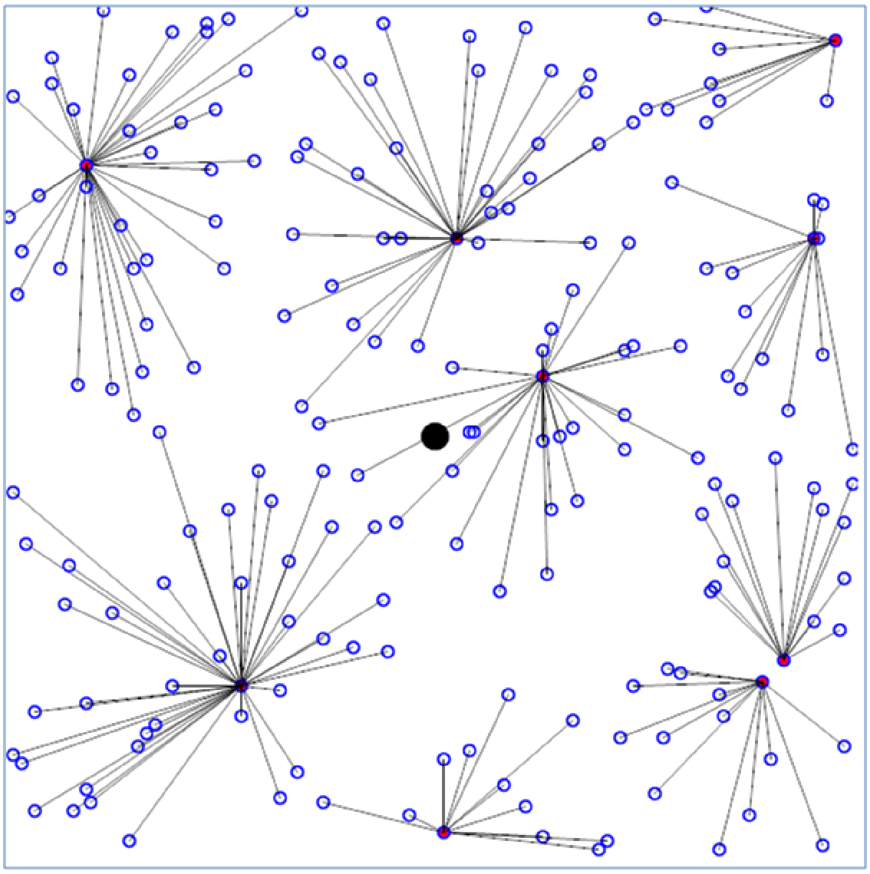

Figure 3.1: Cluster-head distribution example for LEACH with $P = 0.05$.

3.3 CHEF Clustering Protocol

In this section, we describe CHEF (Cluster-Head Election with Fuzzy) protocol proposed by Kim *et al*. The motivation behind this protocol is that using fuzzy logic can reduce gathering data and calculating overheads [11]. Thus, the lifetime of the sensor network can be prolonged. This clustering protocol is aimed to overcome the significant defects of LEACH, which are caused by its pure probabilistic characteristics. Cluster-heads produced by LEACH may be too close to each other and they may be located at the edges of the WSN [11]. To overcome these defects, CHEF takes two parameters into account which are the residual energy of each sensor node and the local distance.

CHEF is also a distributed algorithm like LEACH which makes local decisions to select cluster-heads. CHEF also uses a threshold for electing tentative cluster-heads. However, this threshold does not change in every round as in the case of LEACH. The optimal threshold

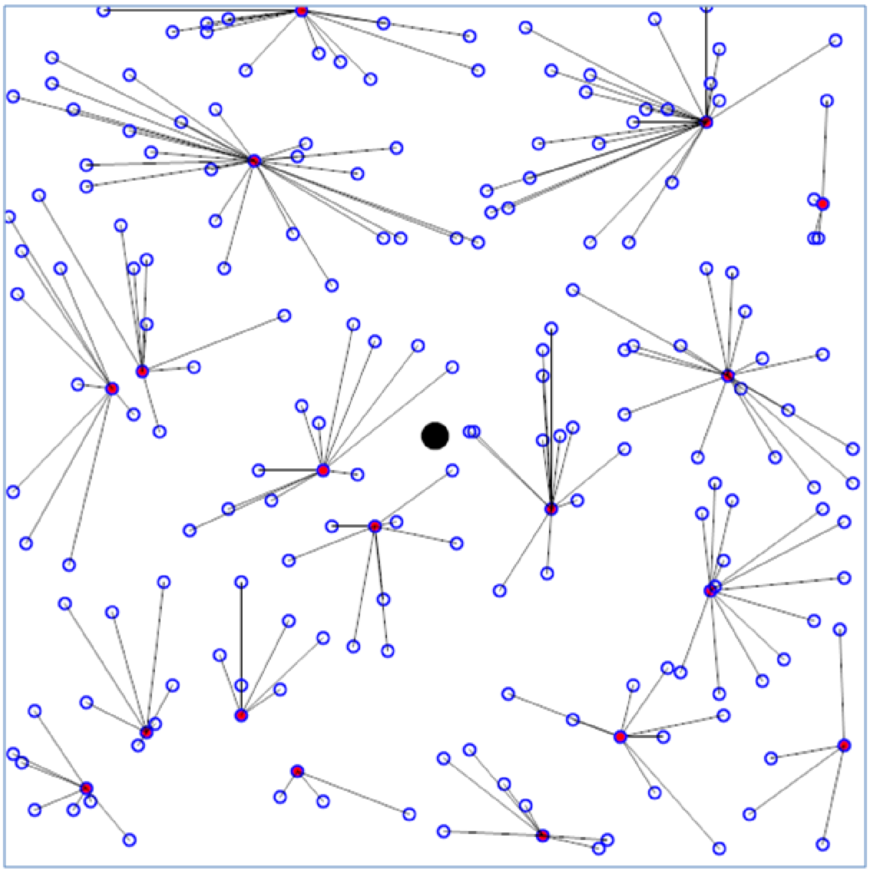

Figure 3.2: Cluster-head distribution example for LEACH with $P = 0.1$.

P_{opt} for CHEF is defined in Equation 3.4.

$$P_{opt} = \alpha \cdot P \tag{3.4}$$

P represents the desired percentage of cluster-heads. α is a constant value which represents the ratio of the candidate for cluster-head. The optimal value of P for a particular WSN is intoduced in [6]. The optimal P can be calculated by using Equation 3.5.

$$P = \frac{\sqrt{n}}{\sqrt{2\pi}} \cdot \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \cdot \frac{\sqrt{A}}{(0.765 \times \sqrt{A} \times 0.5)^2} \cdot \frac{1}{n} \tag{3.5}$$

A represents the area of the wireless sensor network. ϵ_{fs} and ϵ_{mp} denote the amount of energy per bit consumed in the RF amplifier [11] with respect to d_0. If the transmission distance is smaller than d_0, then ϵ_{fs} is used for the amount of energy dissipated. Otherwise, ϵ_{mp} is used. d_0 can be calculated by using Equation 3.6.

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \tag{3.6}$$

The optimal cluster-head radius r for CHEF is fixed and calculated by using Equation 3.7. In this equation n represents the total number of sensor nodes in the WSN.

$$r = \sqrt{\frac{A}{\pi \cdot n \cdot P}} \tag{3.7}$$

In every round, each qualifying sensor node, which is the node that generates a random number less than P_{opt}, calculates its chance for becoming a cluster-head by blending its residual energy and local distance values using predefined fuzzy if-then mapping rules. Local distance is defined in [11] as the sum of distances between a particular node and the nodes within r distance. The fuzzy mapping rules for CHEF is listed in Table 3.1.

A particular sensor node generates a random number between 0 and 1, and checks whether this number is less than P_{opt} or not. If this condition is satisfied, then the particular node becomes a tentative cluster-head. Tentative cluster-heads compete with each other locally to

```
P ← desired percentage of tentative cluster-heads
nodeState ← CLUSTERMEMBER
clusterMembers ← empty
myClusterHead ← this
μ ← rand(0,1)
if μ < P_opt then
   Compute the chance using fuzzy if-then mapping rules
   Advertise CandidateClusterHeadMessage(ID, chance)
   On receiving CandidateClusterHeadMessage from node N
   if chance < N.chance  then
      myClusterHead ← N
   end if
   if myClusterHead = this  then
      Advertise ClusterHeadMessage(ID)
      nodeState ← CLUSTERHEAD
      On receiving JoinClusterHeadMessage from node N
      add N to the clusterMembers list
   else
      Send JoinClusterHeadMessage(ID) to node myClusterHead
   end if
else
   On receiving all ClusterHeadMessages
   myClusterHead ← the closest cluster-head
   Send JoinClusterHeadMessage(ID) to the closest cluster-head
end if
```

Algorithm 3.2: Clustering Algorithm of CHEF protocol

Table 3.1: Fuzzy if-then mapping rules for CHEF

Rule No	Energy	Local Distance	Chance
1	Low	Far	Very Low
2	Low	Medium	Low
3	Low	Close	Rather Low
4	Medium	Far	Medium Low
5	Medium	Medium	Medium
6	Medium	Close	Medium High
7	High	Far	Rather High
8	High	Medium	High
9	High	Close	Very High

become an actual cluster-head. During this competition, each tentative cluster-head advertises a message that includes its *chance*, and it waits for advertisements of other tentative cluster-heads in the vicinity. If it receives an advertisement message that includes a higher chance than its own chance, then it quits competition. If it is the node with the greatest chance, then it becomes a cluster-head. The remaining cluster formation process is similar to LEACH in which each ordinary node joins to the closest cluster. Pseudo-code for cluster-head election and cluster formation procedure for a single sensor node is given in Algorithm 3.2.

We have developed a CHEF simulation to compare our proposed algorithm with CHEF. Figure 3.3 and Figure 3.4 show two different cluster-head distribution examples for two different particular rounds. The number of deployed sensor nodes in the examples are 100 and 200, respectively. α constant is set to 2.5 in both of the examples.

CHEF protocol guarantess that any two cluster-heads cannot exist within *r* distance [11] as also seen in Figure 3.3 and Figure 3.4.

3.4 EEUC Clustering Protocol

In this section, we describe EEUC (Energy-Efficient Unequal Clustering) protocol proposed by Li *et al*. EEUC is a distributed competitive clustering algorithm that elects cluster-heads by local competition. This behavior of EEUC is dissimilar to LEACH [13]. The relay traffic increases as we get close to the base station in multi-hop wireless sensor networks. EEUC takes this observation into consideration and tries to decrease the intra-cluster workload of

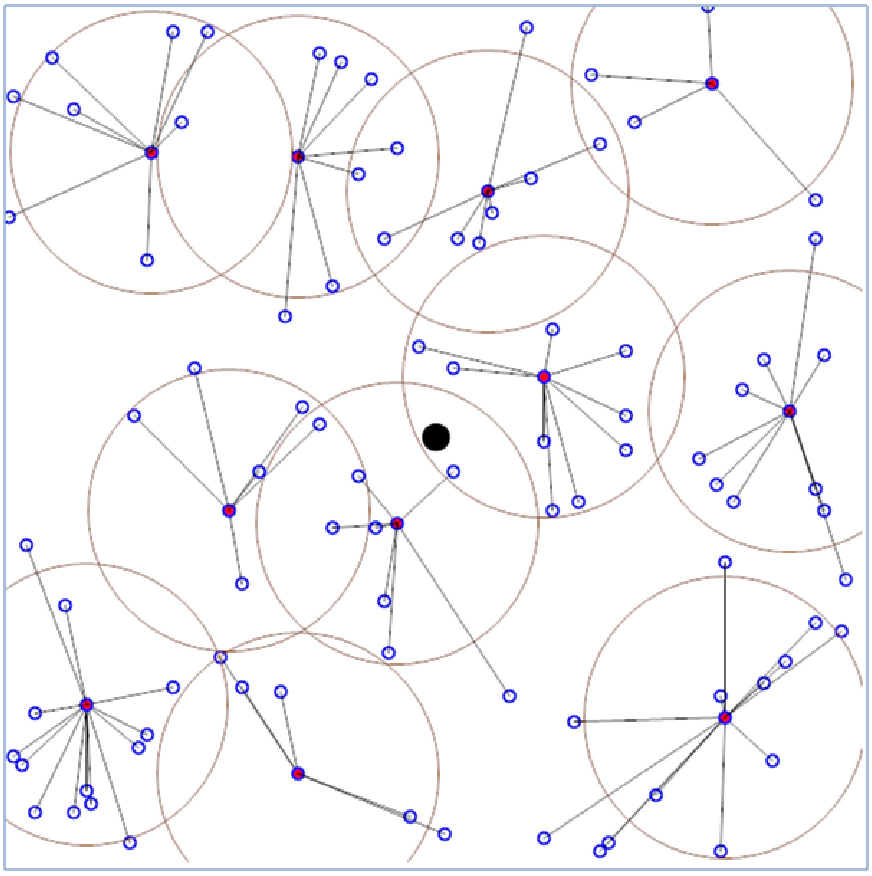

Figure 3.3: Cluster-head distribution example for CHEF where $\alpha = 2.5$ and the number of sensors is 100.

