1-1-2012

An Improvement on Low-Energy Adaptive Clustering Hierarchy (Leach) Porotocol for Wireless Sensor Networks

Negin Behboudi
Ryerson University

AN IMPROVEMENT ON LOW-ENERGY ADAPTIVE CLUSTERING HIERARCHY (LEACH) POROTOCOL FOR WIRELESS SENSOR NETWORKS

by

Negin Behboudi

BSc., Islamic Azad University, Iran, 2006

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Science

In the program of Computer Science

Toronto, Ontario, Canada, 2012

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research

NEGIN BEHBOUDI

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

NEGIN BEHBOUDI

AN IMPROVEMENT ON LOW-ENERGY ADAPTIVE CLUSTERING HIERARCHY (LEACH) POROTOCOL FOR WIRELESS SENSOR NETWORKS

Negin Behboudi

Master of Science, Computer Science, 2011

Ryerson University, Toronto, Ontario, Canada

ABSTRACT

Based on LEACH, a new clustering protocol for wireless sensor networks is proposed and implemented. A new visualization method is also introduced.

There are two outcomes from the implemented protocol. The first outcome is prolonged network lifetime. The second outcome is an increase in flexibility of the location of the base station. Another contribution of this thesis is development of a visualization tool that helps users to understand the energy behavior of the sensors in similar applications.

The first outcome —prolonged network lifetime —are due to considering the distance of each node from the base station while clusters are formed. The energy dissipation for transmitting certain amount of data is defined as a piecewise function which is divided by certain distance threshold. A piece of this piecewise function is implemented in this work, which leads to increased flexibility in the location of the base station.

ACKNOWLEDGEMENTS

I would like to extend my gratitude towards my supervisor Dr. Abolreza Abhari. Without his invaluable insight and continuous assistance, this work would impossible.

I would also like to thank my parents for the invaluable love and unending support that they have given to me during the last two years.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ACRONYMS

ADC analog-to-digital –converter

BS Base Station

CH Cluster Head

EDACH Energy-Driven Adaptive Clustering Hierarchy

GPS Global Positioning System

LEACH Low-Energy Adaptive Clustering Hierarchy

LEACH-C LEACH-Centralized

LMAC Lightweight Medium Access Control

MAC Medium Access Control

µAMPS micro-Adaptive Multi-domain Power-aware Sensors

RSSI Received Signal Strength Indication

S-MAC Sensor MAC

SPIN Sensor Protocols for Information via Negotiation

TDMA Time-Division Multiple Access

TREEPSI Tree based Energy Efficient Protocol for Sensor Information

TBC Tree based Clustering for Energy Efficient Wireless Sensor Networks

TEEN Threshold Sensitive Energy Efficient Sensor Network

WSN Wireless Sensor Network

WINS Wireless Integrated Network Sensor

CHAPTER 1

INTRODUCTION

“Wireless sensor networks (WSNs) have been identified as one of the most important technologies for the 21st century.”[1]. The tiny, low cost and low power sensors are able to communicate within a short range and work together to form a sensor network for gathering data from a field. These sensors have data processing and communication capabilities [2]. They have also enabled us to monitor and collect data in any environment. They sense the conditions in which they are surrounded and transform their data to electronic signals. The electronic signals are transmitted over radio waves to the base station (BS). Processing such electronic signals reveals some valuable characteristics of that environment. The usefulness of WSNs is more noticeable when they are used in inaccessible areas since there is no need to adhere to a specific network structure. Another unique feature that represents a significant improvement over traditional networks is the cooperative effort of sensor nodes [3]. Raw data is collected by sensor nodes. Since the sensor nodes are equipped with an on-board processor, the raw data may be manipulated as desired. For instance, for a sensor node collecting temperature data the values retained may be limited to temperatures less than a certain threshold. As the main power source for all nodes is a battery, the energy supply for each sensor node is constrained. The primary goal in designing WSNs is maximizing network lifetime as it is impractical to change or replace exhausted batteries [4]. Such constraint necessitates energy awareness in designing WSNs.

There are two competing objectives in the design of WSNs. The first objective is the capability to exchange large amount of data between the nodes and the base station. The second constraining objective is minimizing the energy consumption. The two competing objectives reveal the importance of efficient routing protocol in WSNs. Therefore, many routing algorithms have been proposed due to the challenges in designing an energy efficient network. Among all the proposed methods, hierarchical routing protocols greatly satisfy the limitations and constraints in WSNs [5]. Hierarchical routing protocols, also known as cluster-based routing, is mainly considered as a two layer architecture where one layer is engaged in cluster head selection and the other layer is responsible for routing. A cluster head (CH) in hierarchical routing is the node which is responsible for collecting data from other nodes in the cluster, aggregating all data and sending the aggregated data to the base station.

A specific clustering protocol known as LEACH (Energy-efficient communication protocol for wireless microsensor networks) [6] is analyzed in this work. As part of this work, our analysis of LEACH leads to the development of a new energy-efficient protocol known as WEEC (A Weighted Energy Efficient Clustering for Wireless Sensor Networks) [7].

When working with a large amount of time varying data, another important issue that should be considered is the graphical representation of such data to aid in the visual identification of network behaviour. Energy consumption is central to this work and the energy level of each node in the WSN is of particular interest, since the energy level of each node is finite. An accurate and effective visualization tool would provide a quick and accessible means to view the energy level of each node in the field to support the development of routing algorithms that minimize energy consumption.

1.1 Motivation

Sensor nodes are used for event detection, continuous sensing and local control of actuators [3]. There are many applications due to the wireless connection and micro-sensing features of WSNs. For example in military applications, WSNs enable commanders to constantly monitor the status of their troops. Moreover, information about the condition and availability of the equipment in the battlefield could be obtained by using WSNs [3]. Health applications are other examples of applications of WSNs that highlight the importance of research in this area. Accidental falls are especially hazardous to the health of elderly people. Such accidents could be monitored by the installed sensor networks [8, 9]. The installed sensor network could also be used for detecting heart attack as well as monitoring blood pressure. However, for developing useful and efficient applications, the challenges and obstacles apparent in the design of such networks should be properly addressed and solved. One such challenge is particularly important and relates to the development of energy efficient WSNs. Sensor nodes in WSNs' applications are battery constrained thus innovative techniques are needed to eliminate energy inefficiency that shorten the network lifetime.

From the military perspective, critical data may be required at certain point in the future from nodes where energy levels are low. Therefore the identification of certain nodes which their energy level is less than a threshold is needed. Without a means to visualize the energy levels of each node, it is difficult to assess the energy distribution in the WSN and node failures may be lead to catastrophic results. A tool to visualize the residual energy of each node provides a method to avoid such events.

In the medical field, a tool to visualize the residual energy of WSN nodes may also prove useful. As explained in the healthcare example previously, the monitoring of patients requires constant collection of crucial data to detect, for example, falls, imminent heart attacks or dangerous blood pressure levels. Maintaining such data collection rates requires efficient energy consumption, while ensuring that node failures are identified prior to their occurrence. Monitoring the residual energy level of each WSN node with an effective visualization tool is a method to prevent such node failures.

1.2 Objective and Scope

The main objective of this thesis is to improve upon the LEACH protocol [6] and propose an optimized algorithm for the clustering in order to prolong network lifetime. Prolonging network lifetime is the way to provide energy efficient WSNs. This research was inspired by the fact that LEACH occupies a very important position in the area of wireless sensor networks [10]. Most of the hierarchical routing algorithms, aiming to prolonging network lifetime, have been derived from the LEACH.

In WSNs, sensing, computation and communication are three parameters that consume power. Minimizing the communication cost is the primary concern while sensing and computation parameters are secondary objectives in designing such networks. This is because communication cost is higher than sensing and computation costs [4]. Hence, the main objective of the current work is minimizing the communication cost by appropriate cluster-head node election.

In hierarchical routing protocols, cluster heads consume more energy since they are in charge of data aggregation as well as communications inside and outside of the cluster. The secondary objective of this work is to achieve a network with an optimum number of cluster heads at each round of data transmission. Although the focus of this work is on clustering, there are other techniques for reducing energy consumption such as topology control.

A requirement in some algorithms for clustering is that the location of the base station should remain at a specific distance from the field. The third objective of the work is to remove the fixed distance requirement to increase flexibility in the location of the base station. For achieving these objectives some assumptions have been made.

The final objective of this thesis is to develop a new method for the visualization of the residual energy data of WSN nodes to aid users in understanding the energy levels of WSN nodes deployed across the field.

There are some assumptions have been made in the WEEC protocol and that most of them are directly came from the LEACH. The initial energy of all the nodes is the same; the communication environment is considered to be error free; the location of nodes are known by the base station; all the nodes are able to send data and receive data from the base station; the base station has no energy limitation; all the messages that are being sent have the same number of bits and all sensor nodes are the same in terms of size and performance.

1.3 Thesis Contributions

Heinzelman et al. proposed LEACH protocol in [6]. The protocol described by Heinzelman et al. is modified in this work which leads to the development of the WEEC protocol. The major contributions of this work can be summarized as following:

Improved network lifetime: To achieve this goal, WEEC considers the distance of the nodes to the base station as an important factor in cluster head selection phase and assigns a probability to each node which is analytically derived from the distance of the nodes to the base station.

Flexibility in the location of the base station to the field: WEEC precisely considers all the factors and formulas in LEACH. By analyzing the formulas in LEACH, another formula is derived that is used in WEEC and can support any distance of the base station to the field.

An effective visualization tool: We also designed a new visualization method for representing the nodes' residual energy in the field. This tool utilizes actual data to provide a visual representation of the residual energy level of each node. By mapping residual energy data to the colour and size of three-dimensional cubes, we have enhanced and clarified the knowledge of the residual energy level of each node for the user. This visualization tool shows the changes of sensors energy level in each iteration. Furthermore, this visualization tool is not application-specific and can be used by any similar algorithm that deals with energy management of WSNs and needs to monitor the energy level of the nodes based on their locations.

Although the network lifetime has been prolonged in the WEEC protocol in compare to the LEACH, for applications in which having more areas covered by sensor nodes is desired LEACH would be more efficient. In contrast, in applications where obtaining data from small areas of the field is considered important, WEEC would be more efficient.

1.4 Outline

The remainder of the thesis is organized as follows. Chapter 2 provides background information related WSNs for general understanding and describes challenges related to WSNs. Various hierarchical clustering protocols in this field are also discussed in Chapter 2. We present our work in Chapter 3 which begins with a description of our main contribution, namely the WEEC protocol. Following the description of the WEEC protocol a comparison is drawn between the LEACH and WEEC protocols to highlight their similarities and differences. Chapter 4 offers simulation results for both LEACH and WEEC protocols.

Chapter 5 summarizes and concludes this work and suggests some directions for future research.

CHAPTER 2

BACKGROUND AND RELATED WORKS

This chapter is an overview of the basic concepts of WSNs as well as an introduction to some of the important routing protocols. Section 2.1 contains description of WSN and its characteristics. Section 2.2 presents works related to routing protocols in WSNs. In Section 2.3 the importance of visualization in WSNs is explained as well as some works that have been done in this area.

2.1 Wireless Sensor Network

The recent advances in wireless technologies have enabled the smaller and less expensive products which enhance communication speed significantly. Since early 1990s, the research on wireless sensor networks has intensified due to important applications they support such as target tracking and remote environmental monitoring. Two examples of applications of WSNs include biomedical health monitoring [11, 12] and natural disaster relief [13]. Annually, numerous workshops and conferences with focus on WSNs are being held.

2.1.1 Components and Characteristics

Wireless Sensor Networks consist of hundreds or thousands of nodes. Since most of the times the position of the sensors does not need to be pre-determined, they randomly deployed in any inaccessible area. For measuring the properties of the environment, in which they are located, they can be equipped with various sensors such as optical, thermal or mechanical. Having an on-board processor enabled this type of network to carry out some computations and transmit the required data instead of transmitting the raw data. Figure 2.1 shows a typical WSN with nodes scattered in the field [14].

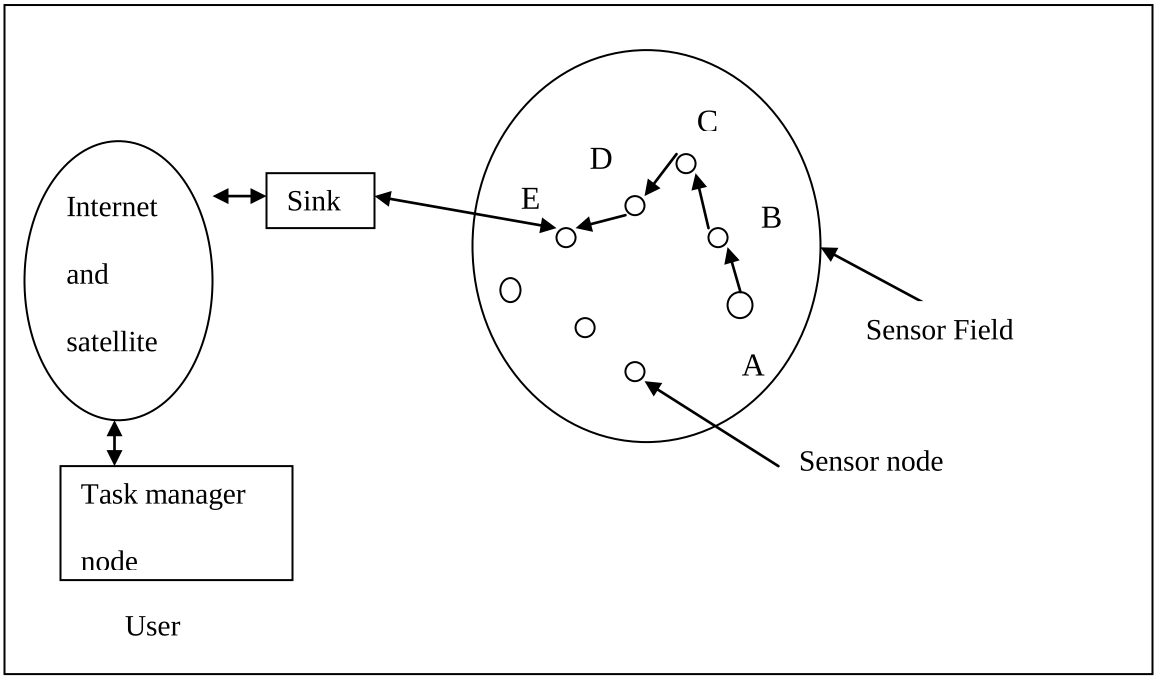


Figure 2.1: Sensor nodes scattered in a field [14]

In Figure 2.1 the sensor field is the total area covered by sensor nodes. Each of the sensor nodes shown in Figure 2.1 has the ability to sense the environment parameters. When node A transmits data to the base station it follows multi-hop routing protocol. Node A transmit data to node B which then transmits to node C. Node C then forwards the data to Node D. Finally node E aggregates its own data with data sent from nodes A, B, C and D and then sent it to the base station. In this type of the network sink (base station) has more computational, power and communication resources and acts as gateway between the sensor nodes and other type of networks such as internet or satellite.

2.1.2 Hardware

AS shown in Figure 2.2, each sensor node consists of four major components: sensing unit, processing unit, power unit and transceiver unit [14].

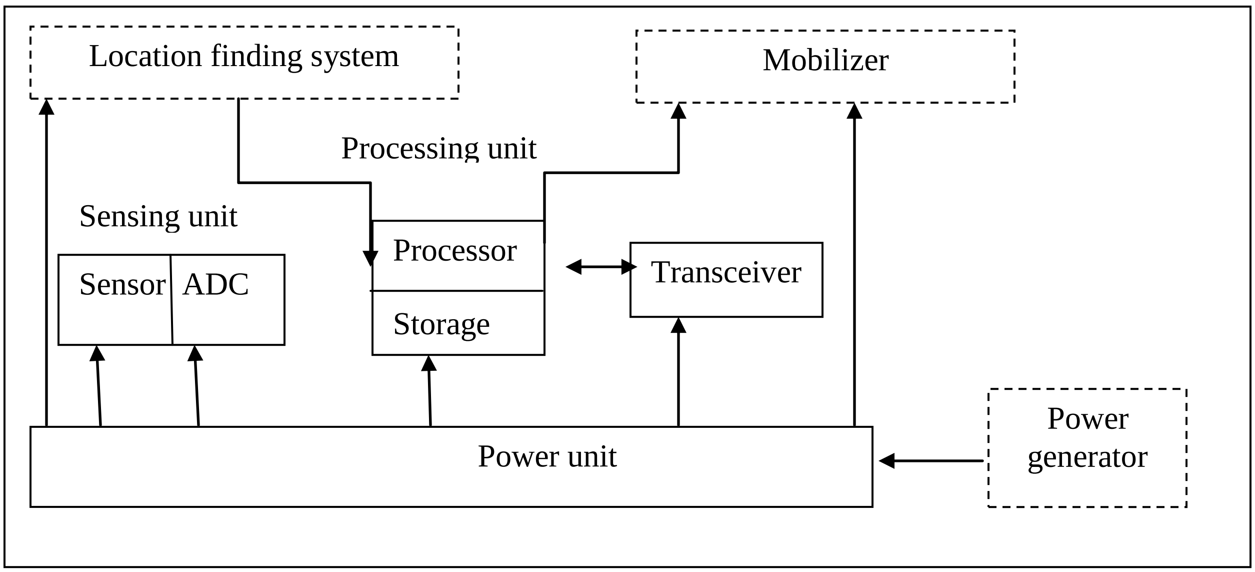


Figure 2.2: The components of a sensor node [14]

Sensor unit consists of sensor and analog-to-digital –converter (ADC) [14]. Sensor collects data and only transmits it if certain pre-defined conditions are met. The data is transmitted as an analog signal to the sensor and then is converted to digital signal by ADC. The processing unit uses the digital signal and analysis the attribute of the sensed data. Transceiver unit connects the node to the network via radio transmitter. In addition to these four basic units, there are some application-dependent components such as location finding system and mobilizer. Depending on the application, some networks may need the exact location of the sensor nodes. These types of sensors are equipped with location finding system. Sensors that should be moved in the field due to carry out certain tasks in the network are equipped with mobilizer [14]. Each of the basic units faces tremendous design challenges under constraints of size, energy and cost. Properly gathering information is one of the most important issues that should be improved in sensing units. On the other hand, transceiver units need efficient protocols to prolong the lifetime of the batteries.

Because of the world wide attention in the topic of WSNs, there are so many sensor node products offered in the market every year. Some example of these products are WINS (Wireless Integrated Network Sensors) [15], µAMPS (micro-Adaptive Multi-domain Power-aware Sensors) [16], MICA2 [17], ZSTAR [18] and PicoRadio [19]. There are also many choices available for processors, sensors and power supplies, which provides an indication that there is a large variety of sensor units in the market [20].

2.1.3 MAC Layer Protocol

In WSNs, sensing, computation and communication are three tasks that consume power. Among these factors communication consumes more energy [4]. Therefore in WSNs the energy consumption due to communication is higher than energy consumed for performing computations. One way to reduce energy consumption due to communication is to apply an efficient medium-access control (MAC) protocol. MAC protocol controls antenna activities and is responsible for reliable connections and has an important role in having a successful and collision-free network. Energy efficiency, device management and efficient resource sharing are three key factors that should be considered in design MAC layer protocol [4].There are many

MAC protocols that have been developed to achieve these goals. Since most of the current MAC designs for WSNs are broadly divided into TDMA (Time Division Multiple Access) protocol [21], we first describe TDMA protocol and provide two examples that use TDMA in designing their medium access control.

TDMA (Time Division Multiple Access): The best choice for an energy efficient MAC protocol in WSNs is TDMA [22]. TDMA protocols create a time schedule for transmitting and receiving data for each node in the network. Based on the schedule nodes are just active during the assigned time slots and shut down their radio interfaces other times. Shutting down the radio interface during inactive period leads to energy saving. Preventing transmission interference is one of the most important issues that should be considered in designing TDMA. Transmission interference may cause by two neighbor nodes sending data at the same time or by two non-adjacent nodes sending data to the same receiver. The importance of rescheduling should not be ignored in designing TDMA. Updating the schedule can be done either on demand when the topology of the network changes due to issues such as node failure or periodically [22].

S-MAC (Sensor MAC): S-MAC [21] protocol designed merely for wireless sensor networks. In this protocol time is divided into long frames and each frame has an active part and sleeping part. Nodes turn off their radio during sleeping time to preserve energy and communicate with their neighbors during the active part. In fact during the active part all the messages that have been queued during sleeping part would be sent. In other word, S-MAC tries to save energy by avoiding overhearing. Moreover, the energy consumed on idle listening is reduced since all the messages are packed in the active part. However, if message generating event occurs during the sleeping part the latency increases because in that case the message should be queued until the start of the active part [23].

LMAC (Lightweight MAC): LMAC [24] uses the TDMA (Time-Division Multiple Access) approach to provide a collision free network. LMAC applies a distributed algorithm as explained in [25] for dividing time slots among nodes unlike TDMA that divides the time slots by a central manager. Nodes organize time into slots and each slot is divided to three sections: communication request, traffic control and data section. When a node wants to send its data, it waits for its time slot and broadcasts a packet consist of the destination and length in the control section and immediately after transmits the data. Nodes that are not an intended receiver turn off their radio during the data transmission. LMAC extends the network lifetime by 3.8 times in compare to S-MAC [24].

2.1.4 Network Routing Protocols in WSNs

2.1.4.1 Routing Challenges and Design Issues in WSNs

Since WSNs are restricted in terms of bandwidth, battery and power consumption, a major goal in the design of WSNs is prolonging network lifetime. Therefore, many factors should be considered and several challenges should be addressed. The following are the factors that need to be considered: node deployment; energy consideration; scalability and fault tolerance.

Nodes can be deployed in the field either manually or randomly. In manual deployment, nodes are manually located in the field and the routing paths are predetermined and known. In random deployment, nodes randomly scattered in a field.

WSNs are battery driven and there is usually no option for recharging. Therefore an energy efficient form of communication is essential. In multi-hop networks, explained in section 2.1.1 and illustrated in Figure 2.1, each node performs both as a transmitter and a receiver. As such, power failure in nodes causes significant impact on the network and force routing changes that affect the topology of the network.

One of the most important design issues in WSN is scalability. The number of sensor nodes in a field may extend hundreds or thousands. As such any routing protocol should be able to work with such huge network.

In WSNs there are situations that nodes fail due to lack of power, environmental interference or physical damage. The nodes' failure should not impact on the overall task of the network and the MAC and routing protocols should provide other routes in order to keep transmitting the data to the base station.

2.1.4.2 Flat and Hierarchical Routing Protocols

Network routing protocols are in charge of routing scheme as well as maintaining the network structure in WSNs. There are three types of network structure: flat routing [26, 27], hierarchical routing [28, 29 and 30] and location-based routing [31, 32, 33 and 34]. However, in order to focus in our area of research, we present further discussion of only flat and hierarchical routing protocols.

Flat Routing: In flat routing protocols nodes play the same role and have similar functionality in transmitting and receiving data. In this type of network it is not possible to assign a global identifier to each node due to large number of nodes. Therefore, base station send queries to different part of the field and waits for the data from sensors in selected parts of the field. This approach is called data centric routing [5]. SPIN (Sensor Protocols for Information via Negotiation) [35] and DD (Direct Diffusion) [36] are two examples of the data centric routing protocols that save energy by data negotiation and omitting the redundant data.

Hierarchical (Cluster-based) Routing: In this kind of routing method, nodes play different roles in transmitting and receiving data. Some of the nodes are responsible for processing and communication, while other nodes can be used for sensing the target area. Hierarchical routing is mainly considered as two layer architecture where one layer is engaged in cluster head selection and the other layer is responsible for routing. Cluster head in hierarchical routing is the node which is responsible for collecting data from other nodes in the cluster, aggregating all data and sending the aggregated data to the base station. Creating clusters and assigning communication task to cluster heads contributes to a more scalable and energy efficient network [5]. The main goal of all the hierarchical routing protocols is to appropriately create clusters and choose cluster heads in order to reserve energy in the network. Some examples of the hierarchical routing protocol will be presented in section 2.2.

Followings are some of the advantages of hierarchical protocols summarized from the related research [5, 37 and 38].

Hierarchical Routing is a feasible solution for reducing energy consumption in WSNs. Within a cluster, cluster head manages the member nodes and assigns them tasks which lead to reduction in redundant data transmission. Moreover, cluster head has some responsibilities such as data collection and data aggregation from their respective cluster members. Energy consumption greatly reduced in this kind of routing method since the total data messages sent to the base station is minimized by data aggregation.

Hierarchical Routing effectively assigns each node different task according to the ability of that sensor node. This approach offers balanced distribution of energy in the network. It can achieve by selecting higher energy nodes to perform the responsibility of cluster heads while lower energy perform sensing duties in the target area.

Hierarchical Routing can easily achieve collision free network by applying a proper MAC protocol explained in section 2.1.3. After creating clusters, it is the responsibility of the cluster heads to create a transmission schedule for the member nodes and broadcast it to all the nodes in its respective cluster. Sensor nodes send, receive and listen data based on the assigned time slot and sleep other times in order to conserve energy in the system. By using the hierarchical routing protocol the number of data collision between the nodes would be reduced [37].

2.2 Literature Review

In this section, an overview of some of the significant related work is provided with the focus on the hierarchical protocols.

2.2.1 LEACH: Low-Energy Adaptive Clustering Hierarchy for Wireless Microsensor Networks

Heinzelman et al. [6, 39], proposed a new adaptive clustering algorithm in which the strategy of selecting the nodes acting as cluster heads is random and rotates among the nodes in each cluster. If there are n nodes in the cluster, LEACH guarantees that during r rounds all the nodes would be selected as cluster head once. This approach solves the problem of conventional clustering algorithms in which cluster heads are fixed during the network lifetime. When cluster heads are fixed, nodes chosen to be cluster heads would die soon while other nodes' batteries are almost full.

The radio hardware dissipation model assumed in LEACH [39] is a simple radio model. In this model, when the distance of a node transmitting data to other nodes or the base station is greater than the threshold d_0, the multipath (mp) fading channel model is used (d^4 power loss). When the distance between a node transmitting data to other nodes or the base station is less than d_0, the free space (fs) channel model is used (d^2 power loss) [39]. Therefore the energy dissipates by the radio to transmit l bit message to the distance of d calculates as shown in formula (2.1):

$$E_{TX}(l,d) = \begin{cases} lE_{elec} + l\epsilon_{fs}d^2 & \text{d} < d_0 \\ \\ lE_{elec} + l\epsilon_{mp}d^4 & \text{d} >= d_0 \end{cases} \qquad (2.1)$$

In formula (2.1) l is number of bits, E_{elec} is the energy dissipation to run the radio electronics, $\in_{fs}$ and ϵ_{mp} are the energy dissipation values to run the amplifier for close and far distances respectively.

LEACH consists of two phases: Set up phase (cluster formation phase) and the steady-state phase.

Set up Phase:

First step is cluster head selection. At the first of each round, each node selects a random number between 0 and 1 and compares it to the threshold T shown in formula (2.2). If the selected random number is less than the threshold T, the node would be selected as a cluster head for the current round. Assuming there are N nodes in the field with k number of cluster heads, LEACH ensures that all the nodes become cluster heads only once in every $\frac{N}{K}$ round. This can be done by defining an indicator function ($C_i(t)$) confirming if the node has become a cluster head in the recent rounds ($r\ mod\ \frac{N}{k}$) or not.

$$T_i(t)= \begin{cases} \frac{K}{N-K*(r\ mod\ \frac{N}{k})} & C_i(t)=1 \\ 0 & C_i(t)=0 \end{cases} \qquad (2.2)$$

In formula (2.2), the value of zero for $C_i(t)$ implies that the node i has become cluster head in the last ($r \; mod \; \frac{N}{k}$). In this case node i would not participate in cluster head selection phase. r is the number of current round and k is the optimum number of cluster heads (K_{opt}). The value of k has derived from formula (2.3) and is set to 5 for certain criteria shown in Table 2.1.

$$K_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}} \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \frac{M}{d^2{}_{toBS}} \tag{2.3}$$

Table 2.1: Parameters set in LEACH experiments

Parameter	**Value**
Area of the field	100m*100m
Location of BS	50m,175m
Number of Nodes	100
E_{elec}	50 nJ/bit
ϵ_{fs}	10pJ/bit/m^2
ϵ_{mp}	0.0013pJ/ bit/m^4
E_{DA}	5 nJ/bit/signal
Data Packet size	500 byte

In Table 2.1 E_{elec} is the electronic energy, ϵ_{fs} *and* ϵ_{mp} are the energy dissipation values to run the amplifier and E_{DA} is the energy for data aggregation.

After cluster head selection phase, all the selected cluster heads send an advertisement message to all the non-cluster head nodes in the field. Based on the received signal strength of the advertisement message, the non-cluster head nodes decide their cluster heads for the current round and send back a join request message to their selected cluster heads informing their membership which leads to cluster formation. For avoiding collision in the network LEACH uses

TDMA schedule, explained in section 2.1.3. According to the number of nodes in each cluster, the cluster head creates a TDMA schedule and broadcasts it to its member nodes.

Steady State Phase

The steady state phase is the data transmission step. During this phase, nodes in each cluster send their data based on the allocated transmission time to their local cluster heads. To reduce the energy dissipation, the receiver of all non-cluster head nodes would be turned off until the nodes' defined allocated time. After receiving all the data from the nodes, the cluster head aggregates all the data sent from the member nodes into a single signal and transfers it to the base station.

2.2.2 TREEPSI (Tree based Energy Efficient Protocol for Sensor Information)

Satapathy at al. in [40], proposed a clustering protocol which is an improvement over LEACH. In TREEPSI, the nodes are distributed randomly in the field. The next step is constructing the hierarchical tree which can be done either by the base station or using a common algorithm in each node. For collecting sensed data from the field, two methods are introduced. In the first approach, the root node sends a small control packet to all the children nodes, using a standard tree traversal algorithm while in the second approach, all the children nodes start sending their data to their parents until it reaches the root node. However, the second approach needs a multiplexing scheme such as TDMA for avoiding collisions. Having all the fused data, the root of the tree sends all the gathered data to the base station. The root node of the tree is fixed until it dies due to a lack of battery power. After the death of the root node, another node will take over and become a root and a new tree-like path will be constructed. TREEPSI is simulated in both (50m*50m) and (100m*100m) fields with the base station located at (25m, 150m) and (50m, 300m) respectively. In order to improve the routing protocol in the tree, the authors consider a threshold distance for choosing a parent and its child. If the distance between two nodes is more than a predefined threshold, these two nodes will not considered as parent and child since there should be a node that is located closer. Although TREEPSI is an improvement over LEACH, the tree path is not reconstructed until the root node dies and this causes imbalanced energy consumption in the field.

2.2.3 TBC (Tree based Clustering for Energy Efficient Wireless Sensor Networks)

To solve the unbalanced energy consumption of the nodes in TREEPSI [40], Kim et al in [41] proposed another Tree-based clustering protocol known as TBC (Tree-Based Clustering for Energy Efficient Wireless Sensor Networks). TBC is an improvement over LEACH and TREEPSI. In the TBC algorithm, the first phase is the cluster formation. Cluster Head selection is based on the LEACH algorithm with the same threshold explained in formula (2.2). After selecting cluster heads for the current round, the selected cluster heads send an advertisement message to all other nodes in their cluster. The nodes which received the advertisement message choose their cluster heads based on received signal power. The nodes also send the Join-REQ message to their selected cluster heads. The message sent to the cluster heads includes the node's ID as well as the location of the sender node. Constructing a tree in each cluster is the next step. The distance of the nodes to the cluster head is used to determine the level of a tree inside each cluster and the tree would be formed after level determination. Next, the cluster head makes a decision and chooses a parent for all the nodes in the cluster. Accordingly, every node in the cluster would send their data to their assigned parents based on the TDMA scheduling that the cluster head had broadcasted to all the cluster members. The main difference between the

TBC and the TREEPSI protocols is that the TBC constructs several trees while in the TREEPSI just one tree is formed in the field. For simulation results, authors deployed 100 nodes randomly in field of (100m*100m) with the base station located at (50m, 120m) from the field. Two levels of initial energy (0.25J, 0.5J) are assumed with a data packet size of 500 bytes. The radio model assumed in the TBC is the same as LEACH. The free space model is used between the cluster head and non-cluster head nodes while the multipath model is assumed for the communication between cluster heads and the base station.