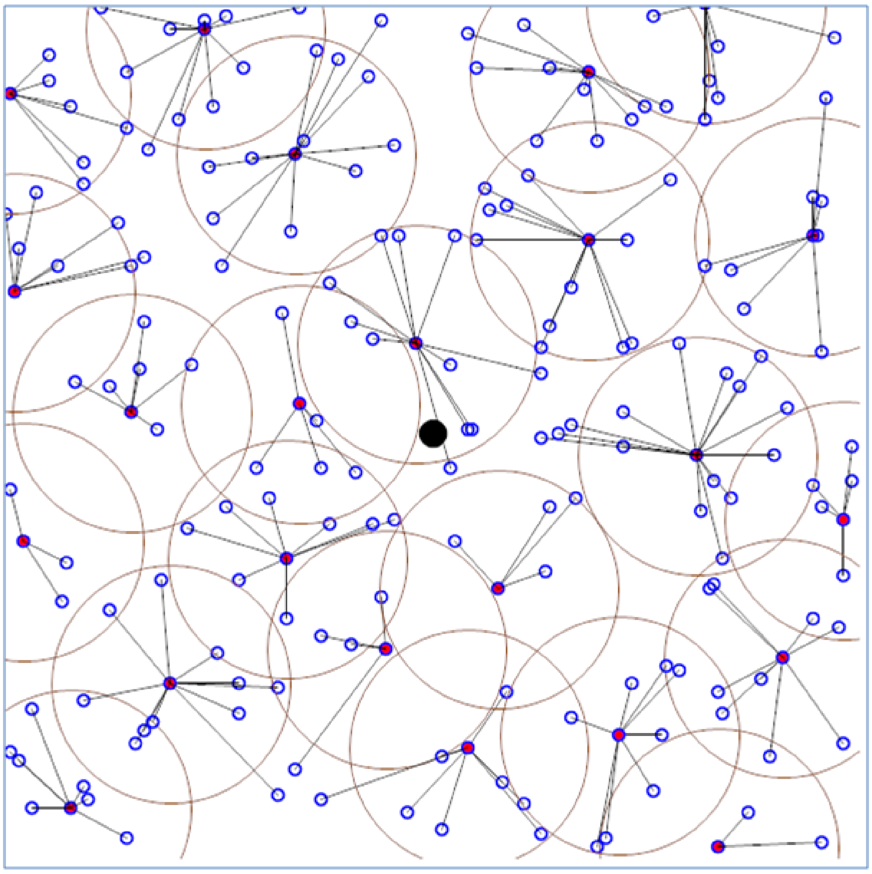

Figure 3.4: Cluster-head distribution example for CHEF where $\alpha = 2.5$ and the number of sensors is 200.

the cluster-heads that are closer to the base station. If a cluster-head has less intra-cluster workload, then it can contribute to inter-cluster communication more. EEUC achieves this goal by assigning smaller competition ranges to the sensor nodes that are closer to the base station. In other words, the competition range of the tentative cluster-head decreases as its distance to the base station decreases [13].

First, EEUC selects tentative cluster-heads which are the candidate sensor nodes for becoming an actual cluster-head. In every round, each sensor node has the same probability T to become a tentative cluster-head. Each sensor node generates a random number between 0 and 1. If this number is smaller than T, then it becomes a tentative cluster-head. After tentative cluster-heads are selected, the remaining sensor nodes sleep until cluster-head election is completed. Each tentative cluster-head has a competition range R_{comp} that is calculated considering its distance to the base station. R_{comp} can be calculated using the Equation 3.8.

$$R_{comp}(s_i) = (1 - c\frac{d_{max} - d(s_i, BS)}{d_{max} - d_{min}})R^0_{comp} \tag{3.8}$$

d_{max} and d_{min} represent the maximum and minimum distances between sensor nodes and the base station. $d(s_i, BS)$ denotes the distance between the node s_i and the base station. R^0_{comp} represents the predefined maximum competitive range. c is a constant coefficient between 0 and 1. We can change the range of the R_{comp} function by adjusting c coefficient.

Similar to CHEF, EEUC guarantees that any two cluster-heads cannot exist within R_{comp} distance. This situation is illustrated in Figure 3.5. In this figure, S_1 and S_2 cannot be both cluster-heads, because S_2 is inside of S_1's competition range. On the other hand, S_3 and S_4 can be elected as cluster-heads together, since neither S_3 nor S_4 are in other's competition range.

Each tentative cluster-head keeps a list of its adjacent tentative cluster-heads which is denoted by S_{CH}. A tentative cluster-head s_j is adjacent to s_i if s_j is in competition range of s_i or s_i is in competition range of s_j. Each tentative cluster-head broadcasts a competition message with a broadcast radius R^0_{comp} to populate its adjacent tentative cluster-heads list. This competition message contains the node ID, competition radius and the residual energy level. Each tentative cluster-head listens for competition messages of other tentative cluster-heads. When a tentative cluster-head receives a competition message from another tentative cluster-head, it

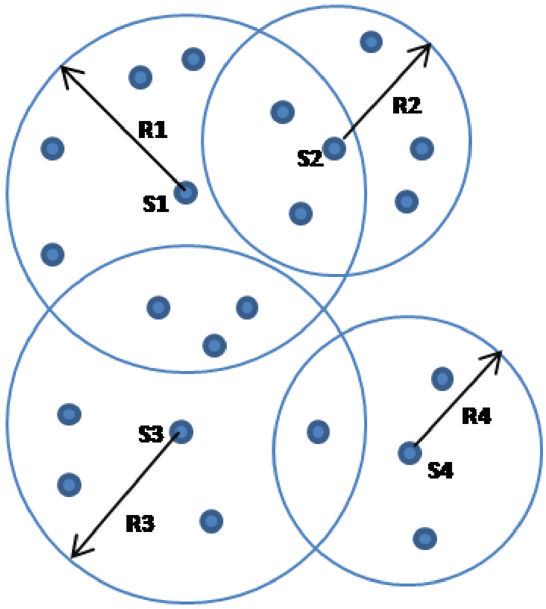

Figure 3.5: The competition ranges of different tentative cluster-heads

checks whether the source tentative cluster-head is adjacent to itself or not. If it is adjacent, then it is added to adjaceny list. After each tentative cluster-head finishes populating its adjaceny list, it makes a decision whether to become an actual cluster-head or quit cluster-head election. If the residual energy of the tentative cluster-head is greater than all the adjacent tentative cluster-heads' residual energy, then it becomes a cluster-head. Otherwise, it quits cluster-head election. After a tentative cluster-head becomes a cluster-head, it broadcasts a final message to advertise its cluster-headship to other sensor nodes in its adjaceny list.

After cluster-head election phase is completed, each cluster-head broadcasts an advertisement message through the network. Ordinary sensor nodes in the network join to the closest cluster just like in LEACH and CHEF. Pseudo-code for cluster-head election and cluster formation procedure for a single sensor node is given in Algorithm 3.3 and 3.4.

Since we compare our proposed algorithm with EEUC, we have developed a EEUC simulation. Figure 3.6 and Figure 3.7 illustrate two different cluster-head distribution examples for two different particular rounds.

$T \leftarrow$ probability to become a tentative cluster-head
$nodeState \leftarrow$ CLUSTERMEMBER
$clusterMembers \leftarrow$ empty
$myClusterHead \leftarrow this$
$S_{CH} \leftarrow$ empty
$\mu \leftarrow$ rand(0,1)
if $\mu < T$ **then**
    $beTentativeHead \leftarrow TRUE$
    Broadcast $CompeteHeadMessage(ID, R_{comp}, residualEnergy)$
else
    On receiving all $ClusterHeadMessages$
    $myClusterHead \leftarrow$ the closest cluster-head
    Send $JoinClusterHeadMessage(ID)$ to the closest cluster-head
    EXIT
end if
On receiving a $CompeteHeadMessage$ from node N
if $d(this, N) < N.R_{comp}$ OR $d(this, N) < this.R_{comp}$ **then**
    Add N to S_{CH}
end if

Algorithm 3.3: Clustering Algorithm of EEUC protocol

while $beTentativeHead = TRUE$ **do**
    if $this.residualEnergy > N.residualEnergy, \forall N \in S_{CH}$ **then**
        Broadcast $FinalHeadMessage(ID)$ to the tentative cluster-heads in the list
        $nodeState \leftarrow$ CLUSTERHEAD
        On receiving $JoinClusterHeadMessage(ID)$ from node N
        add N to the $clusterMembers$ list
        EXIT
    end if
    On receiving a $FinalHeadMessage$ from node N
    if $N \in S_{CH}$ **then**
        Broadcast $QuitElectionMessage(ID)$
    end if
    On receiving a $QuitElectionMessage$ from node N
    if $N \in S_{CH}$ **then**
        Remove N from S_{CH}
    end if
end while

Algorithm 3.4: Clustering Algorithm of EEUC protocol (cont'd.)

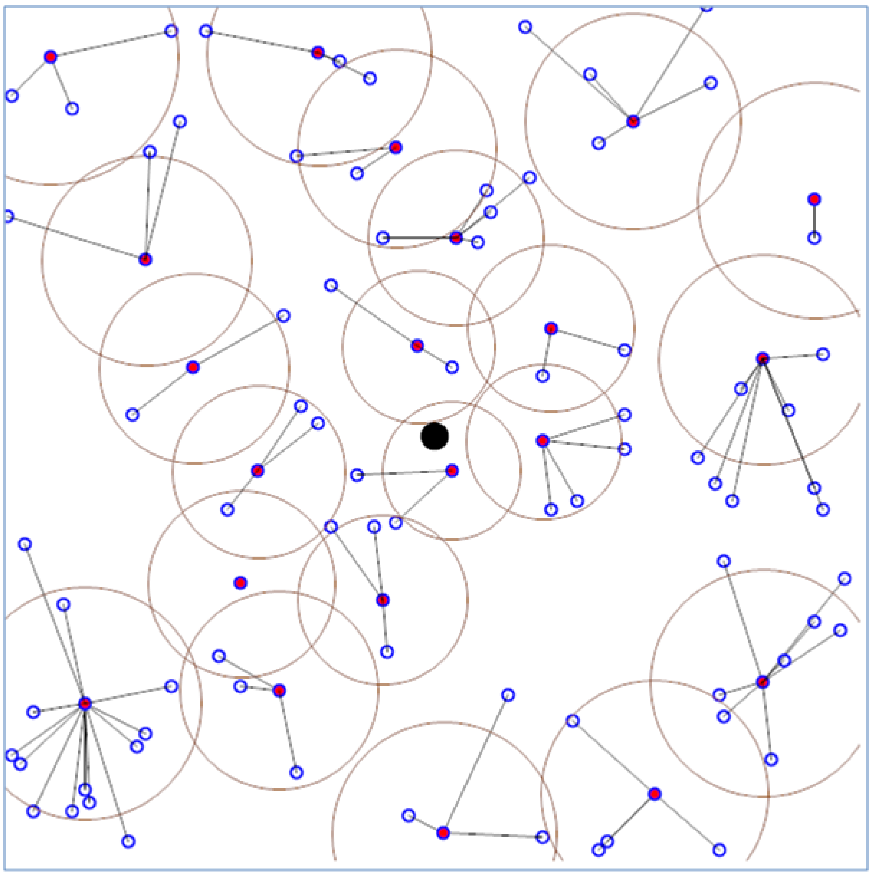

Figure 3.6: Cluster-head distribution example for EEUC where $c = 0.5$, $R^0_{comp} = 30$, $T = 0.4$ and the number of sensors is 100.

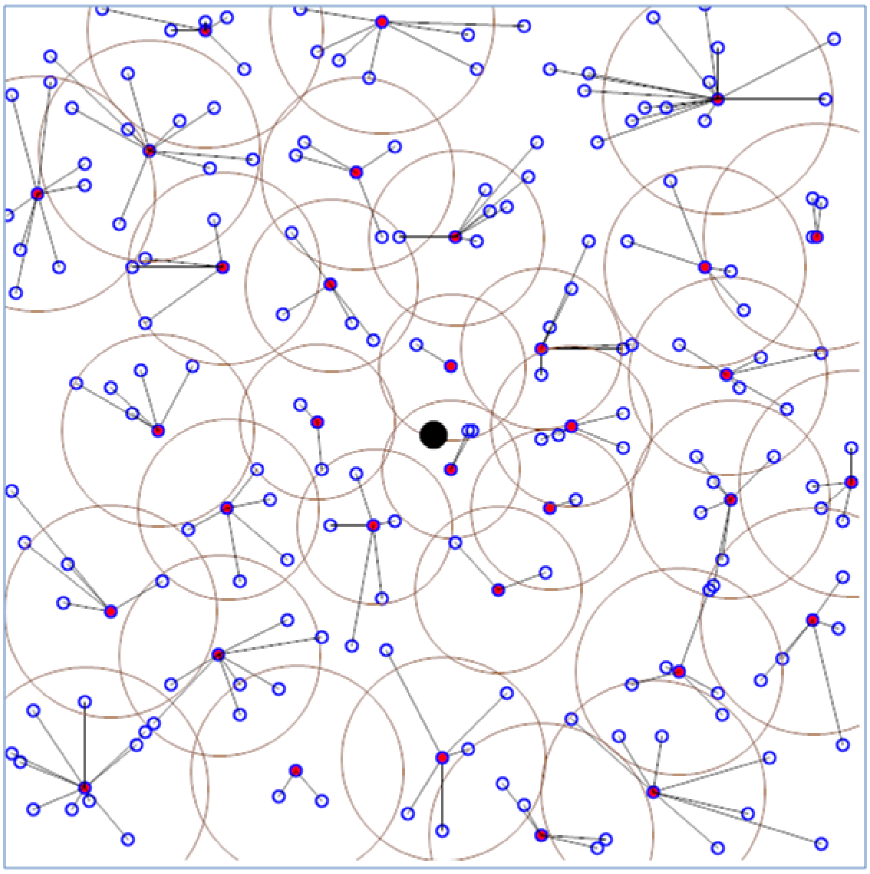

Figure 3.7: Cluster-head distribution example for EEUC where c = 0.5, R^0_{comp} = 30, T = 0.4 and the number of sensors is 200.

3.4.1 EEUC Inter-Cluster Multi-Hop Routing Protocol

We briefly describe the characteristics of EEUC inter-cluster multi-hop routing protocol, because we use it as a multi-hop communication protocol in our simulations. Each cluster-head aggregates the data which comes from its cluster members, and sends it to the base station via multi-hop communication [13]. Li *et al.* proposed a multi-hop routing protocol for inter-cluster communication. The overview of EEUC multi-hop routing protocol for inter-cluster communication is depicted in Figure 3.8 [13].