2.2.4 EDACH (Energy-Driven Adaptive Clustering Hierarchy)

In EDACH [42], authors proposed another improvement over LEACH by dividing the field into three parts as shown in Figure (2.3):

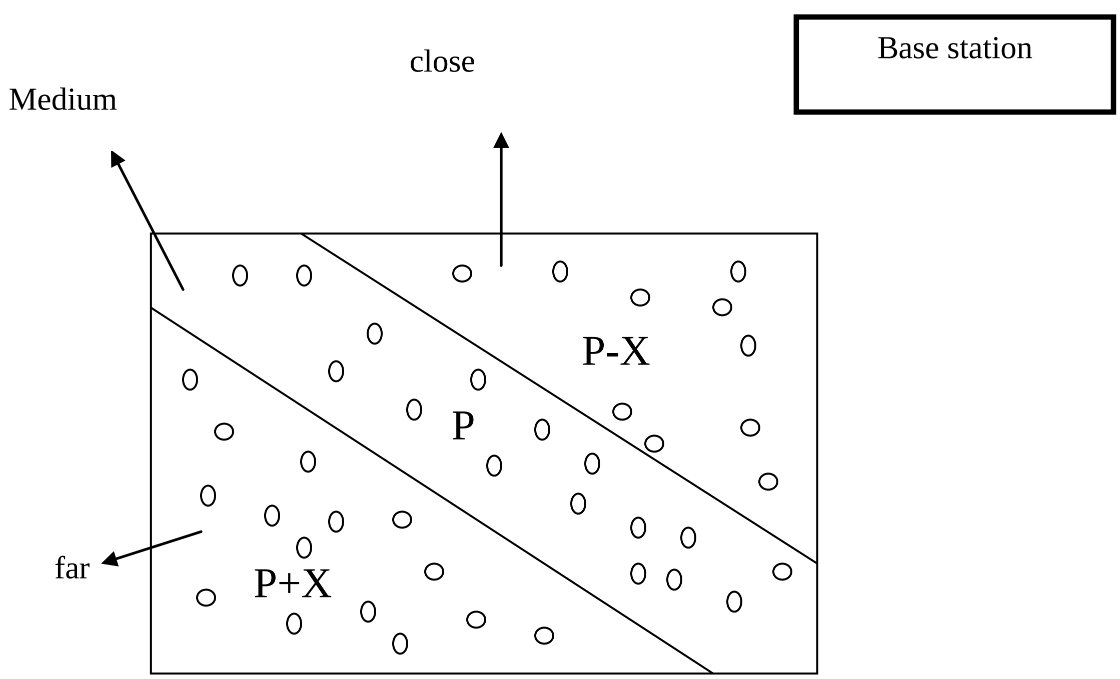


Figure 2.3: Partition of nodes in EDACH [42]

In Figure (2.3), P is the percentage of nodes that will become cluster heads. Since the energy dissipation of the nodes in the clusters is proportional to the distance between the nodes and base station, EDACH forces more nodes to become cluster heads in areas farther from the base station and fewer nodes to become cluster heads in areas closer to base station. The value of X shown in Figure (2.3) is a number between zero and one. Therefore, the threshold for selecting the cluster head is different for different areas of the field. For the nodes located far, medium and close to the base station the value of p is: (1+X) P, P, (1-X) P respectively. Since cluster heads in the clustering algorithms have to receive many packets, their batteries drain more quickly. To solve this problem, EDACH replaces the low-battery cluster head nodes by using a proxy node. The selection of the proxy node is based on the threshold shown in formula (2.4):

$$E_{TH} = \frac{1}{k}\sum_{j=1}^{k} E_{CH(i)} \qquad (2.4)$$

k in formula (2.4) is the cluster head's length of aggregated message. The calculation of proxy node starts after the selection of cluster heads. In other words, if the battery power of the cluster head has depleted and drops below the E_{TH} value, the proxy node election takes place to replace the selected cluster head with another node known as the proxy node. Next, the proxy node broadcasts a message and a new TDMA schedule to all the nodes in the cluster. EDACH's experiments have been done in a (50m*50m) field with the base station located at (85m, 90m). Two models of residual battery power have been selected, 0.5J and a random value between 0.25J and 0.5J, and the size of messages is set to 2000 bits. The radio model assumed in the EDACH is the same as LEACH [6].

2.2.5 TEEN (Threshold Sensitive Energy Efficient Sensor Network)

Manjeshwar et al., in [43], proposed another algorithm for time critical applications. Since sensing the environment consumes less energy than transmitting the sensed data, TEEN tries to minimize energy consumption by reducing data transmissions. There are two thresholds in this algorithm, hard and soft, which are sent to all member nodes by cluster heads. The hard threshold value is the value of the sensed attribute, while the soft threshold is a small range of values near the hard threshold. Therefore, the hard threshold minimizes the number of data transmissions. The reason is that sensor nodes could only transmit their data while the current value of their attributes hit the hard threshold. The soft threshold reduces more transmissions when there is a minor change in the sensed attribute. New values for both hard and soft thresholds are set, every time that cluster heads change. Figure (2.4) shows the TEEN time line:

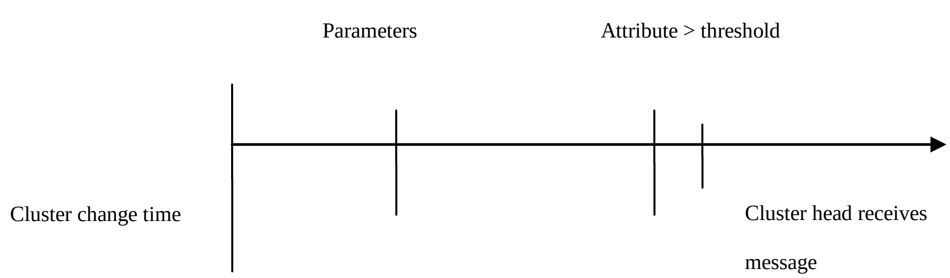


Figure 2.4: Time Line for TEEN [43]

However the energy consumption in the TEEN protocol is decreased in compare to the LEACH. The main weakness of TEEN is the parameter reception issue. If the nodes could not receive the thresholds, they would never send any data and no communication occurs in the network.

2.3 Visualization

Data representation and has become a crucial issue in WSNs since data is collected every few minutes and dealing with raw data is difficult. Hence, developing a visualization platform that collects raw data and transforms it into visual format is strongly needed [49]. WSNs afford limited visibility to in to the applications. The reason is the resource constraint features as well as the absence of user interfaces [50]. Moreover, the hardware design and the communication behavior of such network [51, 52] are two more obstacles in order to have a user friendly and error free WSNs applications. With visualization platforms users are able to easily observe the status of WSNs. Developers of WSNs have been implemented visualization tools. Here are some two of the examples that have been done is this area:

2.3.1 SNAMP: A Multi-sniffer and Multi-view Visualization Platform for Wireless Sensor Networks

Yang et al. in [50] introduced a visualization platform for WSNs which presents network topology; sensing chart view; Packets view and measurement view. In the visualization the network topology and packet rout are shown by colourful lines and node's power is presented by the progress bar under the node. Moreover, the SNAMP software adapts a multi-view design in order to make the software clear and easy to observe. However, SNAMP is not only used for visualizing WSNs, it can also be used for debugging WSNs. The software is able to record and replay network activities so users can watch those recorded activities with different speed. It is also possible to select a route on the topology view and view all the packets associated with that route.

2.3.2 The Implementation of Wireless Sensor Network Visualization Platform based on Wetland Monitoring

In [49], Hu et al. developed a visualization platform for WSNs. In this work authors visualized the routing data which presents the information of the network topology such as nodes' distribution and their relationships. In this visualization users can easily analyze the network status and find out if the network topology should be altered or not. This work is done in three steps: first the nodes and their connections are extracted from the database. Next, the extracted information is stored in arrays. Finally, logical coordination is assigned to each node based on their locations and the shapes and lines are drawn for presenting nodes and their connectivity.

CHAPTER 3

A Weighted Energy Efficient Clustering (WEEC) for Wireless Sensor Networks

This chapter presents a new routing protocol for WSNs. In subsection 3.1 the approach to design the proposed routing protocol and the difference between the proposed routing protocol and the LEACH [6] protocol are discussed. Subsection 3.2 explains the visualization algorithm and finally subsection 3.3 concludes the proposed algorithm.

3.1 Our proposed method: A Weighted Energy Efficient Clustering

3.1.1 Definitions

In our proposed algorithm, we improved the LEACH protocol and proposed a weighted energy efficient clustering (WEEC) algorithm [7] for WSNs. We take into consideration the location of each node while clusters are forming. The simulation result proves that our proposed scheme noticeably increases the life time of the network. We also take advantage of clustering infrastructure and data aggregation as used in LEACH. At first we explain the system and radio model in our proposed algorithm. Apart from assumptions iv. and viii., the following assumptions are drawn directly from the LEACH.

System Model

i. The initial energy of all the nodes is the same.

ii. The communication environment is error free.

iii. All the nodes are able to send data and receive data from the base station.

iv. The location of nodes is known by the base station using RSSI-based localization technique [44].

v. Sensor nodes are uniformly distributed in a two dimensional field.

vi. The base station has no energy limitation.

vii. All the messages that are being sent have the same number of bits.

viii. All sensor nodes are the same in terms of size and performance.

ix. The location of the base station is fixed during data transmission.

Radio Model

The radio model used in our algorithm is the same as LEACH which is described by formula (2.1) in chapter 2, section 2.2.1.

For the importance of the radio model and it's relation to distance, we re-described it in this chapter. Figure (3.1) represent a network model in which the base station located far from the field. Far in this case is defined as any distance which is greater than the threshold d_0. In this model since the distance of the base station to the field is greater than the threshold d_0, we consider that base station is located far from the field. When data is sent within a large distance (greater than d_0), the energy dissipation for data transmission of l bits message follows the multipath model [45]:

$$E_T(l,d)=lE_{elec}+l\epsilon_{mp}d^4 \qquad (3.1)$$

Where l is number of bits, E_{elec} is the energy dissipation to run the radio electronics and ϵ_{mp} is the energy dissipation to run the amplifier.

Thus in Figure (3.1), the communications between cluster heads (shown as black dots) and the base station follows the multipath model. Moreover, "Close" is defined as situations in which the distance throughout data transition is less than the threshold d_0. In this case the energy dissipation for transmitting data follows a free space model. Therefore in Figure (3.1), the energy dissipation inside the cluster (while nodes sending data to black nodes) follows the free space model:

$$E_T(l,d) = lE_{elec} + l\epsilon_{fs}d^2 \qquad (3.2)$$

Where l is number of bits, E_{elec} is the energy dissipation to run the radio electronics and ϵ_{mp} is the energy dissipation to run the amplifier.

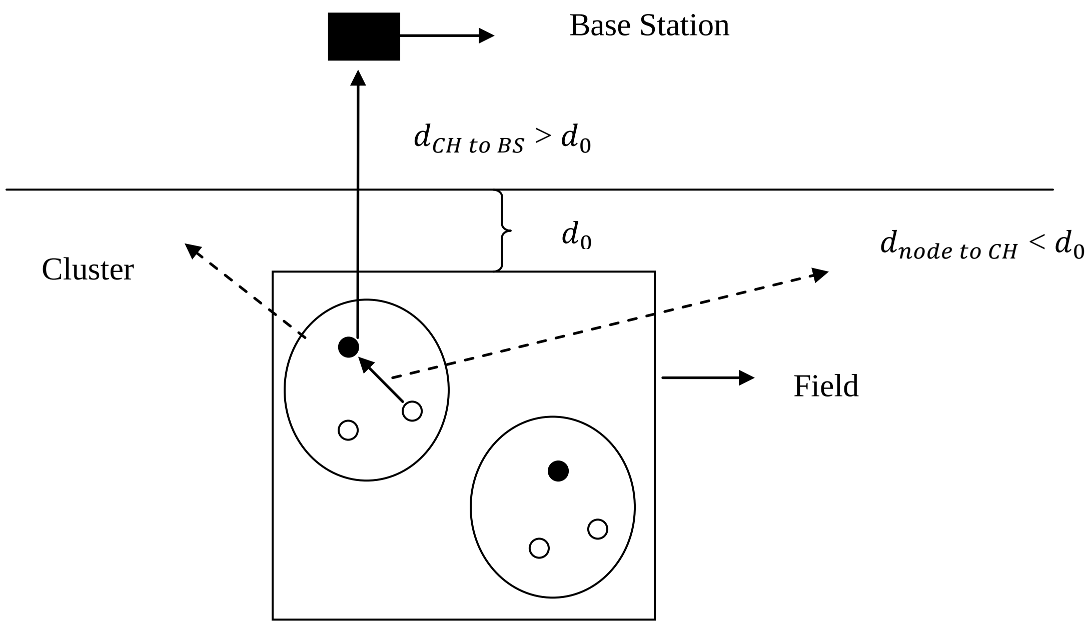


Figure 3.1: Field model when base station located far from the field

To find out which areas are considered far or close first we should calculate the value of d_0. The following equation is used for obtaining the d_0 value [46]:

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \qquad (3.3)$$

In formula (3.3) ϵ_{mp} *and* ϵ_{fs} are the energy dissipation values to run the amplifier for far and close distances respectively.

Consequently, situations in which base station located in a distance less than the threshold d_0 the communications between cluster heads and base stations as well as communications between nodes and cluster heads follow the free space model. Figure (3.2) presents a model in which base station located close to the field.

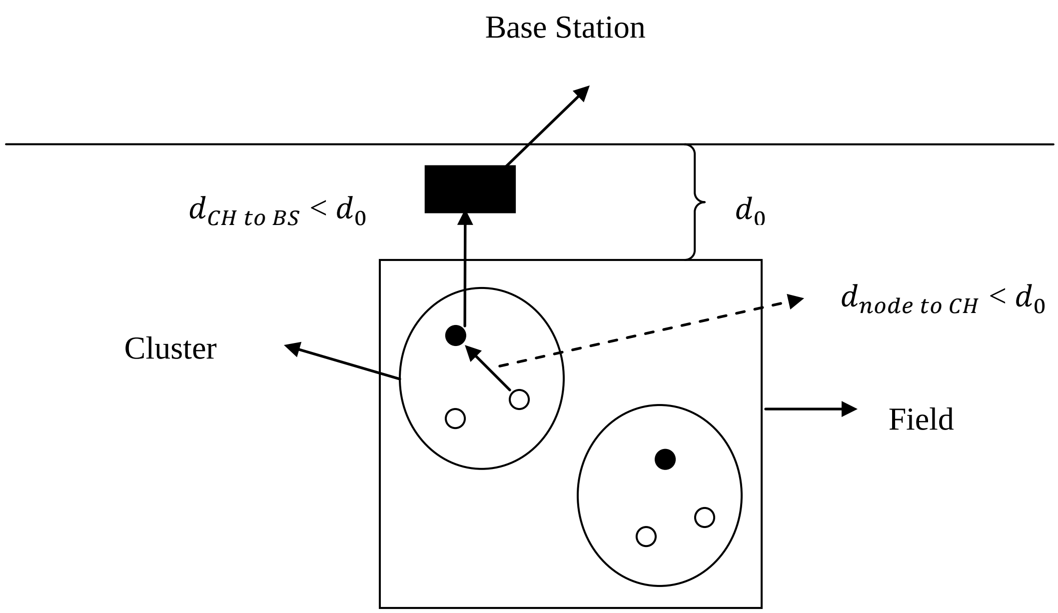


Figure 3.2: Field model when base station located close to the field

3.1.2 Derivation

3.1.2.1 Base station is located far from the field

LEACH calculates the optimum number of cluster heads when the base station located far from the field according to formula (2.3) described in chapter 2, section 2.2.1.

In formula (2.3), the optimal number of clusters has an inverse relation with $d^2{}_{toBS}$ which is distance of the nodes to the base station. This suggests that the further away the base station is from the network, the less number of clusters are desired. Less number of clusters in a region translates into larger clusters and more nodes inside a cluster. In other word, the closer the base station is to the network, the more clusters are desired. More number of clusters in a region translates into smaller clusters and less nodes inside the clusters. Therefore, In this work we focus on cluster size in proportion to the distance to the base station. Figure (3.3) shows the relation between distance to base station and the cluster size [47].

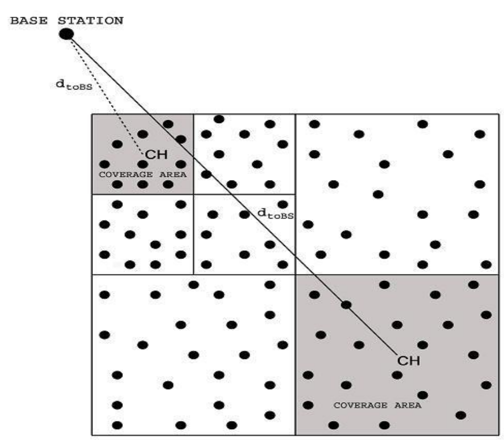


Figure 3.3: Proposed clustering model [47]

Figure (3.3) presents the clustering model in our proposed algorithm. The areas close to the base station contains more clusters, while the area far from the base station consists of less number of clusters.

Figure (3.4) shows better understanding of the relation between number of cluster heads and cluster size. Fields A and B are the same size. Black nodes show cluster heads while the white nodes represent non-cluster head nodes. In Field A, four cluster heads has been selected (four clusters) while in Field B two cluster heads has been created. Accordingly, clusters in Field A are smaller in size in compare to clusters in Field B. Consequently, the size of the clusters has an inverse relation to number of cluster heads (clusters).