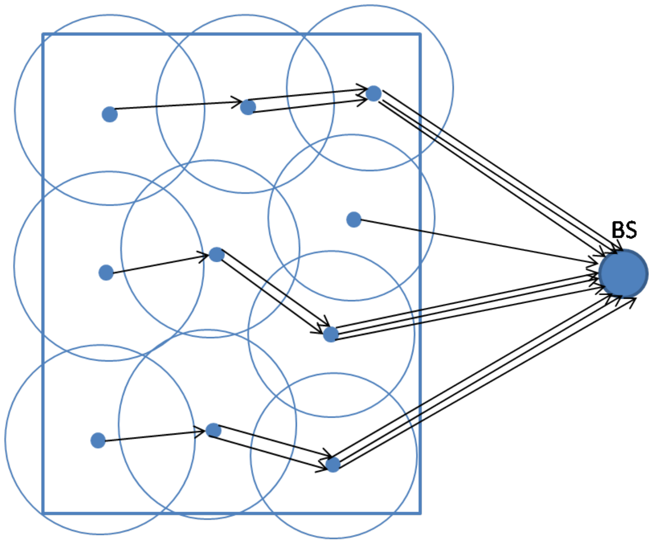

Figure 3.8: EEUC multi-hop routing protocol overview

In EEUC routing protocol, if a particular cluster-head's distance to the base station is smaller than a predefined threshold TD_MAX, it transmits its data packet to the base station directly. Otherwise, that particular node tries to find a forwarding relay node. Each cluster-head keeps a set of candidate forwarding cluster-heads in order to forward its data to the next hop. This candidate set R_{CH} is populated for a particular cluster-head s_i using the set definition in Equation 3.9.

$$s_i.R_{CH} = \{s_j | d(s_i, s_j) \leq k.s_i.R_{comp}, d(s_j, BS) < d(s_i, BS)\} \tag{3.9}$$

k denotes the minimum integer that generates a minumum number of members for R_{CH}. If a proper k cannot be found for s_i, then it forwards its data to the base station via direct transmission.

EEUC routing protocol considers two criteria to select a forwarding node from R_{CH}. These criteria are the residual energy of each forwarding node and the link cost. Link cost that is represented by d^2_{relay} is calculated using the Equation 3.10.

$$d^2_{relay} = d^2(s_i, s_j) + d^2(s_j, BS) \tag{3.10}$$

First, s_i reduces the number of candidates by choosing the two smallest d^2_{relay} nodes. After that, it selects the node that has more residual energy as its relay node among two remaining candidates. Each cluster-head chooses its relay node and trasmits its aggregated data to that relay node. Consequently, the data packets are delivered to the base station through the relay nodes via multi-hop communication.

3.5 EAUCF Clustering Algorithm

In this section, we describe our proposed clustering algorithm EAUCF (Energy-Aware Unequal Clustering with Fuzzy). EAUCF is a distributed competitive unequal clustering algorithm similar to EEUC. It makes local decisions to determine competition radius and to elect cluster-heads. The main difference between EEUC and EAUCF is their competiton radius estimation methods. EEUC only considers distance to the base station parameter to calculate competition radius. However, EAUCF employs both residual energy and distance to the base station parameters of the sensor node. Morover, EAUCF takes advantage of using fuzzy logic to calculate competition radius. CHEF is also a fuzzy approach, but it utilizes fuzzy logic for assigning cluster-head chances to tentative cluster-heads. LEACH protocol rotates the cluster-heads periodically in each round by using a probabilistic model. EAUCF also employs a probabilistic model, but it does not elect the final cluster-heads by just depending on this model. It elects the tentative cluster-heads using this model like CHEF and EEUC. First, we explain the main flow of EAUCF in Algorithm 3.5. After that, we get into details of the algorithm.

In every clustering round, each sensor node generates a random number between 0 and 1. If the random number for a particular node is smaller than the predefined threshold T, which is the percentage of the desired tentative cluster-heads, then that sensor node becomes a tenta-

$T \leftarrow$ probability to become a tentative cluster-head
$nodeState \leftarrow$ CLUSTERMEMBER
$clusterMembers \leftarrow$ empty
$myClusterHead \leftarrow this$
$beTentativeHead \leftarrow TRUE$
$\mu \leftarrow$ rand(0,1)
if $\mu < T$ **then**
    Calculate R_{comp} using fuzzy if-then mapping rules
    Advertise $CandidateClusterHeadMessage(ID, R_{comp}, residualEnergy)$
    On receiving $CandidateClusterHeadMessage$ from node N
    if $this.residualEnergy < N.residualEnergy$ **then**
        $beTentativeHead \leftarrow FALSE$
        Advertise $QuitElectionMessage(ID)$
    end if
end if
if $beTentativeHead = TRUE$ **then**
    Advertise $ClusterHeadMessage(ID)$
    $nodeState \leftarrow$ CLUSTERHEAD
    On receiving $JoinClusterHeadMessage(ID)$ from node N
    add N to the $clusterMembers$ list
    EXIT
else
    On receiving all $ClusterHeadMessages$
    $myClusterHead \leftarrow$ the closest cluster-head
    Send $JoinClusterHeadMessage(ID)$ to the closest cluster-head
    EXIT
end if

Algorithm 3.5: Clustering Algorithm of EAUCF protocol tive cluster-head. The competition radius of each tentative cluster-head changes dynamically in EAUCF, because EAUCF uses residual energy parameter with distance to the base station metric of the sensor node to calculate competition radius. It is logical to decrease the service area of a cluster-head while its residual energy is decreasing. If the competition radius does not change as the residual energy decreases, the sensor node runs out of battery rapidly. EAUCF takes this situation into consideration and decreases the competition radius of each tentative cluster-head as the sensor node battery level decreases. Radius computation is accomplished by using predefined fuzzy if-then mapping rules to handle the uncertainty. These fuzzy if-then mapping rules are given in Table 3.2. We have used Mamdani Method [16] as fuzzy inference technique, because it is the most frequently used fuzzy inference technique [5].

Table 3.2: Fuzzy if-then mapping rules for competiton radius calculation in EAUCF

Rule No	Distance to Base	Residual Energy	Competition Radius
1	Close	Low	Very Small
2	Close	Medium	Small
3	Close	High	Rather Small
4	Medium	Low	Medium Small
5	Medium	Medium	Medium
6	Medium	High	Medium Large
7	Far	Low	Rather Large
8	Far	Medium	Large
9	Far	High	Very Large

In EAUCF cluster-head competition radius calculation, we use two fuzzy input variables. The first one is the distance to the base station of a particular tentative cluster-head. The fuzzy set that describes the distance to base the station input variable is depicted in Figure 3.9. The linguistic variables for this fuzzy set are *close*, *medium* and far. We choose a trapezoidal membership function for *close* and far. On the other hand, the membership function of *medium* is a triangular membership function.

The second fuzzy input variable is residual energy of the tentative cluster-head. The fuzzy set that describes residual energy input variable is illustrated in Figure 3.10. *low*, *medium* and *high* are the linguistic variables of this fuzzy set. *low* and *high* linguistic variables have a trapezoidal membership function while *medium* has a triangular membership function.

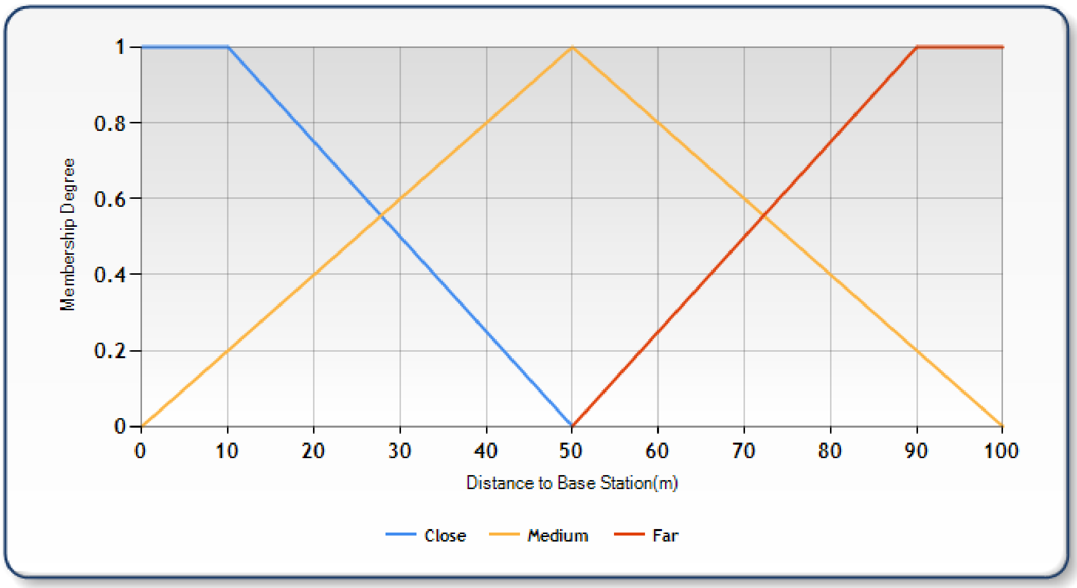

Figure 3.9: Fuzzy set for fuzzy input variable $DistanceToBase$

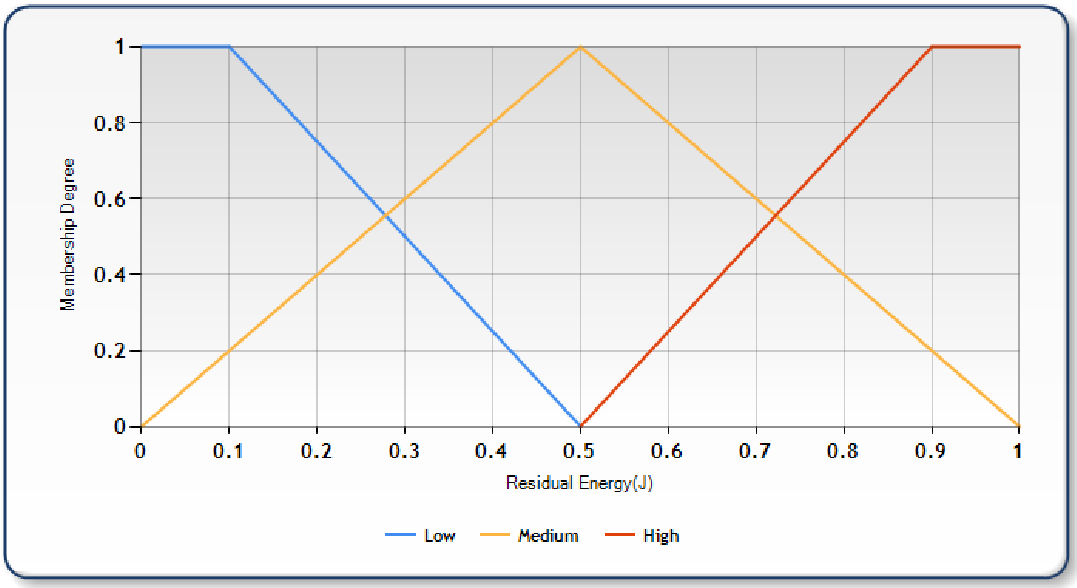

Figure 3.10: Fuzzy set for fuzzy input variable $ResidualEnergy$

The only fuzzy output variable is the competition radius of the tentative cluster-head. Fuzzy set for competition radius fuzzy output variable is demonstrated in Figure 3.11. We have 9 linguistic variables which are *very small*, *small*, *rather small*, *medium small*, *medium*, *medium large*, *rather large*, *large* and *very large*. *very small* and *very large* have a trapezoidal membership function. The remaining linguistic variables are represented by using triangular membership functions.

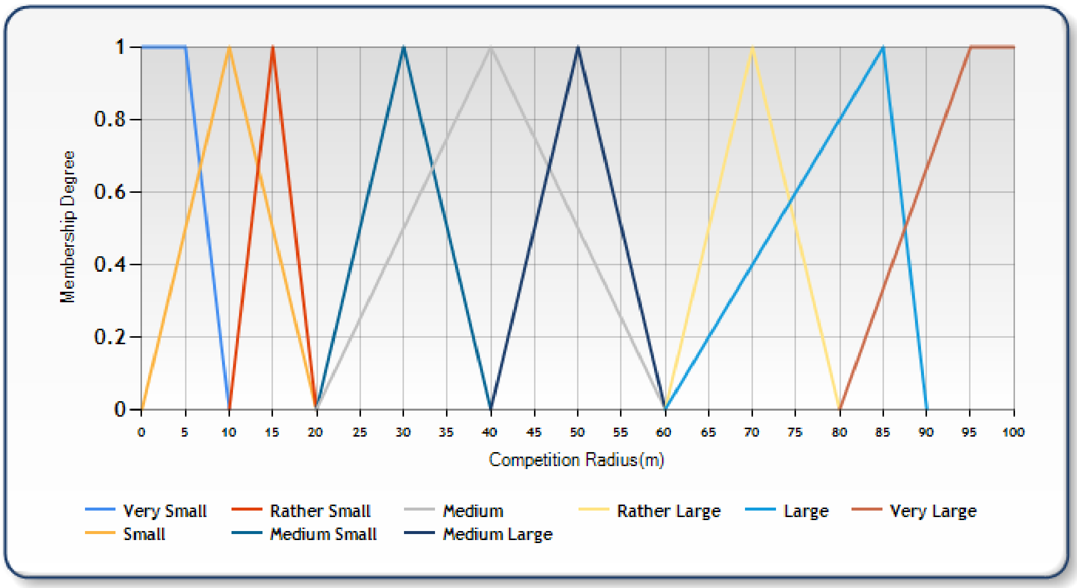

Figure 3.11: Fuzzy set for fuzzy output variable *CompetitionRadius*

If a particular tentative cluster-head's battery is full and it is located at the maximum distance to the base station, then it has the maximum competiton radius. On the contrary, if a particular cluster-head's battery is near empty and is the closest node to the base station, then it has the minimum competition radius. The remaining intermediate possibilities fall between these two extreme cases.

The maximum competition radius is a static parameter for a particular wireless sensor network. The base station broadcasts the value of this parameter to the entire network. Thus, all the sensor nodes know the maximum competition radius, in advance. Each of the sensor nodes can calculate their relative competition radius according to the value of this parameter. The maximum distance to the base station is also a static parameter, because we assume that the sensor nodes are stationary. Each sensor node can determine their relative position to the base station considering the maximum distance to the base station in the WSN.

The change of competition radius according to residual energy and distance to the base station parameters is demonstrated by the examples in Table 3.3. In these examples, the maximum distance to base station is 127 m and the maximum competition radius is set to 60 m. In example 1, 2 and 3, the residual energy levels of the nodes are identical and equal to 1 J. However, their distances to the base station are different. As we get closer to the base station, the competition radius of the sensor node decreases. In example 4 and 5, the distance to the base station is identical, but energy levels are different. The node which has a lower energy has a lower competition radius.

Table 3.3: Examples for fuzzy cluster competition radius calculation

Example No	**Distance to Base(m)**	**Residual Energy(J)**	**Competition Radius(m)**
1	112.90	1.0	45.31
2	20.12	1.0	20.61
3	65.92	1.0	31.61
4	84.31	0.999	38.88
5	84.31	0.70	33.47
6	103.77	0.50	36.65
7	122.80	0.64	46.02
8	112.89	0.59	41.08
9	99.30	0.29	34.63
10	8.94	0.76	12.46

To clarify how we use fuzzy logic to determine cluster-head competition radius for each tentative cluster-head, we give a detailed example here. Suppose we have a tentative cluster-head with residual energy 0.6 J and its distance to the base station is 70 m. The maximum competition radius is set to 50 m and the maximum distance to the base station is 100 m. We evaluate 9 if-then mapping rules which are listed in Table 3.2. To get the membership degree of input variables, we use the membership functions of the linguistic variables. The membership degrees for residual energy and distance to the base station is depicted in Figure 3.12 and 3.13 respectively.

Table 3.4 illustrates the triggered fuzzy if-then rules according to the fuzzified values of residual energy and distance to the base station. Since we use AND in fuzzy if-then mapping rules, we use the minimum operator to get the membership degree of the competition radius for each rule. For example, in rule 8, $DistanceToBase$ is 0.5 Far and $ResidualEnergy$ is 0.8 $Medium$.

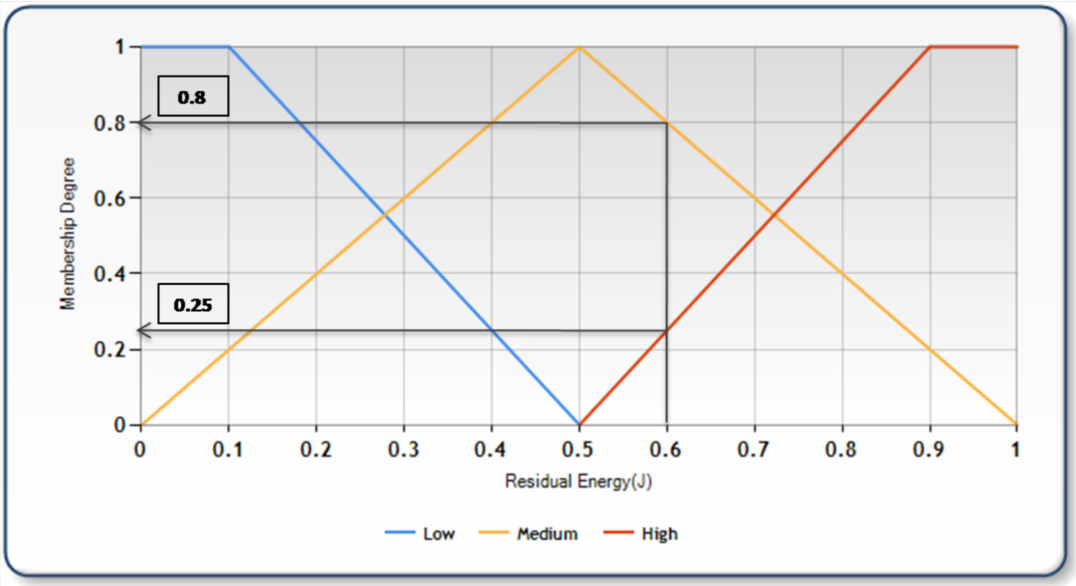

Figure 3.12: Fuzzification of crisp *ResidualEnergy* input variable for value 0.6 J

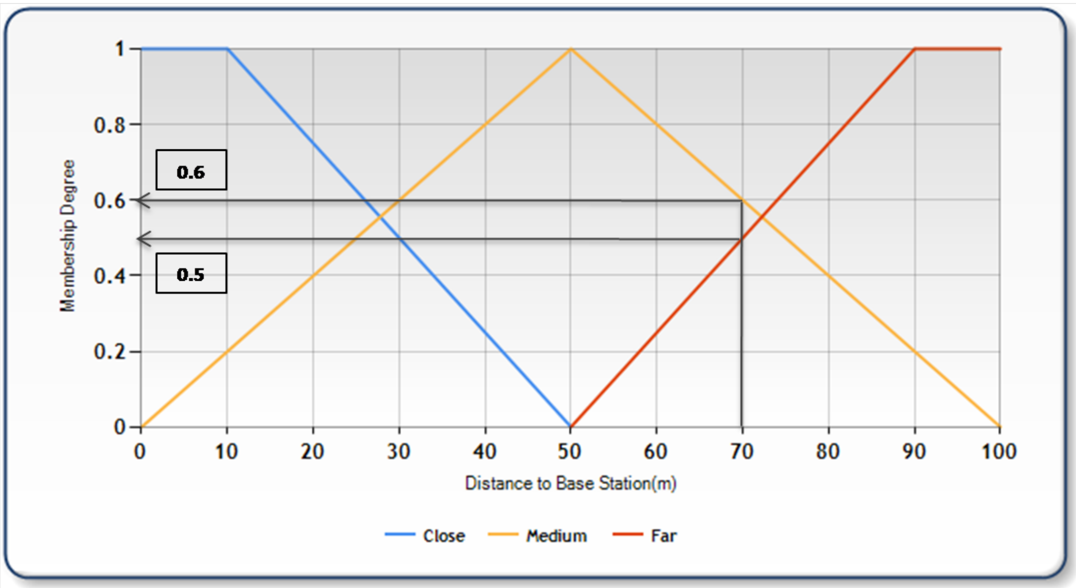

Figure 3.13: Fuzzification of crisp *DistanceToBase* input variable for value 70 m

Therefore, we get $CompetitionRadius$ 0.5 *Large* by applying the minimum operator.

Table 3.4: Evaluation of fuzzy if-then mapping rules

Rule No	DistanceToBase	ResidualEnergy	CompetitionRadius
1	Close=0	Low=0	Very Small (min=0)
2	Close=0	Medium=0.8	Small (min=0)
3	Close=0	High=0.25	Rather Small (min=0)
4	Medium=0.6	Low=0	Medium Small (min=0)
5	Medium=0.6	Medium=0.8	Medium (min=0.6)
6	Medium=0.6	High=0.25	Medium Large (min=0.25)
7	Far=0.5	Low=0	Rather Large (min=0)
8	Far=0.5	Medium=0.8	Large (min=0.5)
9	Far=0.5	High=0.25	Very Large (min=0.25)

After we evaluate the rules, and cut the $CompetitionRadius$ fuzzy set with the calculated input membership degrees, we get the results which are depicted in Figure 3.14. To obtain a crisp competition radius value, we perform defuzzification using COA (Center of Area) method. Equation 3.11 represents the formula that is used to calculate COA. By applying this equation, we get the competition radius approximately 29.24 m.

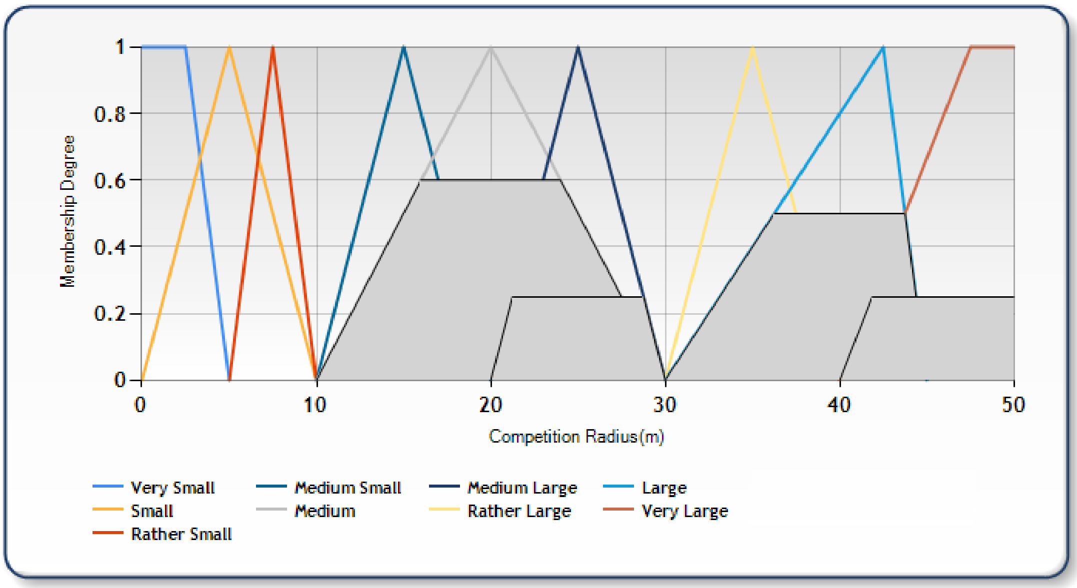

Figure 3.14: Output of evaluation of fuzzy if-then mapping rules

$$DefuzzifiedCompetitionRadius = \frac{\int_x \mu_{CompetitionRadius}(x)xdx}{\int_x \mu_{CompetitionRadius}(x)dx} \quad (3.11)$$

After each tentative cluster-head determines its competition radius, cluster-head competition begins. Each tentative cluster-head advertises *CandidateClusterHeadMessage* to compete with other tentative cluster-heads locally. This message is advertised to the tentative cluster-heads which are inside the maximum cluster-head radius like in EEUC. It includes node ID, competition radius and residual energy level of the source node. Residual energy is the key parameter in cluster-head competition. If a tentative cluster-head receives a *CandidateClusterHeadMessage* from another tentative cluster-head which is in its competition range and the residual energy of the source node is greater than the residual energy of the receiving node, then the receiving node quits cluster-head competition and broadcasts a *QuitElectionMessage*. If a particular tentative cluster-head has the highest residual energy level among the tentative cluster-heads which it recieves a *CandidateClusterHeadMessage* from, then it becomes a cluster-head. This competition guarantees that there does not exist another cluster-head in the competiton radius of a particular cluster-head. After cluster-head election is completed, each ordinary sensor node joins to the closest cluster like in LEACH, CHEF and EEUC.

In order to compare our approach with CHEF, LEACH and EEUC, we have developed an EAUCF simulation. Figure 3.15 and Figure 3.16 illustrate two different cluster-head distribution examples for two different particular rounds.

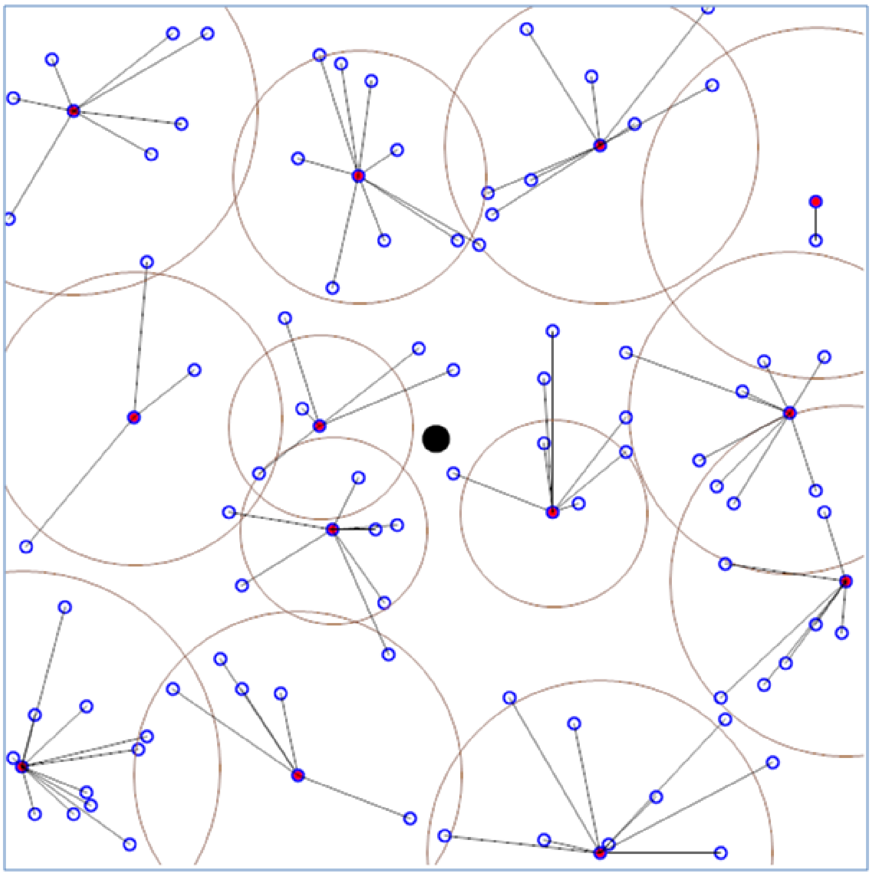

Figure 3.15: Cluster-head distribution example for EAUCF where $R^0_{comp} = 60$, $T = 0.3$ and the number of sensors is 100.