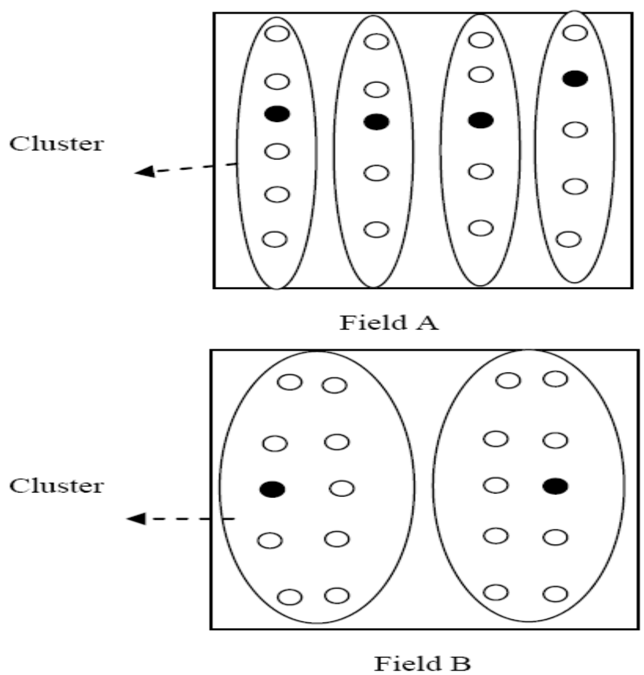


Figure 3.4: The inverse relation between the number and size of clusters

In our experiments, we consider a field of (100m*100m) with N=100 nodes scattered in it. In Figure (3.5), there are two nodes $n1$ and $n2$ in a field. Node $n1$ is located in the Cluster A with the area of S_A while node n_2 is located in the cluster B with area of S_B.

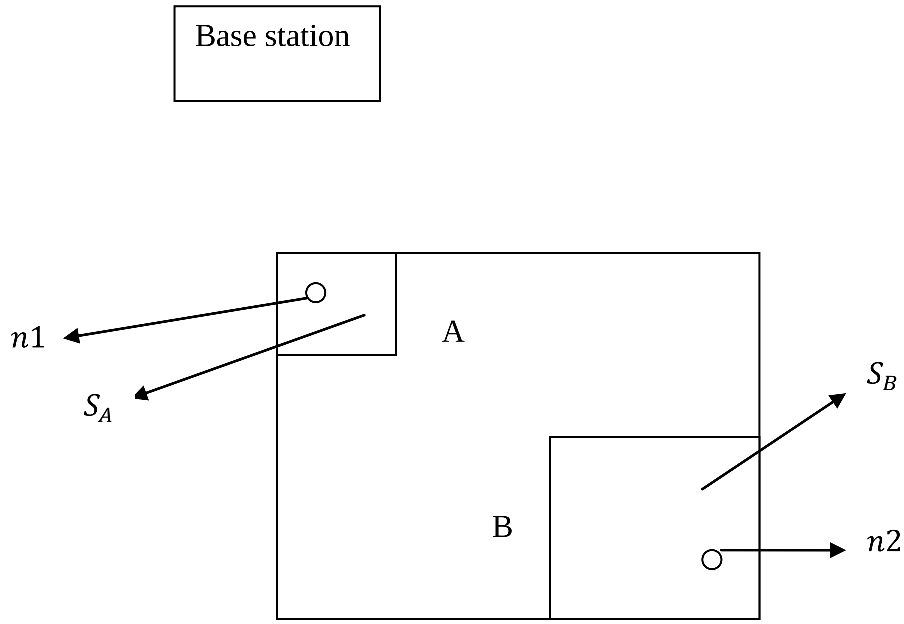


Figure 3.5: An image of the field with two nodes in it.

By using the Geometric Probability, the probability of selecting the node $n1$ in the area of A and $n2$ in the area of B are P_{n1} and P_{n2} as follows:

$$P_{n1} = \frac{S_{n1}}{S_A} \qquad (3.4)$$

S_A Is the area of A and S_{n1} is the area taken by node n1 (size of the node)

$$P_{n2} = \frac{S_{n2}}{S_B} \qquad (3.5)$$

S_B Is the area of B and S_{n2} is the area taken by node n2 (size of the node)

Based on assumption viii, explained in section 3.1.1, the nodes are the same in size thus the areas taken by nodes are equal:

$$S_{n1}=S_{n2} \tag{3.6}$$

Therefore, the ratio of the probability P_{n1} to the probability P_{n2} is as follows:

$$\frac{P_{n1}}{P_{n2}} = \frac{S_B}{S_A} \tag{3.7}$$

According to [39], the area used by each cluster is $\frac{M^2}{K}$ where k is defined as number of cluster heads obtained from formula (2.3) and we have explained it in chapter 2, section 2.2.1. Therefore the equation (3.7) becomes as follows:

$$\frac{P_{n1}}{P_{n2}} = \frac{\frac{M^2}{K_{n2}}}{\frac{M^2}{K_{n1}}} \tag{3.8}$$

Since M^2(area of the field) is constant, we have

$$\frac{P_{n1}}{P_{n2}} = \frac{K_{n1}}{K_{n2}} \tag{3.9}$$

K is obtained according to formula (2.3), explained in chapter 2, section 2.2.1. All the parameters in formula (2.3) are constant for all the nodes except the distance to the base station since distance of each node to the base station is different. Therefore:

$$\frac{P_{n1}}{P_{n2}} = \frac{\frac{1}{d^2{}_{n1\ to\ BS}}}{\frac{1}{d^2{}_{n2\ to\ BS}}} \quad \text{Or} \quad \frac{P_{n1}}{P_{n2}} = \frac{d^2{}_{n2\ to\ BS}}{d^2{}_{n1\ to\ BS}} \tag{3.10}$$

Formula (3.10) explains the reason we set the threshold based on distance during clustering phase. The probability of selecting the farthest node in a region to the base station is the lowest, while the probability of selecting the closest node to the base station is the highest. This means when $d_{n2\ to\ BS}$ is maximum P_{n2} is minimum. Consequently, fewer nodes would choose to perform as a cluster head in further areas. That results in a large-scale cluster. On the other hand, when $d_{n1\ to\ BS}$ is minimum the P_{n1} (probability of selecting $n1$ as a cluster head) is maximum which contributes to more cluster head selection, accordingly more clusters. Figure (3.6) shows an approximate image of nodes' probability in the field in regards to node's distance to the base station. For better understanding, the field is divided in to three regions: close, median and far to the base station. The nodes that are located closer to the base station are bigger in size, representing higher probability of cluster head selection versus the nodes shown as small dots representing lower probability since they are located farther from the base station.

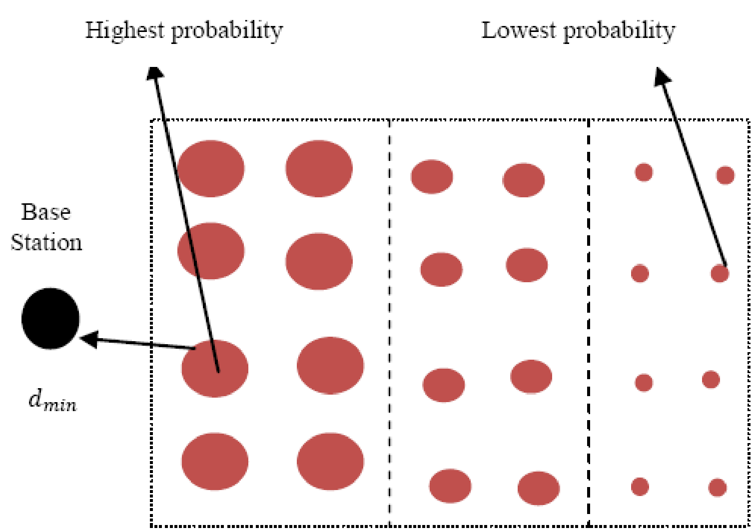


Figure 3.6: the size of the dots indicates the probability of each node being selected as a cluster head.

Consider a field which consist of three nodes A, B and C, based on formula (3.10), we have:

$$\frac{\mathrm{P_B}}{\mathrm{P_A}} = \frac{\mathrm{d^2}_{\text{A to BS}}}{\mathrm{d^2}_{\text{B to BS}}} \quad \text{Or} \quad P_B = P_A * \frac{\mathrm{d^2}_{\text{A to BS}}}{\mathrm{d^2}_{\text{B to BS}}} \tag{3.11}$$

$$\frac{\mathrm{P_C}}{\mathrm{P_A}} = \frac{\mathrm{d^2}_{\text{A to BS}}}{\mathrm{d^2}_{\text{C to BS}}} \quad \text{Or} \quad P_C = P_A * \frac{\mathrm{d^2}_{\text{A to BS}}}{\mathrm{d^2}_{\text{C to BS}}} \tag{3.12}$$

By formulas (3.11) and (3.12), we show that having the probability of one of the nodes (for example P_A), the probability of all other nodes in the field would be obtained (Since we know the location of each node the node's distance to the base station is also known). Next steps should be taken to calculate the probability of one of the nodes which lead to calculate all the nodes' probabilities in the network:

Same as LEACH [39], the probability of each sensor node i , P_i, is chosen in a way that the probability of the expected number of cluster heads is equal to the optimum number of cluster heads (K_{opt}). In other words:

$$\sum_{i=1}^{N} P_i = K_{opt} \quad \text{Which is} \quad P_1 + P_2 + \cdots + P_n = K_{opt} \tag{3.13}$$

By factoring P_1 from the equation we have:

$$P_1\left(1 + \frac{P_2}{P_1} + \cdots + \frac{P_n}{P_1}\right) = K_{opt} \tag{3.14}$$

Referring to formula (3.10), the ratio of the probabilities is known since it is defined as the distance of the nodes to base station. Therefore, all the parameters in the parenthesis of formula (3.14) are known. Consequently we can obtain P_1 which is:

$$P_1 = K_{opt} / \left(1 + \frac{P_2}{P_1} + \cdots + \frac{P_n}{P_1}\right) \tag{3.15}$$

Considering formulas (3.11) and (3.12), by having the value of P_A , we can calculate all other nodes' probabilities (P_B and P_C).

In this way, we analytically defined the probability threshold for each node in the network for being selected as a cluster head which is based on the distance of the node to the base station.

After assigning a probability to each node, each node selects a random number between 0 and 1 in each round and compares the selected random number to its assigned probability. If the random number is less than the assigned probability, the node would be selected as a cluster head for the next round. Otherwise it would be a non-cluster head.

3.1.2.2 Base station is located close to the field

When the distance of base station to the field is closer than a defined threshold d_0 as explained in section 3.1.1, the communication inside clusters and between cluster heads and base station follow a free space model. In this situation, the formula (2.3), explained in chapter 2, section 2.2.1, which calculates the optimum number of cluster heads is not applicable. The reason is that LEACH calculates formula (2.3) for the situation in which base station is located far from the field. To add more flexibility in our proposed method we calculated the optimum number of cluster heads when the base station is located close to the field.

LEACH assumes that there are N/k nodes per cluster. Therefore the energy dissipated in a cluster in each round is calculated as follows [39]:

$$E_{Cluster} = E_{CH} + \left(\frac{N}{K} - 1\right) E_{non-CH} \sim E_{CH} + \frac{N}{K} E_{non-CH} \qquad (3.16)$$

Where E_{CH} is the energy dissipation of cluster head, N is is the number of nodes, k is number of clusters.

The energy dissipation of each cluster head (E_{CH}) consists of the energy that each cluster heads dissipates in receiving signals from non-cluster head nodes, aggregation the signals and transmitting the aggregated data to the base station [39]. In this model since the base station located close to the field the energy dissipation for transmitting l bits of data follows the free space model. Therefore:

$$E_{CH} = lE_{elec}\left(\frac{N}{K} - 1\right) + lE_{DA}\left(\frac{N}{K}\right) + lE_{elec} + l\epsilon_{fs}d^2 \qquad (3.17)$$

Where l is the number of bits of a message, E_{elec} is the energy dissipation to run the radio electronics, $E_{DA}$is the energy for data aggregation and ϵ_{mp} is the energy dissipation to run the amplifier ($l\epsilon_{fs}d^2$ is the energy for sending the aggregated data to base station which follows the free space model described in section 3.1.1).

Therefore, the total energy dissipation in each round is calculated as follows:

$$E_{Total} = KE_{Cluster} \qquad (3.18)$$

Where K is number of clusters.

Considering the area occupied by each cluster a circle of $\frac{M^2}{K}$ [39] :

$$E_{Total} = l(NE_{elec} + NE_{DA} + K\epsilon_{fs}d^2 + NE_{elec} + N\epsilon_{fs}\frac{1}{2\pi}\frac{M^2}{K}) \qquad (3.19)$$

By setting the first derivative of a function to zero the maximum of the function is obtained. Therefore, in formula (3.19) the Optimum number of cluster heads is obtained by setting the derivative of E_{total} with respect to K to zero as follows:

$$\epsilon_{fs} d^2 = \frac{1}{K^2} \epsilon_{fs} \frac{M^2}{2\pi} N \qquad (3.20)$$

Accordingly, the formula for obtaining the optimum number of cluster heads when base station is close to the field calculated as follows:

$$K_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}} \frac{M}{d_{\ toBS}} \qquad (3.21)$$

Therefore, when the base station is close to the field finding the optimum number of cluster heads is based on formula (3.21). In chapter 4, section 4.4.1 , we calculated the optimum number by using simulations. Next step after obtaining the optimum number of cluster heads in our proposed method is assigning probability to each node which is exactly followed as the steps explained in section 3.1.2.1. The only difference is in formulas (3.9) and (3.10). As shown in formula (3.21), When the base station is located close to the field, the ratio of optimum number of cluster heads (K_{opt}) has an inverse relation to the distance of the nodes to base station ($d_{node\ to\ BS}$). Note that when the base station located far from the field the ratio of optimum number of cluster heads has an inverse relation to $d^2_{node\ to\ BS}$. Therefore, formula (3.10) is changed as follows:

$$\frac{\mathrm{P_{n1}}}{\mathrm{P_{n2}}} = \frac{\frac{1}{\mathrm{d_{n1toBS}}}}{\frac{1}{\mathrm{d_{n2toBS}}}} \quad \text{Or} \quad \frac{\mathrm{P_{n1}}}{\mathrm{P_{n2}}} = \frac{\mathrm{d_{n2toBS}}}{\mathrm{d_{n1toBS}}} \qquad (3.22)$$

Formula (3.22) describes the way we assign a weight to each node in the system. The probability of each node has an inverse relation to its distance to base station. Next steps for finding the value of the nodes’ probabilities are the same as what is described in section 3.1.2.1.

After assigning a probability to each node, nodes select a random number between 0 and 1 in each round and compare their selected random number to their probability. If the random number is less than the assigned probability the node would be selected as a cluster head for the next round. Otherwise it would be a non-cluster head node.

Cluster Formation: Once the cluster heads for the round r are selected, each cluster head node broadcasts an advertisement message to all the nodes in the field. Next, all non-cluster head nodes select their cluster head node according to the received signal strength of the advertisement message they have received from all the cluster heads of the current round. In other words, nodes select their cluster heads which require the minimum amount of transmitting energy. Once the nodes selected their cluster heads, they send a join request message to their selected cluster head nodes. This message consists of node's ID as well as the cluster head node's ID. For avoiding collision, similar to LEACH in our proposed algorithm, the cluster head nodes create a TDMA schedule, described in chapter 2 section 2.1.3, and send to their member nodes. The other advantage of the TDMA schedule is that all the member nodes can turn off their transceivers in order to reduce energy dissipation of each node. After cluster head selection the data transmission begins.

Data Transmissions: After cluster formation and TDMA creation the last phase of the algorithm, data transmission, starts. In this phase all the member nodes turn off their radio and just turn it on during their allocated transmission time. This act minimizes the energy consumption in each node. However, the cluster heads should keep their receiver on all the time during data transmission phase since member nodes are transferring data to them. Note that same as LEACH nodes only transmit data to their cluster heads. Based on assumption iii, explained in section 3.1.1, all the sensors can transmit to and receive data from the base station. Therefore, death of the nodes would not cause any hotspot in the system.

3.1.3 LEACH and WEEC Algorithms

Algorithm 3.1 describes the steps taken in the LEACH protocol Note that lines 6 to 11 are cluster head selection phase. On line 8 nodes select a random number and compare it to the threshold explained in Chapter 2, section 2.2.1. If the selected random number is less than a defined threshold the node is selected as a cluster head for that round. The energy consumption calculation for both cluster heads and non-cluster heads are done in lines 12 and 19 respectively. Line 23 ensures the rounds that no cluster head have been selected should not be counted since no transmission occurred. Lines 3 to 5 are responsible for storing number of dead nodes in the algorithm. While the nodes' energy is less than zero the node is considered dead. On line 18, nodes select their cluster head they belong. As explained earlier, nodes select their cluster heads based on the received signal strength. In the simulation, nodes select their cluster heads based on their distance to the cluster heads. In other words, nodes calculate their distance values to all the cluster heads and select the minimum value.

Algorithm 3.2 describes the steps taken in the WEEC protocol. Note that lines 2 to 4 are responsible to calculate and assign a probability to each node based on formulas (3.10) and (3.22) for far and close distances respectively. The assigned probabilities will be used as a threshold on line 12. The energy consumption calculation for both cluster heads and non-cluster heads are done in lines 15 and 22 respectively. Same as LEACH, only those rounds would be counted that the transmission occurs (line 26 ensures this).

Algorithm 3.1: The LEACH algorithm

```
   // the number of rounds
1  r = 0
2  while nodesAlive do
3      for i = 1 to N do
4          checkNumDeadNodes(arrNodes[i].E < 0)
5      end
       // loop to select clusterheads
6      for i = 1 to N do
7          if arrNodes[i].E > 0 then
8              randomNumber = rand()
9              if randomNumber < Threshold(r,k) then
                   // node i selected as clusterhead
10                 arrNodes[i].Type = ‘Clusterhead’
                   // increase the number of clusterheads
11                 numClusterHeads++
12                 calcDissipationCH()
13             end
14         end
15     end
       // computations for nodes not selected as clusterheads
16     for i = 1 to N do
17         if arrNodes[i].Type != ‘Clusterhead’ AND arrNodes[i].E > 0 then
18             assign nodes to clusters
19             calcDissipationNodes()
20         end
21     end
       // increment number of rounds only if clusterhead selected
22     if numClusterHeads > 0 then
23         r = r + 1
24     end
25 end
```

Algorithm 3.2 The WEEC algorithm

```
// the number of rounds
r = 0
for i = 1 to N do
    P(i) = f(d)
end
while nodesAlive do
    for i = 1 to N do
        checkNumDeadNodes(arrNodes[i].E < 0)
    end
    // loop to select clusterheads
    for i = 1 to N do
        if arrNodes[i].E > 0 then
            randomNumber = rand()
            if randomNumber < P(i) then
                // node i selected as clusterhead
                arrNodes[i].Type = 'Clusterhead'
                // increase the number of clusterheads
                numClusterHeads++
                calcDissipationCH()
            end
        end
    end
    // computations for nodes not selected as clusterheads
    for i = 1 to N do
        if arrNodes[i].Type != 'Clusterhead' AND arrNodes[i].E > 0 then
            assign nodes to clusters
            calcDissipationNodes()
        end
    end
    // increment number of rounds only if clusterhead selected
    if numClusterHeads > 0 then
        r = r + 1
    end
end
```

3.1.4 Comparison between LEACH and WEEC

Table 3.1 shows the steps taken in each round of LEACH and WEEC when the base station located far from the field. Gray area shows the difference between two algorithms while the white areas are common in LEACH and WEEC.

Table 3.1 Different step in LEACH and WEEC when the base station located far from the field

Steps	**LEACH**	**WEEC**
Computation of optimum number of cluster heads (K_{opt})	$K_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}}\sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}}\frac{M}{d^2{}_{toBS}}$	$K_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}}\sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}}\frac{\boldsymbol{M}}{\boldsymbol{d^2{}_{toBS}}}$
Cluster head selection	**a.** Defines a threshold that resets in every N/K rounds: T(t)= $\frac{K}{N-K*(r \bmod \frac{N}{k})}$ **b.** Nodes select random number between 0 and 1 and Compare it to T(t).	**a.** Calculates a probability for each node base on distance of each node to base station **b.** Nodes select random number between 0 and 1 and compare it to their assigned probability values.
Cluster Formation	Based on the received signal strength of the advertisement message sent by CHs nodes choose their clusters	Based on the received signal strength of the advertisement message sent by CHs nodes choose their clusters
Data Transmission	Based on TDMA schedule nodes send data to their CHs afterwards CHs send the aggregated data to BS.	Based on TDMA schedule nodes send data to their CHs afterwards CHs send the aggregated data to BS.

As shown in Table 3.1 the main difference between LEACH and WEEC is during the cluster head selection. As Described in formula (2.2) in chapter 2, section 2.2.1, LEACH guarantees that each node should be selected as cluster head in every $\frac{N}{K}$ round. To do this it defines a threshold T (t). Note that T (t) has a relation with number of round (r) and thus the value of it changes each round. For the sake of simplicity, we re-present formula (2.2) here:

$$T_i(t)= \begin{cases} \frac{K}{N-K*(r \bmod \frac{N}{k})} & C_i(t)=1 \\ 0 & C_i(t)=0 \end{cases} \qquad (3.23)$$

Where k is the optimum number of cluster heads, N is number of nodes, r is number of round and i indicates the nodes. $C_i(t)$ is the indicator function to determine if a node has been selected as a cluster head during the $(r \bmod \frac{N}{k})$ rounds or not. For example if node i has been selected in the last $(r \bmod \frac{N}{k})$ rounds the value of $C_i(t)$ sets to zero.

The threshold defined in LEACH increases in every round. The increased threshold also increases the likelihood of each node to be selected as a cluster head. The following example illustrates the pattern of increasing threshold values for an experiment with nineteen rounds. Only the values of certain number of rounds are shown as they are enough to recognize the pattern of increasing threshold. Note that in the following calculation N=100 and K sets to 5.

r=1 T(1)= 0.052

r=2 T(2)= 0.055

r=10 T(10)= 0.1

r=18 T(18)= 0.5

r=19 T(19)=1

In this approach, during the first rounds, since the threshold of a node being selected as a cluster head is too low, fewer nodes would be selected as cluster heads when compared to the number of nodes being selected as cluster heads in last rounds. The reason that a greater number of nodes are selected as cluster heads in the last few rounds is because the threshold is nearly one. As mentioned in chapter 2, section 2.2.1, LEACH sets the optimum number of cluster heads to 5 when base station located far from the field. Ensuring that the number of cluster heads in each round is close to the optimum number of cluster heads would result in efficient energy consumption [6]. The derivation from the optimum number of cluster heads in the LEACH protocol suggested that the energy efficiency of LEACH could be improved by reducing the deviations from the optimum number of cluster heads. Moreover, further study of the LEACH protocol began with the analysis of the equation (3.24) which defines the energy consumption of cluster heads in the LEACH protocol [6]:

$$E_{CH} = lE_{elec}\left(\frac{N}{K} - 1\right) + lE_{\ DA}\frac{N}{K} + lE_{elec} + l\in_{mp{d^4}_{toBS}} \qquad (3.24)$$

Ignoring the first order terms:

$$\mathrm{E_{CH}} \sim \mathrm{l} \in_{\mathrm{mp{d^4}_{toBS}}} \qquad (3.25)$$

Formula (3.25) indicates that energy consumption by each cluster head is proportional to its distance to base station raised to the power of four. Therefore distance of the cluster heads to base station has a huge impact on energy consumption that could not be ignored in clustering.

3.2 Visualization Algorithms

Following the simulation of either the LEACH or WEEC protocol, data is generated pertaining to the energy levels of each node *i* for *j* rounds. The location of each node *i* is also recorded and stored in separate files. Algorithm (3.3) describes the steps to visualize the energy and location data of WSN nodes deployed in the field.

Each round of simulation is visually represented as a single keyframe in the animation timeline. A keyframe is a frame of animation where values for the height and colour of cubes are marked. The marked values are interpolated between keyframes to produce a smooth animation. The purpose of the `animFramesPerRound` variable is to set the number of frames between rounds or, between keyframes.

The X- and Y-coordinates for node *i* are read from a file on lines 6 and 7 and stored in the arrays $\mathtt{cube}_X$ and $\mathtt{cube}_Y$, respectively. The energy value for node *i* is read from a file and stored in the variable `nodeEnergy` on line 8.

The function `getColour` maps the energy value stored in the variable `nodeEnergy` to a colour value between $\mathtt{colour}_{min}$ and $\mathtt{colour}_{max}$. The colour is stored in HSV format from red to green representing $\mathtt{colour}_{min}$ and $\mathtt{colour}_{max}$ respectively. The function `getHeight` maps the energy value stored in the variable `nodeEnergy` to a height value between $\mathtt{height}_{min}$ and $\mathtt{height}_{max}$. The values returned by the `getColour` and `getHeight` functions are stored in the variables $\mathtt{cube}_c$ and $\mathtt{cube}_h$, respectively.

A few of the parameters for the cube function such as $\mathtt{cube}_X$, $\mathtt{cube}_Y$, $\mathtt{cube}_c$ and $\mathtt{cube}_h$ were defined previously and the remaining parameters are defined as the following: `arrCubes` is an array of *N* cube identifiers; $\mathtt{cube}_w$ is the width of the cube; and, $\mathtt{cube}_d$ is the depth of the cube.

The `cube` function has 2 modes of operation. It either creates a new cube in the scene or it modifies an existing cube in the scene.

If the cube identifier in the array `arrCubes` at index *i* is empty, the `cube` function creates a new cube and returns its identifier. However, if the cube identifier in the array `arrCubes` at index *i* is a non-empty value, the `cube` function modifies the existing cube with that identifier and returns the same identifier value.

Initially, no cubes exist in the animation. Therefore, in the first iteration of the for-loop between lines 5 and 13, the `cube` function creates *N* cubes. During subsequent iterations of the same for-loop, values for the height and colour are modified by the `cube` function for each existing cube.

The `setKeyFrame` function marks the colour and height values at the value stored in the `frame` variable for the cube identifier at index *i* of the `arrCubes` array.

Once height and colour values are marked for all cubes, the `frame` value is incremented to the next keyframe and the values for the next round of simulation are read and marked as the process is repeated for *R* rounds of simulated data.

Algorithm 3.3: The algorithm for visualizing the energy and location data associated with each WSN node in the field.

```
inputs  : Node energy and location data
outputs: R keyframes of animation with N cubes; heights and colours of cubes
         mapped to energy levels of nodes; locations of cubes mapped to
         locations of nodes

// empty array of N cube identifiers
arrCubes[N]
// begin at frame 0
frame = 0
// number of frames to skip between rounds or keyframes
animFramesPerRound = 12
for j = 1 to R do
    for i = 1 to N do
        cube_X[i] = readNodeLocationX(i)
        cube_Y[i] = readNodeLocationY(i)
        nodeEnergy = readNodeEnergyValue(i)
        // map energy value to height and colour of cube
        cube_c = getColour(nodeEnergy,colour_min,colour_max)
        cube_h = getHeight(nodeEnergy,height_min,height_max)
        arrCubes[i] =
        cube(arrCubes[i],cube_w,cube_d,cube_h,cube_c,cube_X[i],cube_Y[i])
        setKeyFrame(arrCubes[i],cube_c,cube_h,frame)
    end
    frame = frame + animFramesPerRound
end
```

Another component of the visualization of node energy and location data is the display of textual information in the viewing window. The display of textual information also includes the current round number and the process is described in Algorithm 3.4.

The animated display of textual information is achieved through a classical technique where one object is drawn on a single blank frame and then redrawn on a subsequent blank frame with minor changes. When the process is repeated for several frames and the frames are played in sequence, the illusion of animation is created.

Algorithm 3.3 executes only once and creates cubes representing the energy levels of nodes, the heights and colours of which, vary over a certain number of frames. In contrast, Algorithm 3.4 is evaluated once for every frame in the animated sequence.

The if-statements on lines 4-6, 7-9 and 10-13 of Algorithm 3.4 evaluate the existence of any text objects for the node energy level, node location and the current round number, respectively. If any such objects exist, they are deleted. This creates a blank frame with only cubes and no textual information.

The node energy and location values are obtained on lines 13 and 14, respectively, using the `getEnergyValue` and `getLocationValue` functions. In the same order, text objects are created for the node energy and location values on lines 16 and 17 using the `createTextObject` function. The current round number is obtained with the expression on line 15 and is displayed by calling the `createTextObject` function on line 18.

To allow for camera movement while maintaining text alignment to the camera, the calls to the function `positionTextObject` on lines 19-21 position text objects based on the current translation and rotation of the camera and also the translation and rotation of cubes in the 3D scene. Note that while the node location text is static in that the coordinates and the node IDs do not change, to allow dynamic alignment to a moving camera, the node location text, too, must be recreated in every frame.

Algorithm 3.4: The algorithm for displaying textual information related to the current node energy level and node location values along with the current round number.

```
inputs : An index value i for the array of cube identifiers arrCubes; an
         integer value for the current frame
outputs: New text objects that replace old text objects for node energy, node
         location and current round in the 3D scene

// the cube identifier for node i
arrCubes[i]
// always contains the current frame value
currentFrame
// number of frames to skip between rounds or keyframes
animFramesPerRound = 12
// if a text object for energy exists for node i, delete it
if exist(energyTextObj[i]) then
    delete(energyTextObj[i])
end
// if a text object for location exists for node i, delete it
if exist(locationTextObj[i]) then
    delete(locationTextObj[i])
end
// if a text object for round number exists, delete it
if exist(roundNumTextObj) then
    delete(roundNumTextObj)
end
nodeEnergy = getEnergyValue(arrCubes[i])
nodeLocation = getLocationValue(arrCubes[i])
// get current round number
currentRound = currentFrame % animFramesPerRound
// create new text objects for energy, location and round number
energyTextObj[i] = createTextObj(nodeEnergy)
locationTextObj[i] = createTextObj(nodeLocation)
roundNumTextObj = createTextObj(currentRound)
// position text; account for camera
// and cube translation and rotation
positionTextObj(energyTextObj[i])
positionTextObj(locationTextObj[i])
positionTextObj(roundNumTextObj)
```

Algorithm 3.5 describes the steps to visualize the cluster formation in each round. Note that all the steps are the same as Algorithm 3.3 except the colour assignment steps. On line 10 the black colour for cluster head is assigned and on line 13 the colour of the non-cluster head nodes would be defined.

Algorithm 3.5: The algorithm for displaying the cluster formation in each round

---

inputs : Node energy, location and cluster data
outputs: R keyframes of animation with N cubes; heights of cubes mapped to energy levels of nodes; locations of cubes mapped to locations of nodes; node clusters are grouped by colour; black cubes indicate cluster heads

// empty array of N cube identifiers
1 $arrCubes[N]$
// begin at frame 0
2 $frame = 0$
// number of frames to skip between rounds or keyframes
3 $animFramesPerRound = 12$
4 **for** $j = 1$ **to** R **do**
5   **for** $i = 1$ **to** N **do**
6         $cube_X[i] =$ `readNodeLocationX`(i)
7         $cube_Y[i] =$ `readNodeLocationY`(i)
8         $nodeEnergy =$ `readNodeEnergyValue`(i)
9         $nodeCluster =$ `readNodeClusterValue`(i)
        // map cluster value to colour of cube
        // a negative value indicates the node is
        // a cluster head
10   **if** $nodeCluster == -1$ **then**
11             $cube_c =$ black
12   **else**
13             $cube_c =$ `getColour`($nodeCluster, cluster_{min}, cluster_{max}$)
14   **end**
        // map energy value to height of cube
15         $cube_h =$ `getHeight`($nodeEnergy, height_{min}, height_{max}$)
16         $arrCubes[i] =$
        `cube`($arrCubes[i], cube_w, cube_d, cube_h, cube_c, cube_X[i], cube_Y[i]$)
17   `setKeyFrame`($arrCubes[i], cube_c, cube_h, frame$)
18   **end**
19     $frame = frame + animFramesPerRound$
20 **end**

3.3 Summary

In this chapter a weighted clustering method for energy efficient data transfer in wireless sensor networks is introduced. In addition, it is shown that the proposed algorithm is expected to perform more efficient by assigning a probability value for each node during their cluster head selection stage. The visualization algorithm which reads energy level and location of each node and visualizes the residual energy of each node by colour and height is also explained.

In the next chapter, the proposed algorithm will be evaluated and demonstrated. Moreover, the proposed algorithm will be compared to LEACH in terms of network lifetime and the number of cluster heads selected in each round.

CHAPTER 4

IMPLEMENTATION AND EVALUATION

In this chapter two hierarchical routing protocols are compared. The first hierarchical routing protocol is the LEACH protocol which is described by Heinzelman et al. in [6,39]. The second hierarchical routing protocol is established in this work and known as WEEC protocol [7]. Simulation experiments are carried out in Matlab (2010b). Four experiments are performed. The simulation parameters for each experiment are provided in Table 4.1. Because the x and y coordinates are randomly chosen for each node, the four experiments are repeated 10 times to minimize the influence of random deployment. At the end, the averaged results from LEACH and WEEC are used for comparison.

Section 4.1 describes the simulation parameters. Sections 4.2 and 4.3 represent the experiment metrics as well as the experimental results respectively. The visualization is shown in section 4.4. Finally, Section 4.5 summarizes and concludes the results.