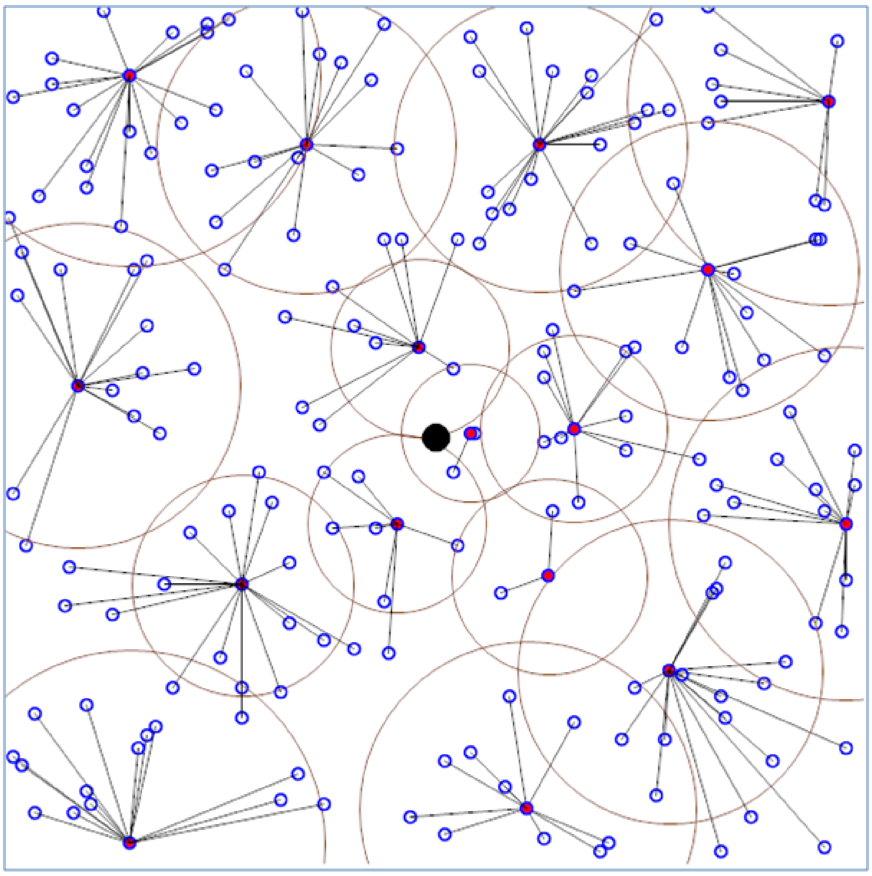

Figure 3.16: Cluster-head distribution example for EAUCF where $R^0_{comp} = 60$, $T = 0.3$ and the number of sensors is 200.

CHAPTER 4

EVALUATION

In this chapter, we present the results of the experiments that we have done to evaluate our algorithm. We compare our clustering algorithm EAUCF with three different algorithms, namely LEACH, CHEF and EEUC. We have implemented a wireless sensor network clustering simulator to evaluate our algorithm. This simulation tool is able to simulate LEACH, CHEF, EEUC and EAUCF for different WSN configurations. We have run several experiments on this tool to evaluate our algorithm. Experimental results have shown that our algorithm performs better than LEACH, CHEF and EEUC in most of the situations tested. Before we get into details of experiments done, we briefly describe the features of our wireless sensor network clustering simulator.

4.1 Wireless Sensor Network Clustering Simulator

It is quite costly to deploy hundreds of sensors to a particular field to conduct experiments for evaluating clustering algorithms. Simulations must be performed repeatedly to generate more reliable results. Therefore, we choose simulation method to evaluate our algorithm EAUCF. We have implemented our own simulation tool to simulate clustering environment, because the existing wireless sensor network simulation tools do not meet all of our expectations. In addition to this, it is a hard task to configure an existing simulation tool for our experiments. Thus, we choose to implement our own simulation tool to make this configuration easier. We have just implemented necessary features that we use in our experiments and exclude the most of the other features that are implemented by other simulation tools but not related to our experiments.

Most of the other wireless sensor network simulators do not provide a graphical user interface (GUI). Our simulation tool provides an interactive interface which enables pausing, accelarating, decelarating, etc. to intervene in clustering simulation process. Several number of clustering simulation scenarios with different configurations can be defined via using this interface. Our simulation tool also provides a detailed results screen that includes detailed charts for visualising the experimental results. It is easy to understand the behaviors of clustering algorithms by using these features. Implementation of this simulation tool is done in C# language on .NET environment.

The major capabilities of our simulation tool is listed below:

- Simulates LEACH, CHEF, EEUC and EAUCF.
- Supports two different radio models for simulating energy dissipation.
- Allows to deploy hundreds of sensors which are located randomly.
- Supports sequential simulation of any number of different types of algorithms.
- Displays the currently running clustering simulation with an interactive user interface.
- Represents the results using FND, HNA and energy-efficiency metrics.
- Displays the results of the performed simulations together for comparing different simulation results.
- Displays the logs of the simulations in the log window.
- Saves the results of the simulations.

4.2 Round Based Evaluation Metrics

The main objective of LEACH, CHEF, EEUC and EAUCF is prolonging the network lifetime by distributing the workload to the sensor nodes evenly. In LEACH, CHEF and EEUC clustering algorithms, the term *round* is used for each of the consecutive periods in which the sensor nodes perform a predefined constant work. For example, in each round, every sensor node forwards 4000 bits of data to its cluster-head. Handy *et al*. used the metrics *First Node*

Dies (FND), *Half of the Nodes Alive* (HNA) and *Last Node Dies* (LND) in [7] for estimating the lifetime of the wireless sensor networks. FND denotes an estimated value for the round in which the first node dies. This metric is useful in sparsely deployed wireless sensor networks. However, in densely deployed wireless sensor networks, death of a single node is not an important issue. Therefore, Handy *et al*. proposed the metric HNA which denotes an estimated value for the round in which the half of the nodes die. In addition to this, they provide another metric LND which denotes an estimated value for the overall lifetime of the network. However, LND is not a very useful metric, because after half of the sensor nodes die, the wireless sensor network becomes almost useless in most of the cases. Therefore, we mostly pay attention to the metrics FND and HNA in order to evaluate our simulation results.

4.3 Scenarios

In order to evaluate our proposed algorithm EAUCF, we have compared EAUCF with LEACH, CHEF and EEUC. In each of the scenarios, we run all of the algorithms on an identical wireless sensor network. We have used the same random seed to generate the identical wireless sensor netwowk topology. First order radio model is used as energy dissipation model. This model simulates the energy consumption of each sensor node for transmitting and receving l bits of data.

In each round of the scenario, clusters-heads are elected and clusters are formed. Afterwards, each ordinary node forwards a certain bits of data to its cluster-head. Each cluster-head aggregates the received data and forwards it to the base station with a particular routing protocol or directly transmits the aggregated data to the base station. LEACH cluster-heads transmit their data packets to the base station directly. In our simulations, CHEF, EEUC and EAUCF cluster-heads can forward their data packets to the base station directly or can use EEUC multi-hop routing protocol.

The area of deployed wireless sensor network is same for all scenarios and is 200x200 m. In each round, each ordinary sensor node transmits 4000 bits of data to its cluster-head. The cluster-head which receives the data from its cluster members, aggregates the received data with a certain aggregation ratio. This aggregation ratio is set to 10% in our simulations. In the simulations of CHEF, Kim *et al*. also used the same aggregation ratio. The length of the aggregated data for a particular cluster-head is calculated using Equation 4.1.

$$L_{agg} = L_{rec} + (L_{rec} * R_{agg} * N) \tag{4.1}$$

In equation 4.1, L_{agg} represents the length of the aggregated data in bits while L_{rec} represents the length of the received data from each cluster member. R_{agg} is the ratio of aggregation and N is the total number of cluster members. For example, if a particular cluster has 20 cluster members each of transmitting 100 bits of data to their cluster-head where the aggregation ratio is set to 10%, then the length of the aggregated data is (100 + (100 * 0.1 * 20)) which is equal to 300 bits.

In all of the scenarios, the desired percentage of cluster-heads for LEACH is set to 0.1. The α value of CHEF algorithm is set to 2.5. The optimal threshold P_{opt} for CHEF is calculated approximately 0.3 for 100 nodes and 0.21 for 200 nodes using Equation 3.4 and Equation 3.5. The threshold T is set to 0.4 and the coefficient c is set to 0.5 for EEUC clustering algorithm. The value of the threshold T is 0.3 for EAUCF.

In order to produce more reliable results, every scenario is simulated for 50 times, and the average of the results are taken. For each of the scenarios, we provide a summary result table which represents the values of FND and HNA metrics for each of the algorithms simulated. After that, we provide a summary chart which illustrates the values of FND and HNA metrics visually. We also generate charts for the distribution of the number of alive sensor nodes and the distribution of the number of clusters per each round. By using these simulation results, we comment on the performance of the simulated algorithms.

4.3.1 Scenario 1

In this scenario, the base station is located at the center of the wireless sensor network. Each cluster-head forwards the aggregated data to the base station directly without using a relay node. The detailed configuration of this scenario is depicted in Table 4.1.

The maximum competition radius is assigned as 30 and 60 m for EEUC and EAUCF, respectively. These are the optimal maximum competition radius values for this scenario. After wireless sensor network is deployed, the maximum distance to the base station is calculated

Table 4.1: Configuration parameters of Scenario 1

Parameter	Value
Network size	200x200m
Base station location	(100,100)m
Number of sensor nodes	100
Initial energy	1 J
Data packet size	4000 bits
ϵ_{amp}	100 pJ/bit/m^2
E_{elec}	50 nJ/bit
Aggregation Ratio	10%

approximately 127.35 m.

The simulation of this scenario yielded the following results. Table 4.2 shows the rounds in which the first node died (FND) and half of the nodes alive (HNA) for each simulated algorithm.

Table 4.2: Scenario 1: Values of FND and HNA metrics for each algorithm

Algorithm	FND	HNA
LEACH	360	628
CHEF	438	773
EEUC	425	772
EAUCF	491	821

As seen in Table 4.2, our proposed algorithm EAUCF performs better than LEACH, CHEF and EEUC for both FND and HNA metrics. The performance of CHEF and EEUC are close to each other, but CHEF performs slightly better than EEUC especially for FND. LEACH has the poorest performance among the four clustering algorithms for this scenario. EAUCF is more efficient than LEACH about 36.4%, CHEF about 12.1% and EEUC about 15.5% if we consider FND metric. It performs better than LEACH about 30.7%, CHEF about 6.2% and EEUC about 6.3% if HNA metric is used for performance evaluation.

LEACH performance is the poorest one, because it does not consider the residual energy level of the sensor nodes during clustering. It uses a pure probabilistic model for clustering, but this model itself is not sufficient for providing the best solution. Since CHEF takes both energy and local distance paremeters into consideration, it performs better than LEACH. EEUC also considers energy and distance to the base station parameters. Hence, it has a better performance than LEACH. EAUCF considers the energy level of each tentative cluster-head in competition radius calculation. This means if a tentative cluster-head has more energy, then it will have a greater cluster radius. In other words, it can serve to more sensor nodes in the local region. This property ensures that EAUCF assigns more work to the cluster-heads which have more energy. This consideration makes EAUCF perform better than other algorithms for this scenario. The summary chart in Figure 4.1 illustrates the comparison of algorithms according to FND and HNA metrics visually.

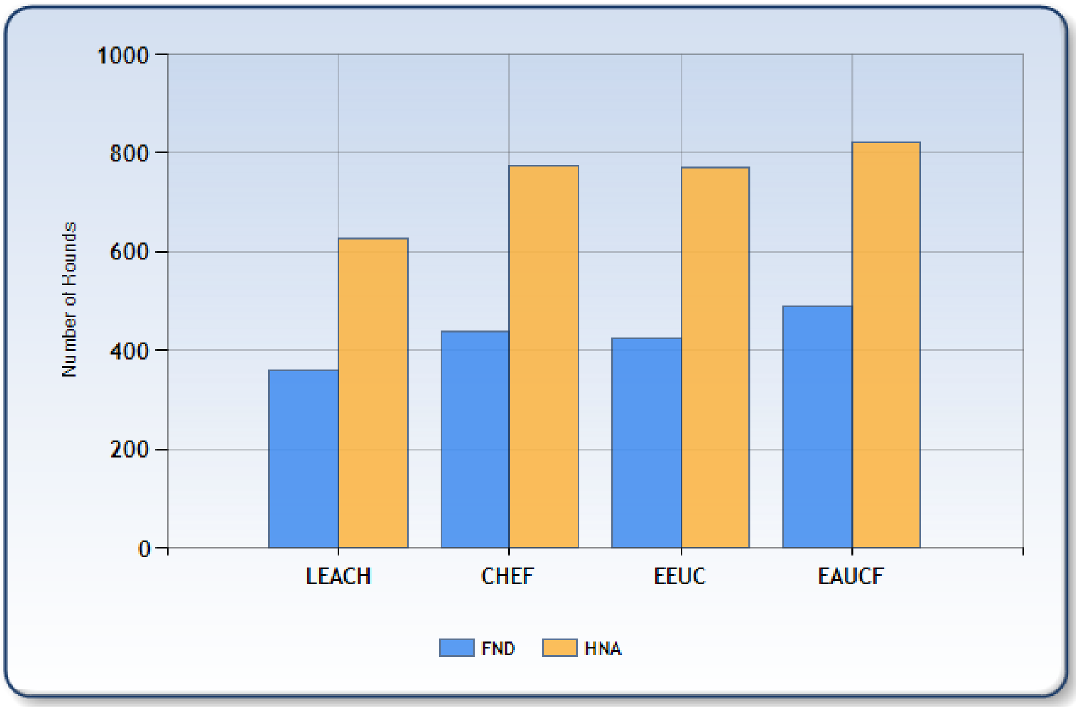

Figure 4.1: Scenario 1: Values of FND and HNA metrics for each algorithm

Figure 4.2 depicts the distribution of the number of alive sensor nodes with respect to the number of rounds for each algorithm. This figure clearly depicts that deaths of sensor nodes for EAUCF begin after all the other algorithms.