4.1 Simulation Environment

In our WSN, there are 100 homogenous sensors which are randomly deployed in a field of (100m*100m). Table 4.1 presents the simulation parameters and settings that have been done in all the experiments of our proposed method as well as LEACH.

Table 4.1: Simulation parameters.

Parameter	**Experiment 1**	**Experiment 2**	**Experiment 3**	**Experiment 4**
Size of the area	100m *100m	100m *100m	100m *100m	100m *100m
BS location	50m,175m	50m,175m	50m,50m	50m,50m
Number of nodes	100	100	100	100
Nodes initial energy	0.5J	0.5J	0.25J	0.25J
E_{elec}	50 nJ/bit	50 nJ/bit	50 nJ/bit	50 nJ/bit
ϵ_{fs}	10pJ/bit/m^2	10pJ/bit/m^2	10pJ/bit/m^2	10pJ/bit/m^2
ϵ_{mp}	0.0013pJ/ bit/m^4	0.0013pJ/ bit/m^4	0.0013pJ/bit/m^4	0.0013pJ/ bit/m^4
E_{DA}	5 nJ/bit/signal	5 nJ/bit/signal	5nJ/bit/signal	5 nJ/bit/signal
Data Packet size	2000 bits	2000 bits	2000 bits	2000 bits

Each simulation parameter is explained in chapter 2, section 2.2.1. Note that the base station is located inside the field for the experiments 3 and 4 which means that the base station is close to the field.

4.2 Experiment metrics

As described in chapter 2, section 2.2.3, Kim and Youn in [41] study the influence of the initial energy of each node. We performed the same experiment to evaluate the influence of the initial energy of each node in the LEACH and WEEC protocols. As in the work by Kim and Young, the initial energy values in our experiments are set to 0.25J and 0.5J. For analyzing and comparing the performance of our proposed method we used two metrics: network lifetime and the number of cluster heads created in each round.

One of the most important factors for evaluating the sensor network is network lifetime. The lifetime of the network depends on the lifetime of each sensor node that is a part of that network. Recharging and replacing the nodes' batteries is impractical in many environments. Therefore, the literature is rich with different algorithms aiming to prolong network lifetime. Various lifetime metrics have been developed [48]. In our experiment we define the network lifetime as the time until all the nodes are dead. We define a network lifetime based on the death of last node since we could still have some feedback about the situation of the field [53].

The importance of the optimum number of cluster heads is because when there are fewer cluster heads in the field than the optimum some nodes have dissipate more energy to send their data since their cluster heads are located far from them. When there are more cluster heads than the optimum, more cluster heads have to send data to the long-haul distances to the base station [6].

4.3 Experiments and Results for Base Station Located at (50m, 175m)

4.3.1 Experiment 1

In this experiment the base station is located far from the field with nodes' initial energy of 0.5 J. As mentioned earlier in chapter 3, section 3.1.1, when the base station is located far from the field the communications inside each cluster follow the free space model. Communications outside the clusters are between cluster heads and the base station and follow the multipath model. Figure 4.1 compares the network lifetimes of the LEACH and WEEC protocols.

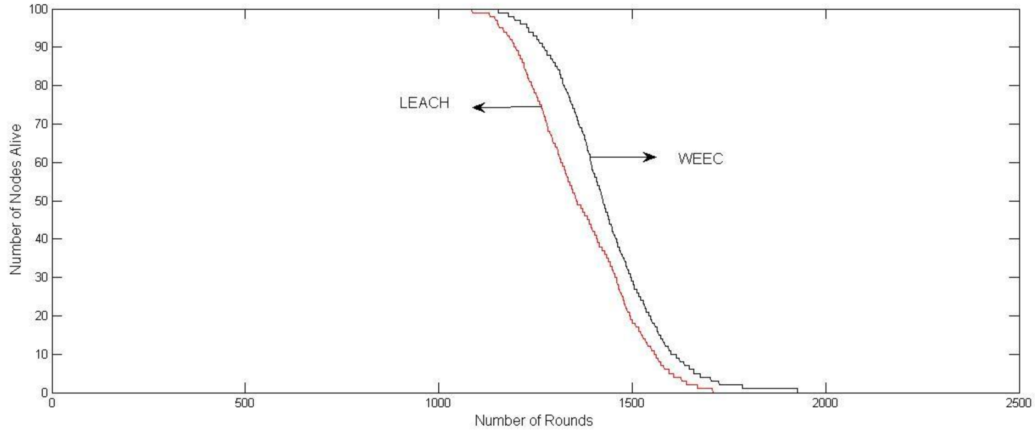


Figure 4.1: The network lifetime comparison between LEACH and WEEC when the base station is located far from the field with nodes' initial energy of 0.5J

The number of rounds before the first node dies in the LEACH protocol and the WEEC protocol is 1087 and 1155 respectively. Nodes begin to die more quickly in the LEACH protocol than the WEEC protocol. The number of rounds before the last node dies in the network using the LEACH protocol is lower than the number of rounds before the last node dies in the network using the WEEC protocol. The number of rounds completed before the last node dies in LEACH and WEEC is 1708 and 1927.

Table 4.2: The rounds that nodes start dying and all the nodes are dead in LEACH and WEEC when base station is far from the field*

Protocol	Round when nodes start dying	Round when all nodes are dead
LEACH	1087	1708
WEEC	1155	1927

***The initial energy of all nodes is 0.5 J**

As explained in section 4.2, the network lifetime, which is defined as the time elapsed until the last node dies, is increased by 12%. The percentage differences in the network lifetimes are calculated using Equation (4.1):

$$Percentage\ Difference = \left|\frac{r_{WEEC} - r_{LEACH}}{r_{LEACH}}\right| \times 100 \qquad (4.1)$$

Where r_{WEEC} is the number of round that the last node dies in WEEC and r_{LEACH} is the number of round that the last node dies in the LEACH.

Figures 4.2 and 4.3 show the number of cluster heads selected in each round for the LEACH and WEEC protocols, respectively. The optimal number of cluster heads is set to five for both algorithms. Figure 4.2 shows that in some rounds LEACH selects up to twenty five cluster heads, which is far from the desired number of cluster heads. However, as shown in Figure (4.3), in the WEEC protocol, the maximum number of cluster heads that are selected in a round is six, which is closer to the desired number of cluster heads. In other rounds, the selected number of cluster heads in WEEC is generally close to five except the last few rounds, when the number of cluster heads declines to one. The reason for the decline in the number of cluster heads is that in the last rounds the majority of the nodes in the field are dead.

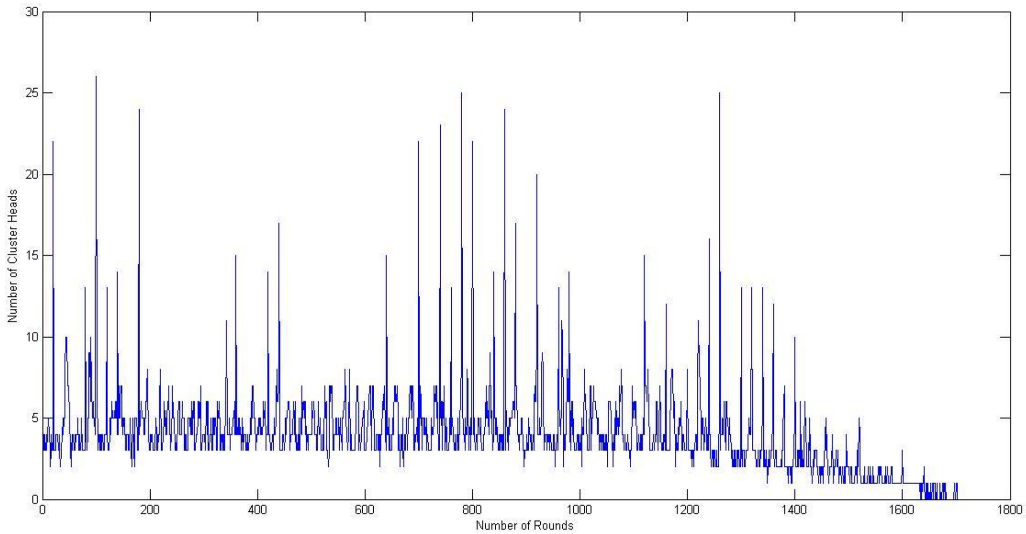


Figure 4.2: Number of cluster heads selected in each round of LEACH in experiment 1 when bas station is located far from the field with an initial energy of 0.5 J for each node.

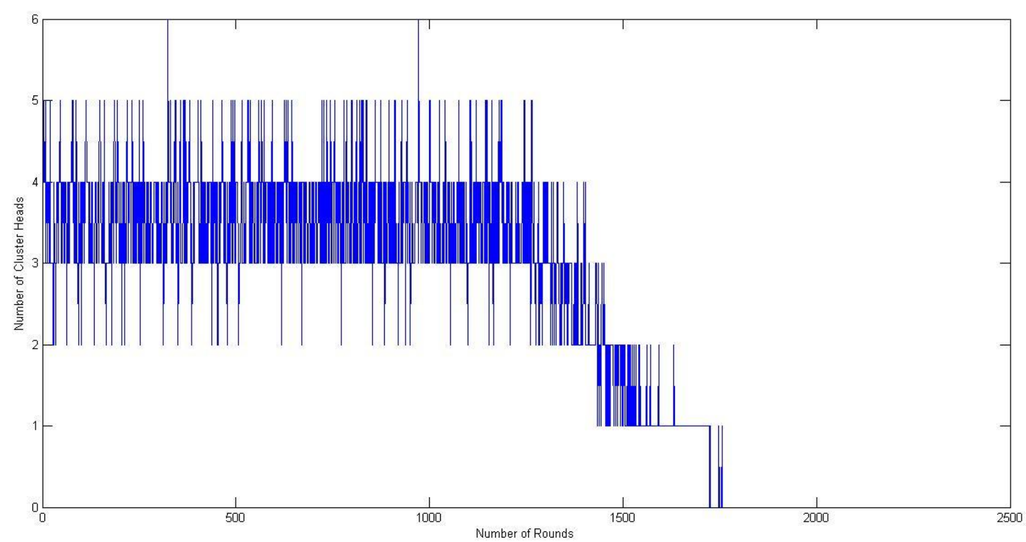


Figure 4.3: Number of cluster heads selected in each round of WEEC in experiment 1 when base station is located far from the field with an initial energy of 0.5J for each node.

According to Figures 4.2 and 4.3, the WEEC protocol is more efficient in clustering since the number of clusters created in each round is closer to the desired number of cluster heads.

4.3.2 Experiment 2

In this experiment, the base station is located far from the field with nodes' initial energy of 0.25 J. As presented in Table 4.1, we did our experiments with different initial energies for each node to study the influence of each node's initial energy on the network lifetime. Our intention was to ensure that the efficiency of WEEC in compare to LEACH does not depend on the initial energy of each node. Figure 4.4 presents the network's lifetime for the LEACH and WEEC algorithms when the initial energy of each node is 0.25 J and the base station is located far from the field at (50m, 175m).

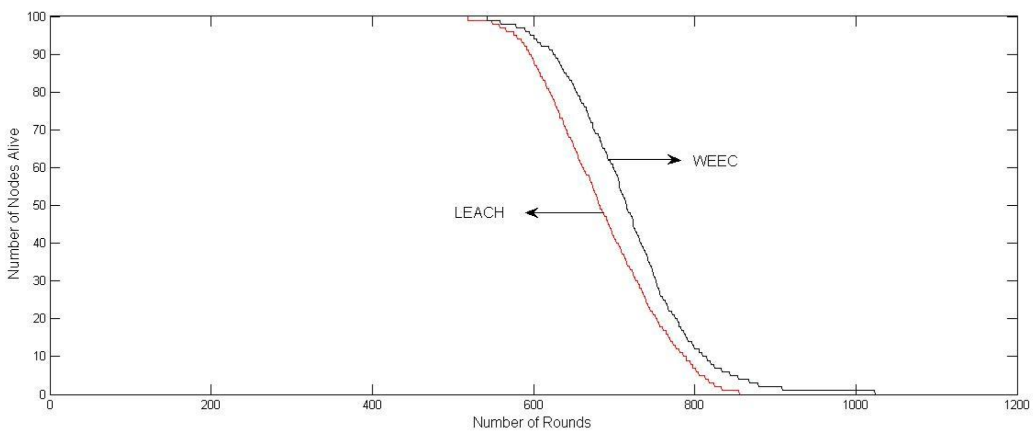


Figure 4.4: Comparison of the network's lifetime between LEACH and WEEC in experiment 2 when the base station located is far from the field with nodes' initial energy of 0.25 J.

Table 4.3 presents the number of rounds in which the first node starts dying as well as the round that the last node died in both algorithms. The number of rounds before the first node dies in the

LEACH protocol and the WEEC protocol is 519 and 543 respectively. Nodes begin to die more quickly in the LEACH protocol than the WEEC protocol. Note the last node dies at round 856 in the LEACH protocol but the round that the last node dies in WEEC is 1024. In this case, the network lifetime is increased by 19% based on Equation (4.1). That is, the total number of rounds elapsed in the WEEC protocol exceeds the total number of rounds elapsed in the LEACH protocol.

Table 4.3: The rounds that nodes start dying and all the nodes are dead in LEACH and WEEC when base station is far from the field*

Protocol	**Round when nodes start dying**	**Round when all nodes are dead**
LEACH	519	856
WEEC	543	1024

***The initial energy of all nodes is 0.25 J**

Figures 4.5 and 4.6 present the number of clusters created in each round in LEACH and WEEC respectively.

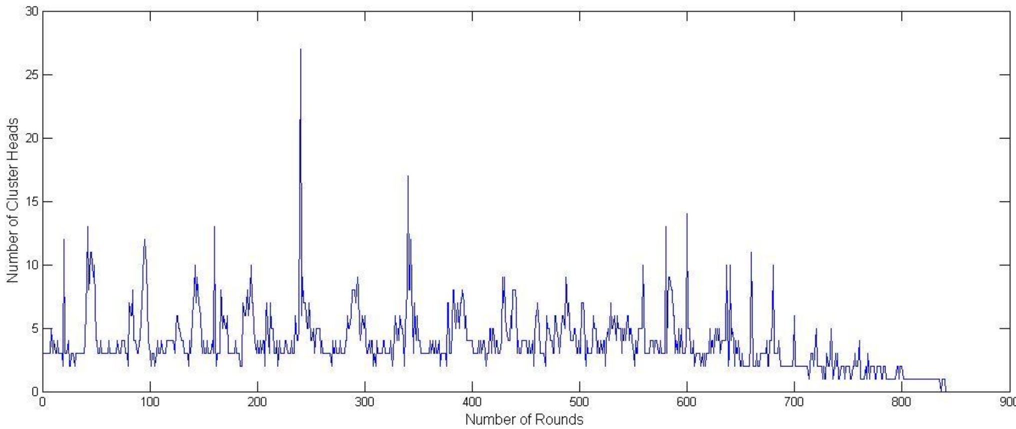


Figure 4.5: Number of cluster heads selected in each round of LEACH in experiment 2 when base station is located far from the field with an initial energy of 0.25 J for each node.

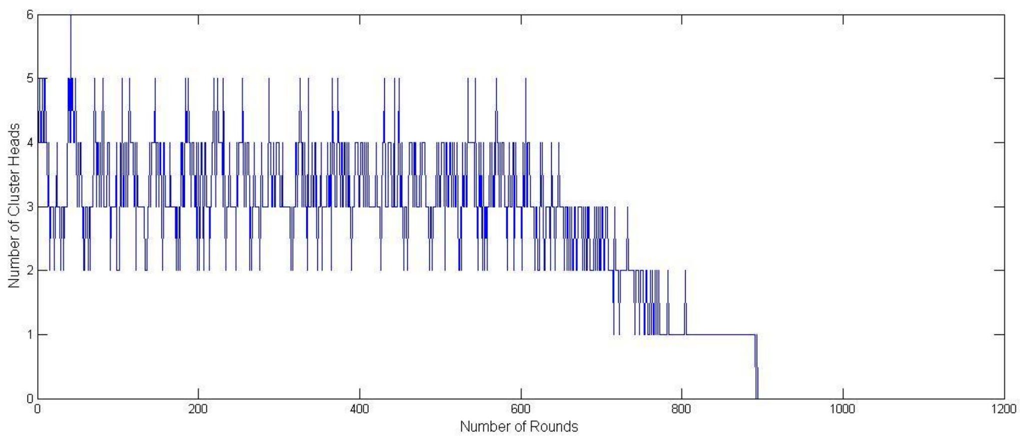


Figure 4.6: Number of cluster heads selected in each round of WEEC in experiment 2 when base station is located far from the field with an initial energy of 0.25 J for each node.

By comparing Figures 4.5 and 4.6, it may be noted that the number of clusters in LEACH fluctuates from 1 to 27, which is not desired as the optimal number of cluster heads computed prior to the simulation is set to 5 for both protocols. In contrast, the number of clusters in WEEC fluctuates between 1 and 6, which is closer to the desired number of cluster heads.

4.4 Experiments and Results for Base Station Located at (50m, 50m)

4.4.1 Obtaining the Optimum Number of Cluster Heads

The computation of the optimum number of cluster heads is modified, since the base station in experiments 3 and 4 is located inside the field. What follows is an explanation of how we calculated the optimum number of cluster heads utilizing the Free Space model.

As explained in chapter 3, section 3.1.2.2 the formula for finding the optimum number of cluster heads when the base station is close or inside the field is calculated as follows:

$$K_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}} \frac{M}{d_{toBS}} \qquad (4.2)$$

The optimum number of cluster heads when the base station is close or inside the field is calculated differently from the situation in which the base station is located far from the field.

We begin experiments 3 and 4 by first studying the significance of the optimum number of cluster heads in terms of network lifetime. To study this effect, the base station is situated at (50m, 50m) and the optimum number of cluster heads is set to 2, 5, 18 and 19. One hundred iterations of the LEACH algorithm are performed and the network lifetimes are compared in Figure 4.7.The impact of the optimum number of cluster heads is apparent as the network lifetime when the optimum number of cluster heads is set to 19 differs significantly from the network lifetime when the optimum number of cluster heads is set to 5 or 2.

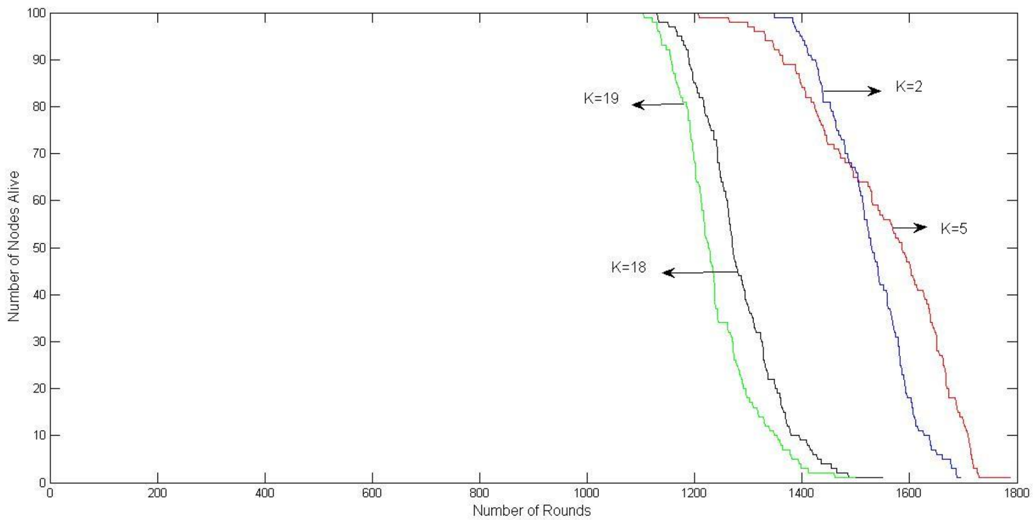


Figure 4.7: Comparing different values of optimum number of cluster heads in LEACH's network lifetime-In this figure base station is inside the field with nodes' initial energy of 0.5 J.

Since it is known that the number of cluster heads significantly affects the network lifetime, we repeat the previous iterations with a narrower range of 2 to 5 for the number of cluster heads. The purpose of repeating the previous iterations is to obtain the optimum number of cluster heads that maximizes the network lifetime when base station is close to the field. Figure 4.8 shows lifetime of the network utilizing LEACH where the optimum number of cluster heads is varied between 2 and 5. Note that the results in Figure 4.8 are averaged over 10 experiments. We reached the conclusion that the optimum number of cluster heads, when the base station is inside the field, is between 2 and 5. The network lifetime is maximized when the optimum number of cluster heads is set to four or, K_{opt}=4.

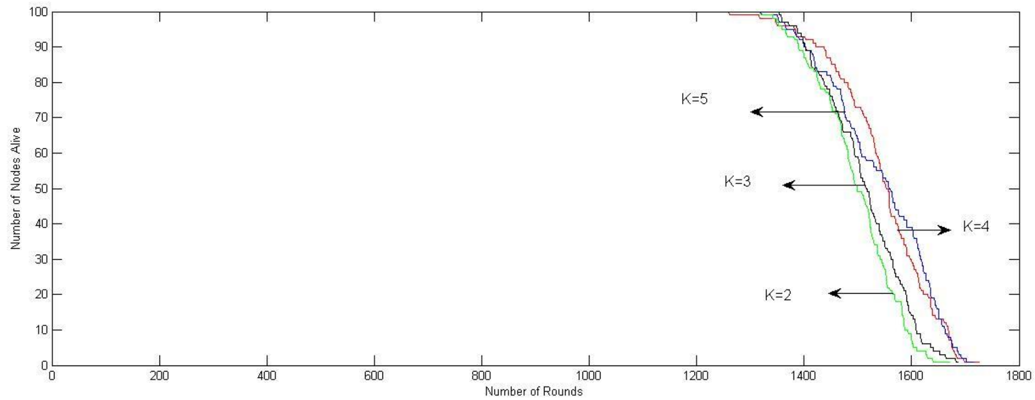


Figure 4.8: Calculating the optimum number of cluster heads for LEACH when base station is inside the field

Table 4.4: Presents the number of rounds that last node dies with different optimum number of cluster heads. Based on the last dead node criteria, it is apparent that the optimum number of cluster head is four.

Protocol	K_{opt}	Round last node dies
LEACH	*2*	1671
	3	1686
	4	1726
	5	1716

4.4.2 Experiment 3

In this experiment, the base station is located inside the field with nodes' initial energy of 0.5 J. The base station is located at (50m, 50m), where it is considered close.

By giving the entire node a specific probability based on their distance to the base station, we improve the network lifetime in WEEC when it is compared to LEACH. Figure 4.9 compares the network lifetimes of LEACH and WEEC.

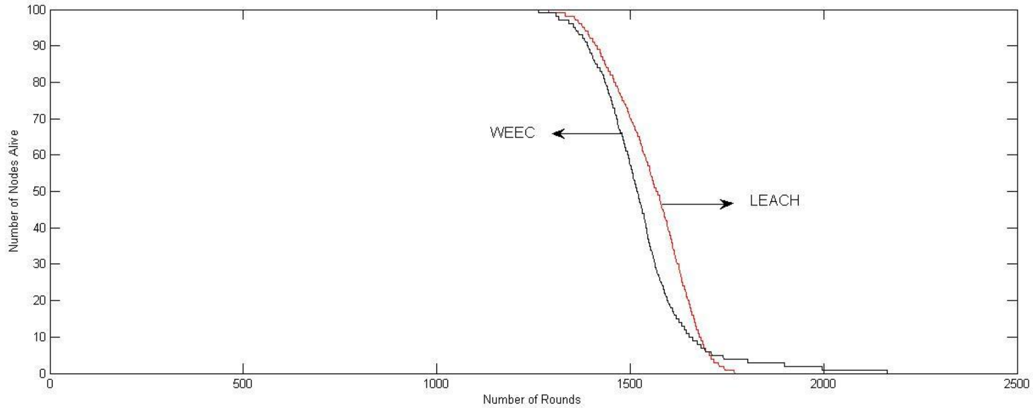


Figure 4.9: Comparison of the network's lifetime between LEACH and WEEC in experiment 3 when the base station is inside the field with nodes' initial energy of 0.5 J.

As shown in Table (4.5), the number of rounds before the first node dies in the LEACH protocol and the WEEC protocol is 1289 and 1263 respectively. Nodes begin to die more quickly in the WEEC protocol than the LEACH protocol. However, the number of rounds before the last node dies in the network using the LEACH protocol is lower than the number of rounds before the last node dies in the network using the WEEC protocol. The number of rounds completed before the last node dies in LEACH and WEEC is 1770 and 2165 respectively. Though the first node dies earlier in the case of the WEEC protocol than in the LEACH protocol, the network lifetime is increased by 22% based on Equation (4.1).

Table 4.5: The rounds that nodes start dying and all the nodes are dead in LEACH and WEEC when base station is located inside the field*

Protocol	Round when nodes start dying	Round when all nodes are dead
LEACH	1289	1770
WEEC	1263	2165

***The initial energy of all nodes is 0.5 J**

By comparing Figures 4.10 and 4.11, it may be noted that the number of clusters in LEACH fluctuates from 1 to 57, which is not desired as the optimal number of cluster heads computed prior to the simulation is set to 4. In contrast, the number of clusters in WEEC fluctuates between 1 and 9, which is closer to the desired number of cluster heads and has less fluctuation. This deviation from the optimum number of cluster heads in the case of LEACH is also observed in Figures 4.5 and 4.6.

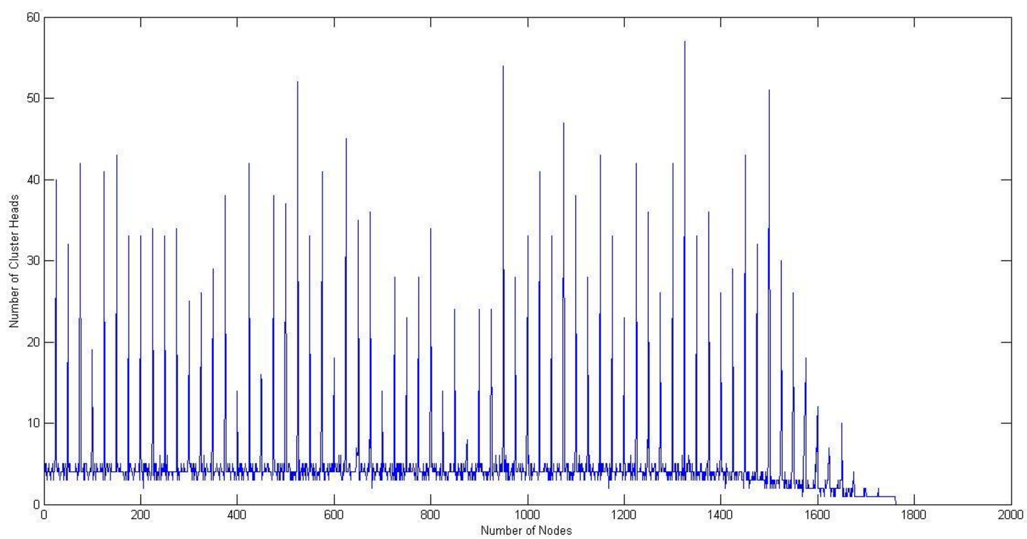


Figure 4.10: Number of cluster heads selected in each round of LEACH when the base station is inside the field with initial energy of 0.5 J for each node.

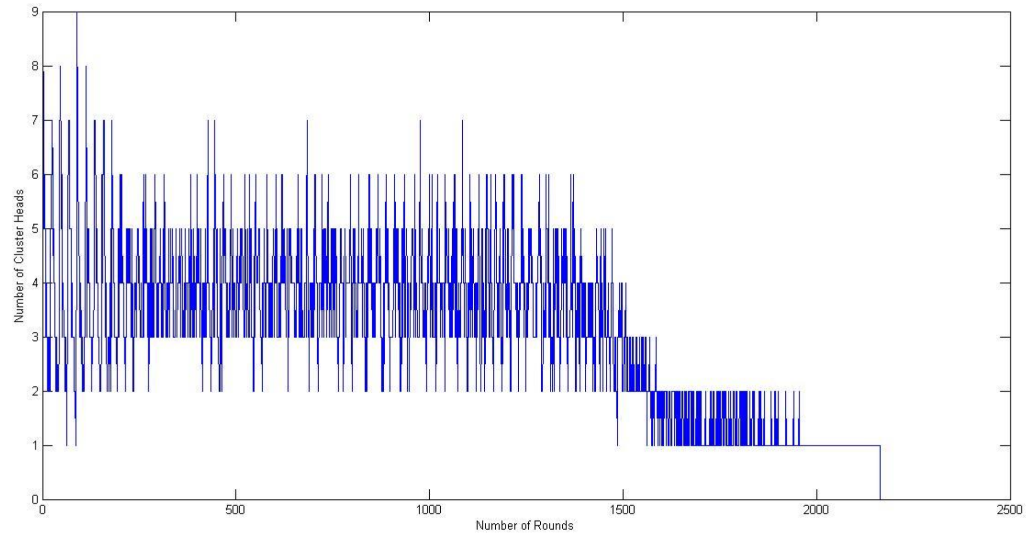


Figure 4.11: Number of cluster heads selected in each round of WEEC when base station is inside the field with initial energy of 0.5 J for each node.

4.4.3 Experiment 4

In this experiment, the base station is located inside the field with nodes' initial energy of 0.25 J.

Figure 4.12 compares the network lifetimes of LEACH and WEEC.

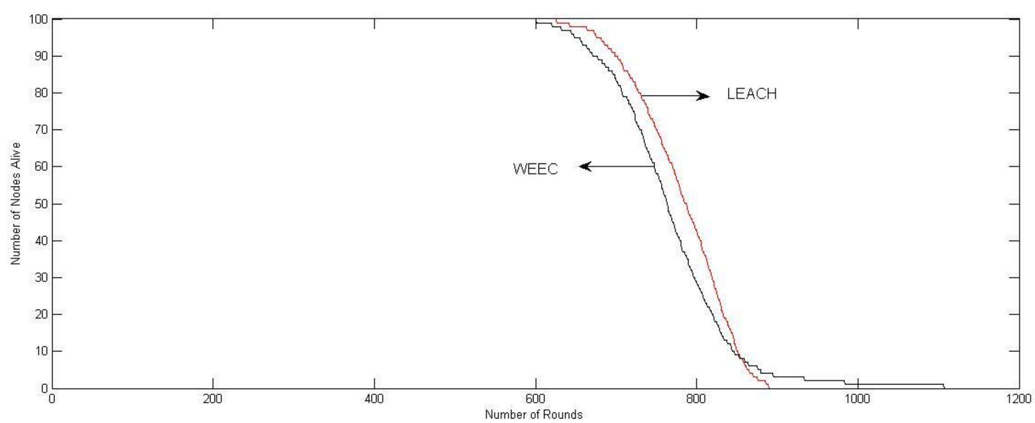


Figure 4.12: Comparison of the network's lifetime between LEACH and WEEC in experiment 4 when the base station is located inside the field with nodes' initial energy of 0.25 J.

As shown in Table (4.6), the number of rounds before the first node dies in the LEACH protocol and the WEEC protocol is 627 and 602 respectively. Nodes begin to die more quickly in the WEEC protocol than the LEACH protocol. However, the number of rounds before the last node dies in the network using the LEACH protocol is lower than the number of rounds before the last node dies in the network using the WEEC protocol. The number of rounds completed before the last node dies in LEACH and WEEC is 890 and 1107 respectively. Though the first node dies earlier in the case of the WEEC protocol than in the LEACH protocol, the network lifetime is increased by 24% based on Equation (4.1).

Table 4.6: The rounds that nodes start dying and all the nodes are dead in LEACH and WEEC when base station is located inside the field*

Protocol	Round when nodes start dying	Round when all nodes are dead
LEACH	627	890
WEEC	602	1107

***The initial energy of all nodes is 0.25 J**

By comparing Figures 4.13 and 4.14, it may be noted that the number of clusters in LEACH fluctuates from 1 to 62, which is not desired as the optimal number of cluster heads computed prior to the simulation is set to 4. In contrast, the number of clusters in WEEC fluctuates between 1 and 9, which is near the desired number of cluster heads. This deviation from the optimum number of cluster heads in the case of LEACH is also observed in Figures 4.5 and 4.6.

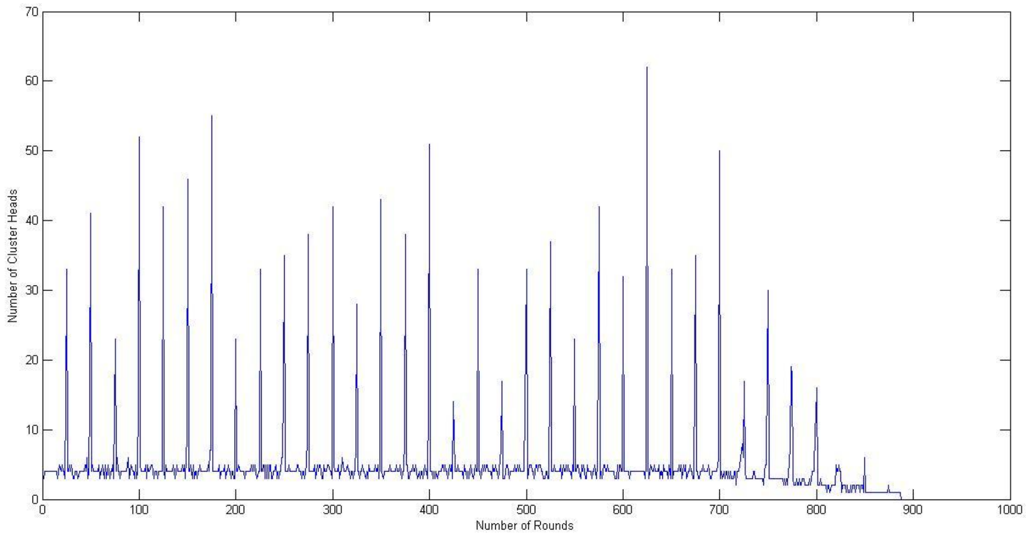


Figure 4.13: Number of cluster heads selected in each round of LEACH when the base station is inside the field with initial energy of 0.25 J for each node.