Figure 4.3 shows the distribution of the number of clusters with respect to the number of rounds for each algorithm. As seen in this figure LEACH, CHEF and EEUC produce approximately constant number of clusters for each round until the first node dies. However, EAUCF slightly increases the number of clusters up to the first node dies. This is because that the cluster radius is directly proportional to energy level of each tentative cluster-head. If energy level decreases, the cluster radius gets smaller. Therefore, the number of clusters is

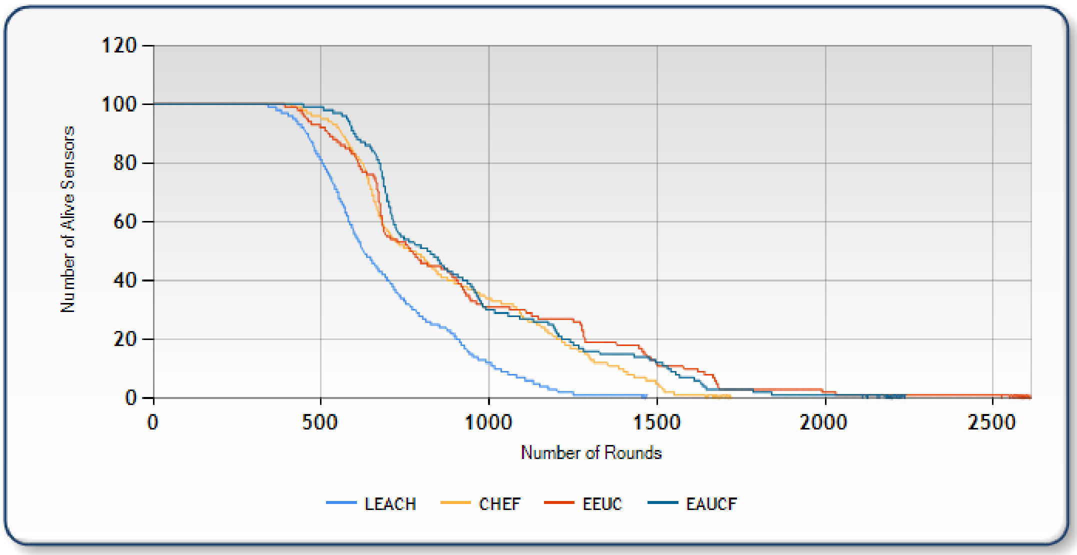

Figure 4.2: Scenario 1: Distribution of alive sensor nodes according to the number of rounds for each algorithm

increased to cover all of the wireless sensor network.

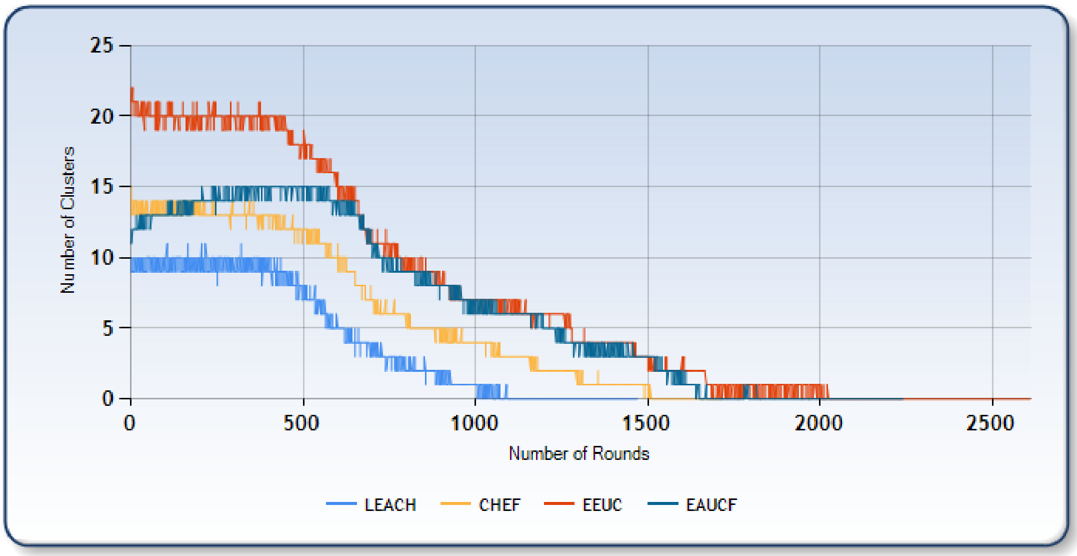

Figure 4.3: Scenario 1: Distribution of number of clusters according to the number of rounds for each algorithm

Table 4.3 represents the total remaining energy for each algorithm at round 500. By using the information in this table, we compare the energy efficiencies of the simulated algorithms. Since every node has 1 J initial energy, the total energy of WSN is 100 J at the beginning. The battery of each sensor node depletes as the number of round increases. At round 500, LEACH has the lowest energy level which is approximately 24 J. The energy levels of EEUC

Table 4.3: Scenario 1: Total remaining energy for each algorithm at round 500

Algorithm	Total Remamining Energy (J)
LEACH	24.47
CHEF	37.49
EEUC	37.76
EAUCF	40.36

and CHEF are nearly identical and approximately equal to 38 J. On the other hand, EAUCF has the highest energy level that is approximately 40 J. This result is parallel to the results which are inferred from FND and HNA metrics.

4.3.2 Scenario 2

In this scenario, the base station is located at the center of the wireless sensor network just like in Scenario 1. However, CHEF, EEUC and EAUCF cluster-heads use EEUC multi-hop routing protocol to forward their data packets rather than directly transmitting them to the base station. By comparing the results of Scenario 1 and Scenario 2, we see the impact of using a multi-hop routing protocol instead of direct routing. The detailed configuration of this scenario is illustrated in Table 4.4.

Table 4.4: Configuration parameters of Scenario 2

Parameter	Value
Network size	200x200m
Base station location	(100,100)m
Number of sensor nodes	100
Initial energy	1 J
Data packet size	4000 bits
ϵ_{amp}	100 pJ/bit/m^2
E_{elec}	50 nJ/bit
Aggregation Ratio	10%

The maximum competition radius is set to 40 and 70 m for EEUC and EAUCF respectively. These are the optimal maximum competition radius values for this scenario. After wireless sensor network is deployed, the maximum distance to base station is calculated approximately 129.42 m.

The simulations of this scenario produced the following results. Table 4.5 shows the rounds in which the first node died (FND) and half of the nodes alive (HNA) for each simulated algorithm.

Table 4.5: Scenario 2: Values of FND and HNA metrics for each algorithm

Algorithm	FND	HNA
LEACH	390	695
CHEF	607	767
EEUC	723	777
EAUCF	761	831

As shown in Table 4.5, EAUCF outperforms LEACH, CHEF and EEUC considering FND and HNA metrics. LEACH has the lowest performance like in Scenario 1. In the first scenario, FND values of CHEF and EEUC are close to each other. However, in this scenario EEUC is better than CHEF about 23.2% considering FND metric. Their HNA performance is still close to each other. EAUCF is more efficient than LEACH about 95.1%, CHEF about 25.4% and EEUC about 5.3% according to FND metric. If we consider HNA metric for evaluation, the performance of EAUCF is better than LEACH about 19.6%, CHEF about 8.3% and EEUC about 6.9%.

In this scenario, the performance of LEACH is the lowest one again, because the same reasons in the first scenario also apply to this scenario. The results of this scenario clearly indicate that unequal clustering algorithms, which are EEUC and EAUCF, perform better than LEACH and CHEF when the multi-hop routing protocol is used. This is because that the batteries of the sensor nodes that are closer to the base station deplete faster. However, EEUC and EAUCF handle this situtation by assigning smaller cluster sizes to the sensor nodes which are closer to the base station. On the other hand, CHEF cannot perform as well as EEUC and EAUCF, because it does not consider the hot spots problem. However, when CHEF cluster-heads use EEUC routing protocol instead of forwarding directly to the base station, it performs slightly better. Since EAUCF considers the energy level of the tentative cluster-heads during cluster radius calculation, the performance of EAUCF is quite better than EEUC. The summary chart in Figure 4.4 represents the comparison of the algorithms graphically considering FND and HNA metrics.

Figure 4.5 illustrates the distribution of the alive sensor nodes with respect to the number of

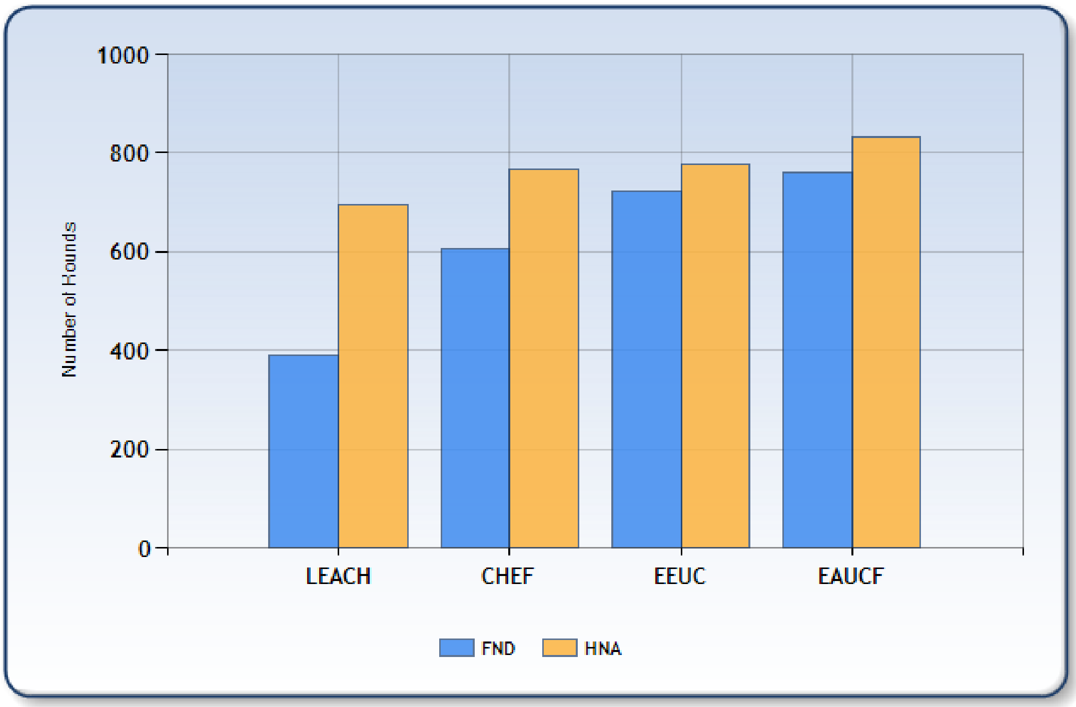

Figure 4.4: Scenario 2: Values of FND and HNA metrics for each algorithm

rounds for each algorithm. This figure clearly shows that our proposed algorithm is more stable than the other algorithms, because sensor node deaths begin later in EAUCF and continue linearly until all sensor nodes die.

Figure 4.6 shows the distribution of the number of clusters with respect to the number of rounds for each algorithm. LEACH, CHEF and EEUC generate constant number of clusters until the first node dies while the number of clusters generated by EAUCF increases. This situtation is also observed in the first scenario. The benefits of this situation is mentioned in Scenario 1.

Table 4.6 shows the total remaining energy for each algorithm at round 500. EAUCF seems to be the most energy-efficient algorithm in this scenario, because it has the highest remaining energy level which is approximately 41 J. The remaining energy levels of EEUC and CHEF are close to EAUCF. On the other hand, LEACH has the lowest remaining energy level which is approximately 28 J. These results are parallel to the results which are inferred from FND and HNA metrics.

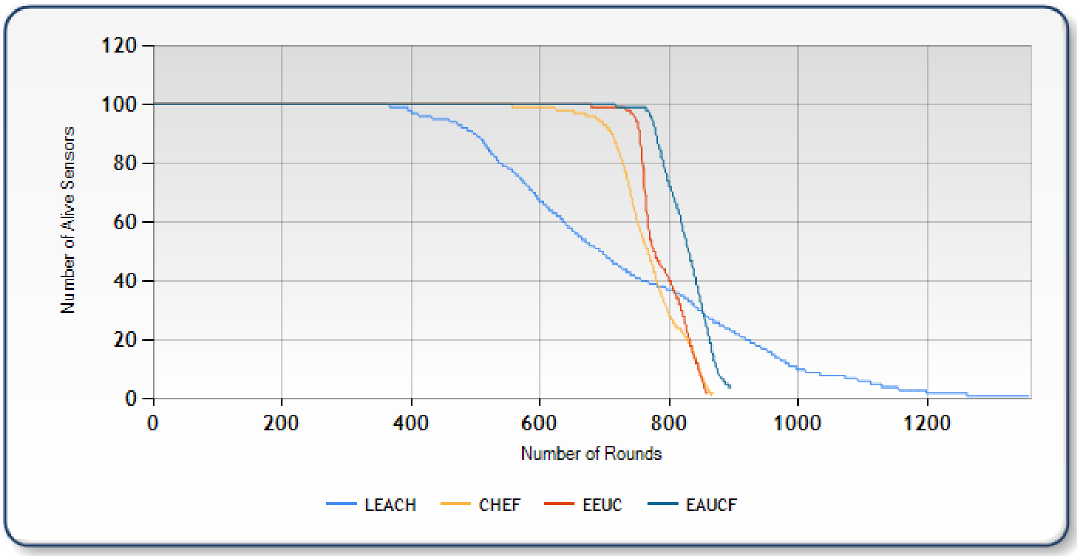

Figure 4.5: Scenario 2: Distribution of alive sensor nodes according to the number of rounds for each algorithm

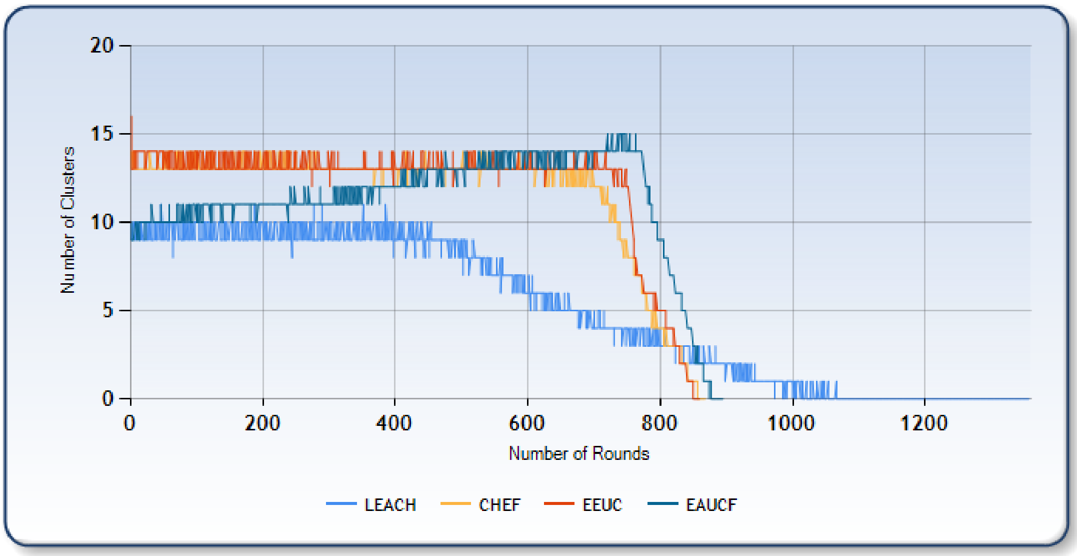

Figure 4.6: Scenario 2: Distribution of the number of clusters according to the number of rounds for each algorithm

Table 4.6: Scenario 2: Total remaining energy for each algorithm at round 500

Algorithm	Total Remamining Energy (J)
LEACH	28.40
CHEF	38.34
EEUC	39.89
EAUCF	41.23

4.3.3 Scenario 3

In this scenario, the base station is located at the center of the wireless sensor network like in Scenario 1 and 2. CHEF, EEUC and EAUCF cluster-heads use EEUC routing protocol for data transmission. In this scenario, the density of the deployed sensor nodes is doubled with respect to Scenario 2. We aim to test the behaviors of the clustering algorithms in different sensor network topologies which have different number of deployed sensor nodes. In other words, we try to find out how clustering algorithms perform in relatively dense and sparse sensor network deployments. The detailed configuration of this scenario is illustrated in Table 4.7.

Table 4.7: Configuration parameters of Scenario 3

Parameter	Value
Network size	200x200m
Base station location	(100,100)m
Number of sensor nodes	200
Initial energy	1 J
Data packet size	4000 bits
ϵ_{amp}	100 pJ/bit/m^2
E_{elec}	50 nJ/bit
Aggregation Ratio	10%

The maximum competition radius is set to 35 and 70 m for EEUC and EAUCF, respectively. These are the optimal maximum competition radius values for this scenario. After wireless sensor network is deployed, the maximum distance to the base station is calculated approximately 137.93 m.

The simulations of this scenario provided the following results. The values for FND and HNA metrics for each algorithm are shown in Table 4.8.

Table 4.8: Scenario 3: Values of FND and HNA metrics for each algorithm

Algorithm	FND	HNA
LEACH	406	875
CHEF	620	848
EEUC	753	840
EAUCF	750	942

As seen in Table 4.8, the simulation results of this scenario comform more or less with the simulation results of Scenario 2. However, the HNA performance of LEACH is increased significantly in this scenario with respect to Scenario 2. EEUC and EAUCF have the highest FND performance among four clustering algorithms. LEACH sensor nodes start to die earlier than the sensor nodes of the other algorithms. EAUCF is more efficient than LEACH about 84.7% and CHEF about 21.0% considering FND metric. The HNA performance of EAUCF is higher than LEACH about 7.7%, CHEF about 11.1% and EEUC about 12.1%.

In this scenario, FND performance of LEACH is significantly lower than the other algorithms. FND performance of LEACH in Scenario 1 and Scenario 2 are close to the performance in this scenario. The reasons of this low performance which are provided in the former scenarios are also valid for this scenario. Unequal clustering algorithms EEUC and EAUCF outperform LEACH and CHEF considering FND metric, because they handle the hot spots problem when multi-hop routing protocol is used for data transmission from cluster-heads to the base station. If we consider the HNA metrics, EAUCF performs slightly better than CHEF and EEUC in densely deployed sensor networks. In addition to this, LEACH's HNA performance is remarkable in this scenario, but still lower than the performance of EAUCF. Figure 4.7 demonstrates the comparison of the values of FND and HNA metrics for each simulated algorithm in a visual manner.

Figure 4.8 shows the distribution of the alive sensor nodes according to the number of rounds for each simulated algorithm. As seen in this figure, the number of sensor nodes of EAUCF algorithm is significantly greater than the other algorithms when the number of alive sensor nodes is 100. This situation implies that EAUCF keeps the wireless sensor network stable for a longer time than the other algorithms.

As depicted in Figure 4.9, EAUCF generates lower number of clusters in the earlier rounds.

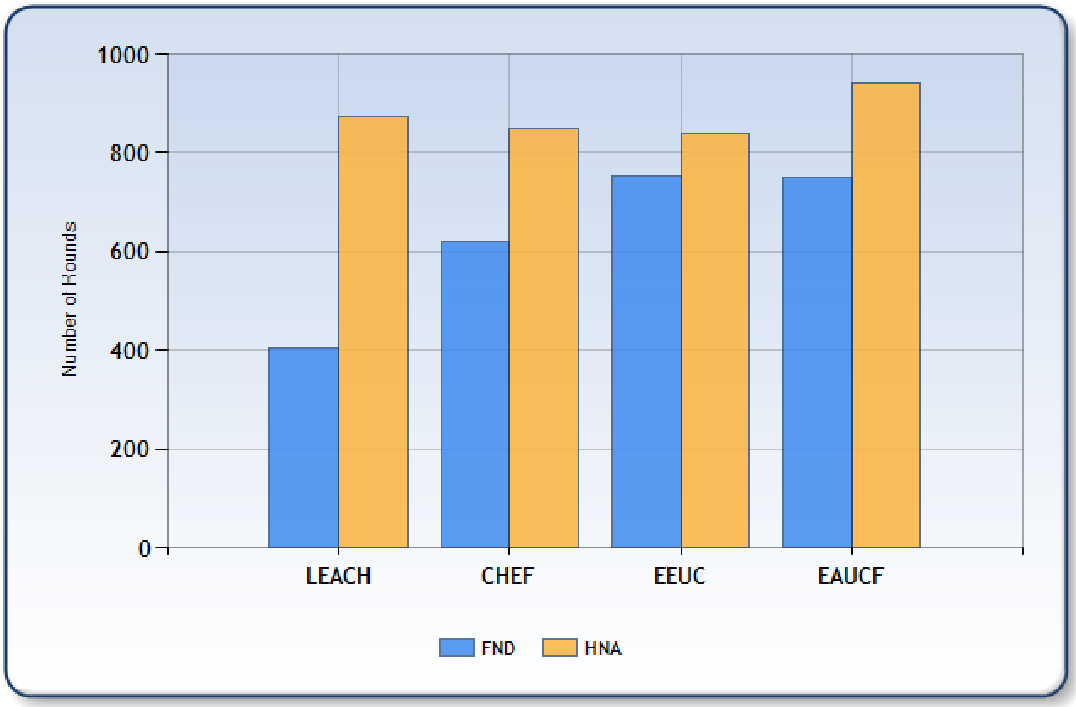

Figure 4.7: Scenario 3: Values of FND and HNA metrics for each algorithm

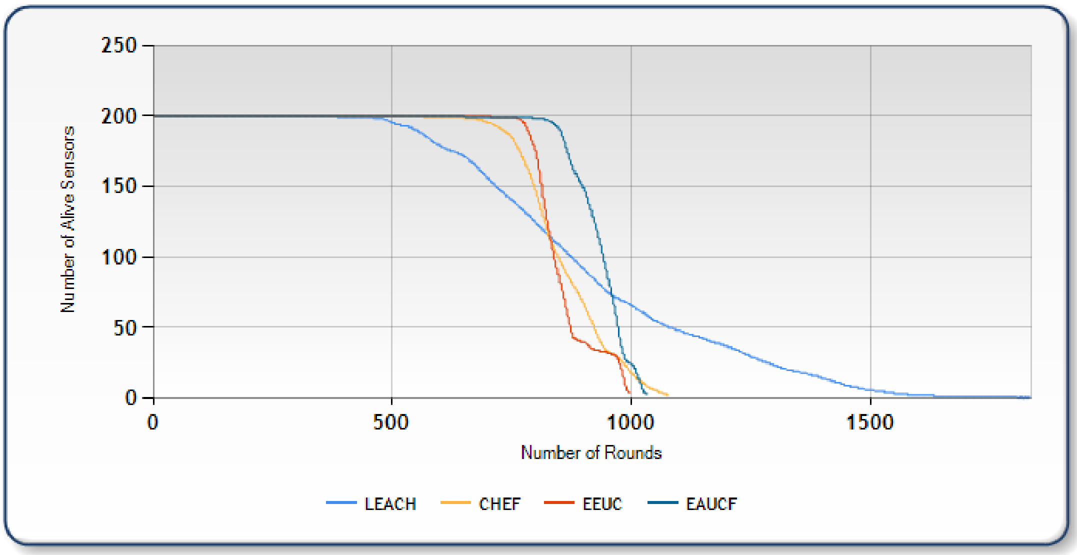

Figure 4.8: Scenario 3: Distribution of alive sensor nodes according to the number of rounds for each algorithm

However, the number of clusters increases in a linear manner until the node deaths begin. This is beacuse that the radius of cluster-heads are reduced as the residual energy decreases. This situation triggers the increase in the number of cluster-heads.

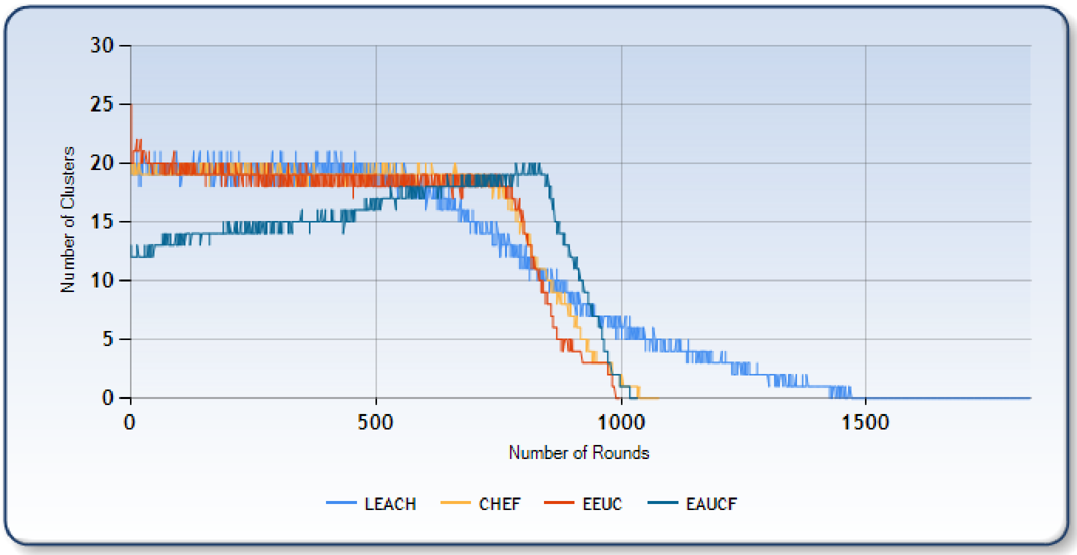

Figure 4.9: Scenario 3: Distribution of the number of clusters according to the number of rounds for each algorithm

Table 4.9 shows the total remaining energy levels for each algorithm at round 500. EAUCF has the highest energy level, which is approximately 96 J, among all of the simulated algorithms. In Scenario 1 and 2, the remaining energy level of CHEF has been nearly same with EEUC. However, in this scenario CHEF has a higher remaining energy level which is approximately 92 J. The sensor nodes of LEACH consumed much more energy up to the round 500 than the other algorithms for this scenario.

Table 4.9: Scenario 3: Total remaining energy for each algorithm at round 500

Algorithm	Total Remamining Energy (J)
LEACH	82.84
CHEF	92.40
EEUC	90.45
EAUCF	96.33

4.3.4 Scenario 4

In this scenario, the base station is located at (100,250) m which is outside of the wireless sensor network. This is different from Scenario 1, 2 and 3 in which the base stations are located at the center. Each cluster-head sends the aggregated data packet to the base station by using EEUC routing protocol except LEACH. By comparing the results of Scenario 2 and 4, we will see how the location of the base station affects the results of the simulations. The detailed configuration of this scenario is illustrated in Table 4.10.

Table 4.10: Configuration parameters of Scenario 4

Parameter	Value
Network size	200x200m
Base station location	(100,250)m
Number of sensor nodes	200
Initial energy	1 J
Data packet size	4000 bits
ϵ_{amp}	100 pJ/bit/m^2
E_{elec}	50 nJ/bit
Aggregation Ratio	10%

The maximum competition radius is set to 60 and 110 m for EEUC and EAUCF, respectively. These are the optimal maximum competition radius values for this scenario. After wireless sensor network is deployed, the maximum distance to base station is calculated approximately 260.28 m.

The simulation of this scenario yielded the following results. Table 4.11 shows the rounds in which the first node died (FND) and half of the nodes alive (HNA) for each simulated algorithm.

Table 4.11: Scenario 4: Values of FND and HNA metrics for each algorithm

Algorithm	FND	HNA
LEACH	173	336
CHEF	159	419
EEUC	346	424
EAUCF	396	445

As seen in Table 4.11, the values of FND and HNA metrics for each algorithm have decreased with respect to former scenarios. This is because that the base station is located outside of the wireless sensor network. Thus, the cluster-heads consume much more energy to trasnmit their data packets to the base station. In this scenario, EAUCF has outperformed LEACH, CHEF and EEUC considering both FND and HNA metrics. CHEF has the lowest FND performance while LEACH has the lowest HNA performance. If we consider FND metric, EAUCF is more efficient than LEACH about 128.9%, CHEF about 149.1% and EEUC about 14.4%. On the other hand, if HNA metric is considered, the performance of EAUCF is greater than LEACH about 32.4%, CHEF about 6.2% and EEUC about 5.0%.

In this scenario, unequal clustering algorithms outperform LEACH and CHEF considering FND metric. This implies that if smaller cluster-head radius values are assigned to the cluster-heads closer to the base station, the beginning of sensor node deaths can be delayed. This is the key observation in all of the scenarios. As we have also observed in the former scenarios, the radius calculation approach of EAUCF makes it perform better than EEUC. The results of this simulation show that unequal clustering approaches perform better even if the base station is located outside of the wireless sensor network. CHEF shows a remarkable HNA performance in this scenario, but its FND performance is the lowest one. CHEF is a clustering algorithm which assigns a static cluster-head radius to all its cluster-heads. Therefore, it cannot handle the hot spots problem. Consequently, the sensor nodes start to die earlier than EEUC and EAUCF which are unequal clustering algorithms. Figure 4.10 illustrates the FND and HNA values of the simulated algorithms on a column chart.

Figure 4.11 shows the distribution of the alive sensor nodes with respect to the number of rounds for each simulated algorithm. As seen in this figure, the sensor nodes of LEACH and CHEF start to die in the earlier rounds. The sensor node deaths for EAUCF starts later than all the other algorithms. EAUCF provides at least 400 stable rounds for this particular wireless sensor network.

The distribution of the number of clusters with respect to the number of rounds for each algorithm is depicted on a fast line chart in Figure 4.12. CHEF generates the highest number of clusters at the earlier rounds. On the other hand, EAUCF generates the lowest number of cluster-heads in the beginning. As the number of rounds increases, EAUCF starts to generate more cluster-heads until first node dies. This approach helps EAUCF to delay the sensor node

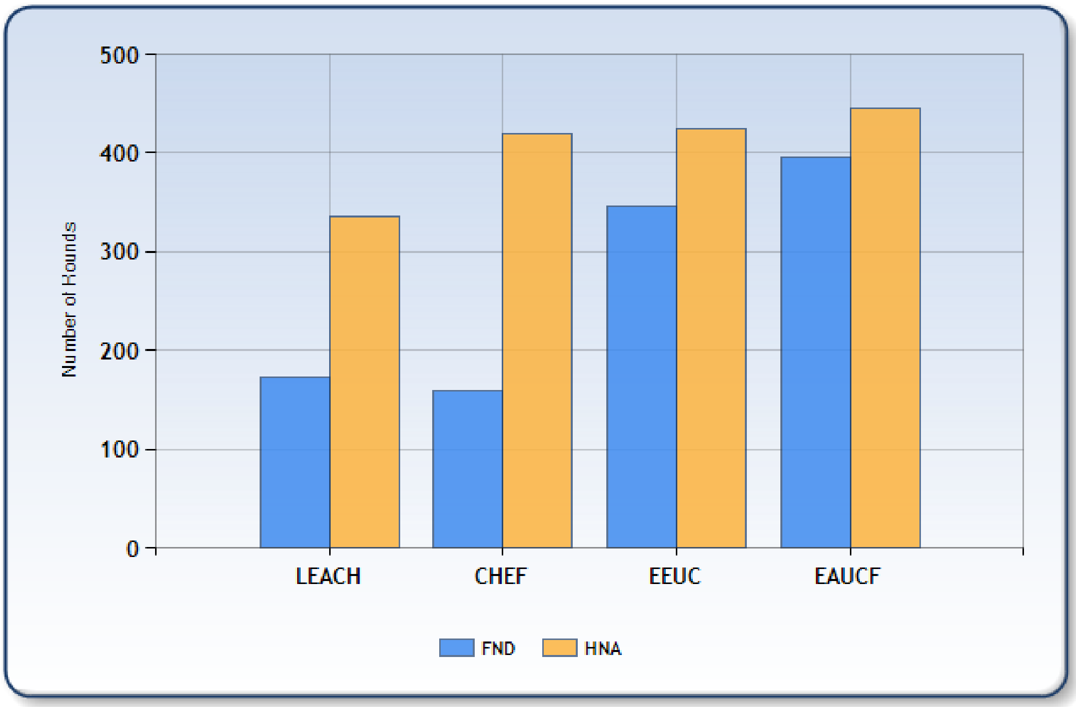

Figure 4.10: Scenario 4: Values of FND and HNA metrics for each algorithm

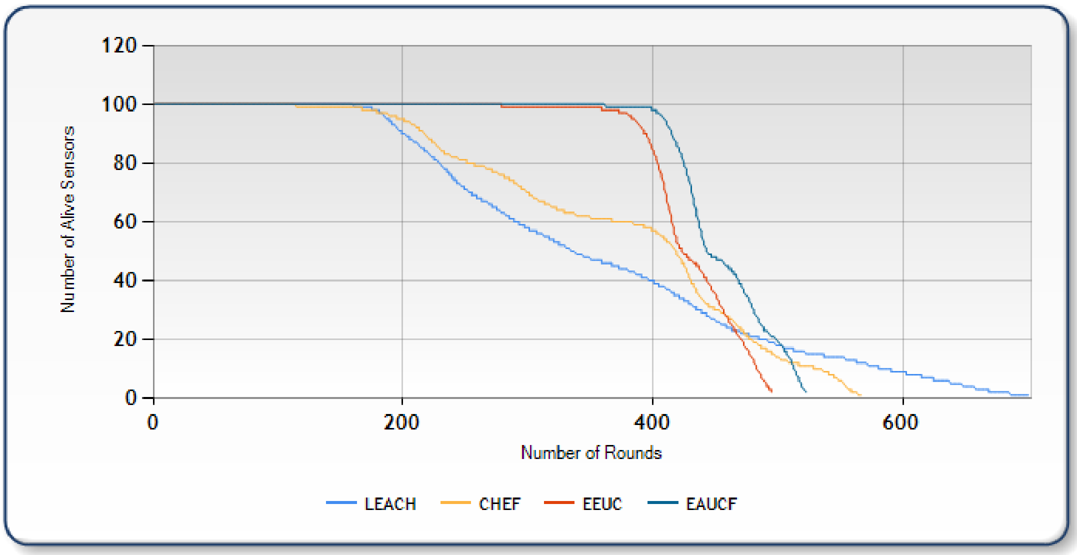

Figure 4.11: Scenario 4: Distribution of alive sensor nodes according to the number of rounds for each algorithm

deaths up to round 400.

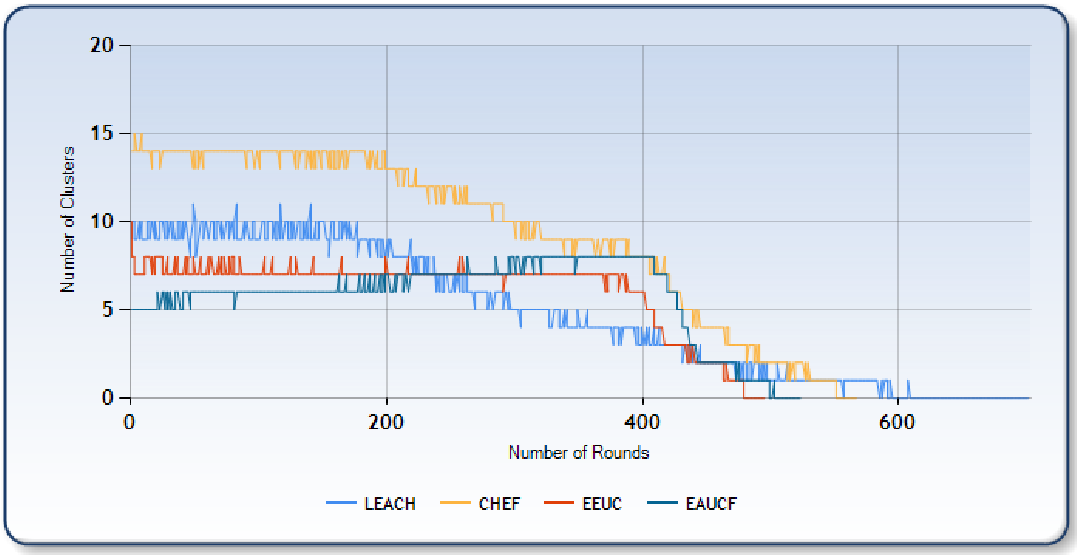

Figure 4.12: Scenario 4: Distribution of the number of clusters according to the number of rounds for each algorithm

Table 4.12: Scenario 4: Total remaining energy for each algorithm at round 250

Algorithm	**Total Remaming Energy (J)**
LEACH	27.14
CHEF	43.66
EEUC	44.78
EAUCF	46.41

Total remaining energy levels at round 250 for each simulated algorithm are given in Table 4.12. EAUCF has the highest energy level which is approximately 46 J. This data represents that EAUCF is the most energy-efficient algorithm for this scenario. The remaining energy levels of CHEF and EEUC are close to each other. As we have also observed in former scenarios, LEACH consumes considerably more energy than other algorithms.

CHAPTER 5

CONCLUSION

In this thesis we have proposed a fuzzy unequal clustering algorithm for wireless sensor networks, namely EAUCF. The main objective of our algorithm is to prolong the lifetime of the wireless sensor network by evenly distributing the workload. To achieve this goal, we have mostly focused on assigning proper cluster-head competition ranges to sensor nodes.

EAUCF adjusts the cluster-head radius values considering energy and distance to the base station parameters of the sensor nodes. We blend these parameters by using fuzzy logic to obtain an appropriate cluster-head radius. If a particular sensor node has a higher residual energy and is located far from the base station, then it has a greater cluster-head radius. On the other hand, if a particular sensor node has a lower residual energy and is close to the base station, then it has a smaller radius. The network traffic increases as we approach to the base station in multi-hop wireless sensor networks. Therefore, the sensor nodes close to the base station die earlier. Our radius adjustment mechanism solves this hot spots problem by reducing the intra-cluster work of the cluster-heads closer to the base station.

After describing our algorithm EAUCF to solve hot spots problem, we have introduced our wireless sensor network clustering simulator tool in order to evaluate our algorithm. We implement LEACH, CHEF, EEUC and our algorithm EAUCF in this clustering simulator. Evaluation is done by comparing the simulation results of four different clustering scenarios. In all of the scenarios, we evaluate the performance of each of the clustering algorithms using the FND and HNA metrics. In addition to this, we evaluate the energy efficieny of the algorithms by comparing the remaining energy levels at a certain clustering round.

We have shown that our proposed algorithm has a better performance compared to LEACH, CHEF and EEUC considering the simulation results. In all of four different scenarios, EAUCF sensor nodes start to die later than other algorithms except Scenario 3. In that scenario, EEUC and EAUCF sensor nodes start to die nearly in the same round. Furthermore, EAUCF has outperformed all of the algorithms considering the HNA metric. In all of the scenarios the total remaining energy level of EAUCF at a certain round is higher than the other algorithms. Therefore, EAUCF is more energy-efficient than other simulated clustering algorithms.

As a result of these experiments, we conclude that fuzzy unequal clustering algorithm EAUCF is a stable and energy-efficient clustering algorithm for wireless sensor networks.

EAUCF algorithm is designed for the wireless sensor networks that have stationary sensor nodes. As a future work, the fuzzy unequal clustering approach of our algorithm can be extended for handling mobile sensor nodes.

Residual energy, distance to the base station and competition radius fuzzy sets can be adjusted in order to find optimal cluster-head radius values. In addition to this, the optimal maximum competition radius values for each scenario can be estimated by applying extensive tests.

In cluster-head competition, we only consider the residual energy of the tentative cluster-heads. Some additional parameters such as node degree, density and local distance may also be employed to improve the performance of EAUCF.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Comput. Netw.*, 38(4):393–422, 2002.

[2] E.M. Belding-Royer. Hierarchical routing in ad hoc mobile networks. *Wireless Communications and Mobile Computing*, 2(5):515–532, 2002.

[3] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, page 270. ACM, 1999.

[4] I. Gupta. Cluster head election using fuzzy logic for wireless sensor networks. Master's thesis, Dalhousie University, Halifax, Nova Scotia, March 2005.

[5] I. Gupta, D. Riordan, and S. Sampalli. Cluster-head election using fuzzy logic for wireless sensor networks. In *Communication Networks and Services Research Conference, 2005. Proceedings of the 3rd Annual*, pages 255–260, 2005.

[6] Y. Han, S. Park, J. Eom, and T. Chung. Energy-Efficient Distance Based Clustering Routing Scheme for Wireless Sensor Networks. *LECTURE NOTES IN COMPUTER SCIENCE*, 4706:195, 2007.

[7] MJ Handy, M. Haase, and D. Timmermann. Low energy adaptive clustering hierarchy with deterministic cluster-head selection. In *IEEE MWCN*. Citeseer, 2002.

[8] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1(4):660–670, 2002.

[9] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Hawaii International Conference on System Sciences*, volume 8, page 8020. Citeseer, 2000.

[10] J. Ibriq and I. Mahgoub. Cluster-based routing in wireless sensor networks: issues and challenges. In *Proceedings of the 2004 Symposium on Performance Evaluation of Computer Telecommunication Systems (SPECTS)*, 2004.

[11] J.M. Kim, S.H. Park, Y.J. Han, and T.M. Chung. CHEF: cluster head election mechanism using fuzzy logic in wireless sensor networks. In *Proceedings of the ICACT*, pages 654–659, 2008.

[12] F. Kuhn, T. Moscibroda, and R. Wattenhofer. Initializing newly deployed ad hoc and sensor networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 260–274. ACM New York, NY, USA, 2004.

[13] C. Li, M. Ye, G. Chen, and J. Wu. An energy-efficient unequal clustering mechanism for wireless sensor networks. In *Proceedings of the 2nd IEEE International Conference on Mobile Ad-Hoc and Sensor Systems*, page 8. Citeseer.

[14] M. Lotfinezhad and B. Liang. Effect of partially correlated data on clustering in wireless sensor networks. In *Proc. of the IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON)*. Citeseer, 2004.

[15] V. Mhatre and C. Rosenberg. Design guidelines for wireless sensor networks: communication, clustering and aggregation. *Ad Hoc Networks*, 2(1):45–63, 2004.

[16] M. Negnevitsky. *Artificial intelligence: A guide to intelligent systems*. Addison-Wesley, Reading, MA, 2001.

[17] C.E. Perkins and E.M. Royer. The ad hoc on-demand distance-vector protocol. 2001.

[18] T. Shu, M. Krunz, and S. Vrudhula. Power balanced coverage-time optimization for clustered wireless sensor networks. In *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, page 120. ACM, 2005.

[19] T. Voigt, A. Dunkels, J. Alonso, H. Ritter, and J. Schiller. Solar-aware clustering in wireless sensor networks. In *Proceedings of the Ninth IEEE Symposium on Computers and Communications*, 2004.

[20] O. Younis and S. Fahmy. HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, pages 366–379, 2004.

[21] O. Younis, M. Krunz, and S. Ramasubramanian. Node clustering in wireless sensor networks: Recent developments and deployment challenges. *IEEE Network*, 20(3):20–25, 2006.

[22] JY Yu and PHJ Chong. A survey of clustering schemes for mobile ad hoc networks. *IEEE Communications Surveys & Tutorials*, 7(1):32–48, 2005.

[23] H.J. Zimmermann. *Fuzzy set theory–and its applications*. Kluwer Academic Pub, 2001.