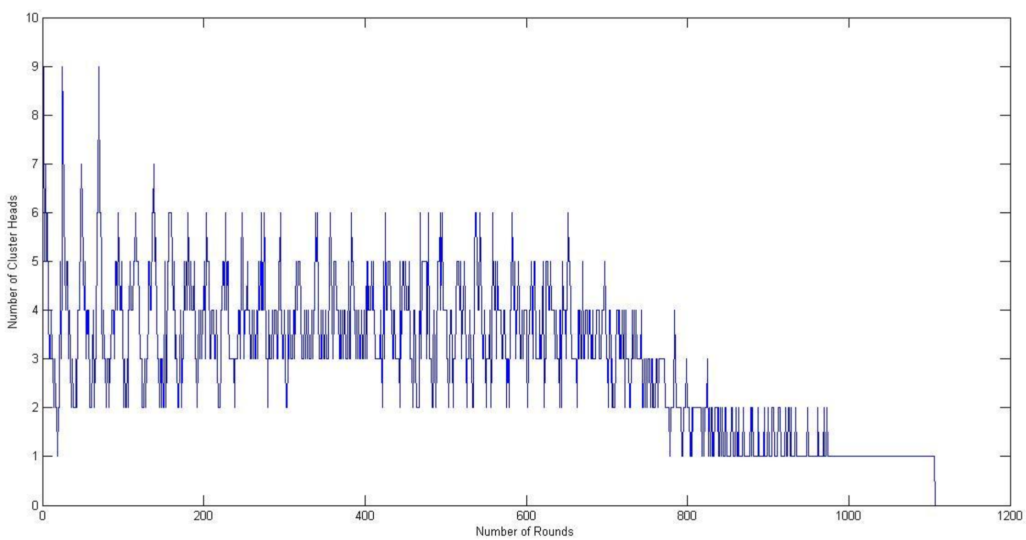


Figure 4.14: Number of cluster heads selected in each round of WEEC when the base station is inside the field with initial energy of 0.25 J for each node.

4.5 Visualization

The visualization is created by mapping accumulated the data collected during simulation to attributes of 3D objects in the Maya 2012 software [54]. Mapping between raw data and visual entities in Maya is made possible by a translation script developed in Python language. Each record of data, which is the residual energy of nodes, is marked by keyframes in the animated 3D scene. The 3D objects available through the Maya API provide a useful and natural means of visualizing textual data with 3D graphics.

In the visualization, node energy levels are displayed as elongated cubes. The heights and colours of the cubes are mapped to the energy level of a particular node. The colour ranges from green to red, which indicates the energy level from highest to lowest, respectively. When nodes die due to battery depletion, the cubes disappear from the field.

Figure 4.15 displays two fields of 10 000m^2 each, with four nodes scattered in each field. The data visualized in this visualization is based on parameters used in Experiment 4, explained in section 4.4.4. The left side of the figure displays the status of 4 nodes in round 1 using the LEACH protocol. The status of 4 nodes in round 1 utilizing the WEEC protocol is displayed on right side of Figure 4.15. The residual energy, in Joules, is shown by the number on top of each cube. The numbers shown at the bottom of each cube identify the particular node by an integer and also indicate the coordinates of each node in the field. The maximum energy level is set to 0.25J. The green colour and the near-maximum height of each cube for both protocols indicate that the energy level of each node in the field is approximately full.

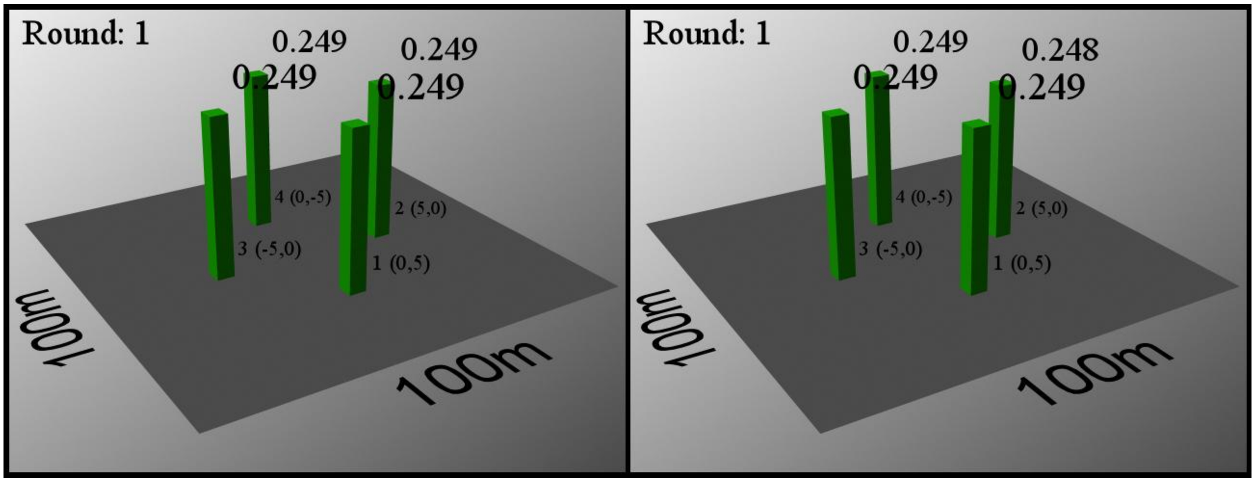


Figure 4.15: Perspective view of two fields in round 1.

Following the simulation of 450 rounds, the energy levels of all nodes are reduced in both algorithms. The fields are presented for the LEACH and WEEC algorithms in Figure 4.16 when the energy levels of all the nodes have been reduced after 450 rounds. The LEACH and WEEC algorithms are displayed on the left and right, respectively.

Similar patterns may be observed for both algorithms. Note that the bright green colours of each cube at the beginning of the simulation have now transitioned to various shades of orange and yellow. The heights of each cube are also shorter when compared to the heights of the cubes at the beginning of the simulation.

Slight differences are observed in energy levels between the two algorithms. The energy levels of nodes using the WEEC protocol, displayed on the right, are slightly higher than the energy levels of nodes using the LEACH protocol that is displayed on the left. This indicates that the energy levels of the nodes have decreased more quickly using the LEACH algorithm than when the WEEC algorithm is used.

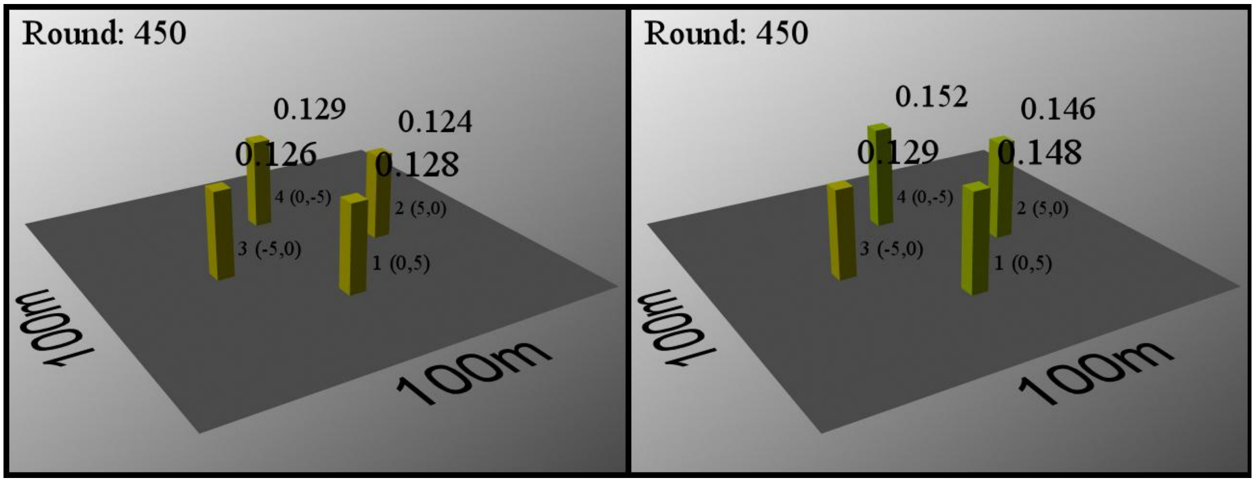


Figure 4.16: Perspective view of two fields in round 450.

Figure 4.17 presents the energy levels of surviving nodes after 890 rounds of simulation using the LEACH (left) and WEEC (right) algorithms.

In the case of the LEACH algorithm displayed on the left of Figure 4.17, nodes 1, 2 and 4 died and thus their energy levels are no longer visible. Node 4 in the LEACH algorithm is nearly dead with only 0.0009J of battery life remaining.

In contrast, the energy levels of nodes using the WEEC algorithm are displayed on the right of Figure 4.17. Note that energy level of only one node, node 3, was completely exhausted. Nodes 1, 2 and 4 have energy remaining, thereby supporting the case that using the WEEC algorithm results in an extended network lifetime.

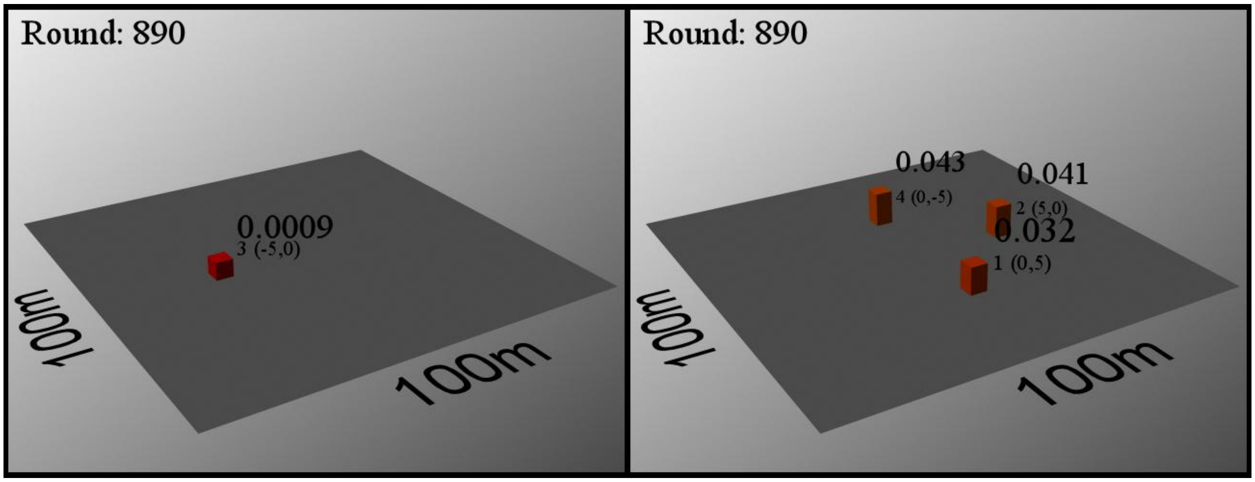


Figure 4.17: Perspective view of two fields in round 890.

Another feature of the visualization tool is visualizing the clustering formation of the field as shown in Figure 4.18.

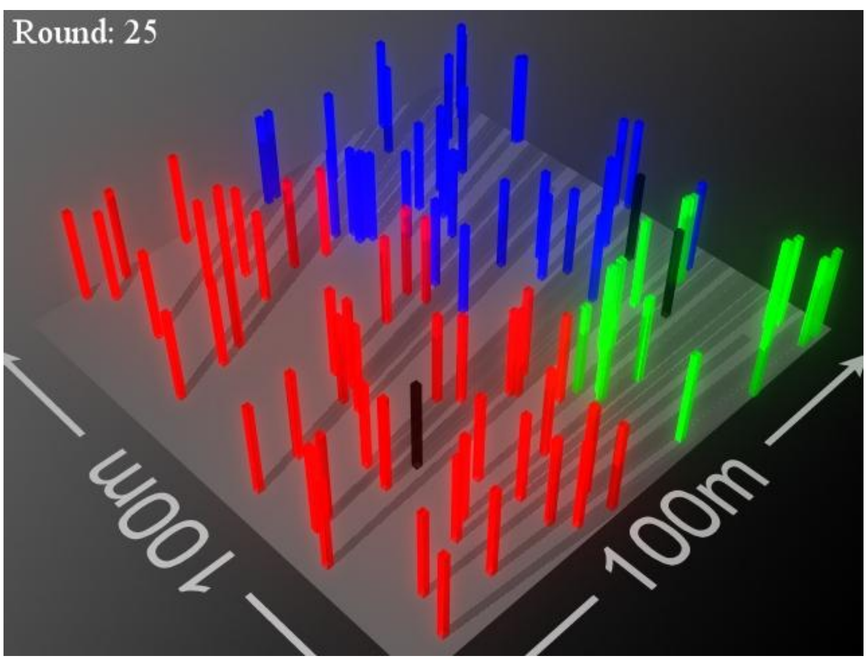


Figure 4.18: The perspective view of the clustering model.

In Figure 4.18, the cluster heads are shown by black cubes and all the non-cluster head nodes belong to each cluster are shown by the same colour but nodes belong to different cluster are represented by different colours. Therefore, the numbers of colours represents the number of clusters and show the cluster formation in each round. The data being used in this visualization is the simulated data from LEACH. In Figure 4.18, there are three clusters formed in the field showing with different colours (red, blue and green) in round 25.

4.6 Summary

In this chapter, by altering the LEACH algorithm, simulation results for a new clustering method are presented. By precisely analyzing the formula for obtaining the optimum number of cluster heads, we come up with a more effective algorithm for clustering in WSNs. Simulation results support that the network lifetime in WEEC exceeds the network lifetime in LEACH. Although WEEC improved network lifetime by up to 24% in some experiments, it is more useful when there are even small feedback about the situation of the field is needed since there are few nodes alive during the last rounds in the network. The new method for visualizing the nodes' energy levels in the field using the software Maya is also represented.

CHAPTER 5

CONCLUSIONS

A major challenge in designing an efficient protocol for wireless sensor networks is maximizing network lifetime. Clustering has been considered one of the most viable methods for addressing the challenge of maximizing network lifetime. In this thesis we proposed a novel weighted clustering method for energy efficient data transfer in wireless sensor networks. Our protocol outperforms LEACH by assigning a probability value to each node during the cluster head selection phase. In other words, the cluster definition algorithm in LEACH is modified to account for the distance between the nodes and the base station. The cluster sizes are adjusted using the analytical derivation of "*optimum number of clusters* "explained in the LEACH protocol. The cluster head selection mechanism in the LEACH protocol results in a great degree of variability in the number of cluster heads. In the WEEC protocol by considering the distance of the nodes to the base station as an important factor in cluster head selection, we reduced the variation in the number of cluster heads and maintained the actual number of clusters close to the optimum number of cluster heads. The consequence of sustaining the actual number of cluster heads close to the optimum value is prolonged network lifetime, since less energy is dissipated in each round.

The restriction for placing the base station in a specific location is relaxed in our work. The WEEC protocol supports the placement of the base station in any location.

The simulation model is designed with a fixed number of nodes uniformly distributed in a square field. Simulation results, presented in Chapter 4, show that WEEC outperformed LEACH in regards to network lifetime in all the experiments. For example, in Experiment 3, explained in chapter 4, section 4.4.2, the network lifetime increased by 22% on average. In the same Experiment, the number of cluster heads selected in the LEACH protocol varies between 1 and 57, while the variation in the number of cluster heads in the WEEC protocol is between 1 and 9.

However, WEEC is more suitable for applications in which even small amount of knowledge from the field is crucial. For example in Experiments 3 and 4, although the network lifetime improved by 22% and 24% respectively, LEACH is able to keep more nodes alive during the last rounds of the simulation in comparison to WEEC. Therefore, when the target of the application is having more areas covered by sensor nodes, LEACH would be more efficient. In contrast, in applications where obtaining data from small areas of the field is considered important, WEEC would be more efficient.

A new visualization method is also introduced in this thesis. The main reason of the visualization described in this thesis is to visualize the residual energy data of WSNs. Therefore, users are able to easily understand the energy level of the nodes in the field. This visualization tool reads the energy levels of nodes as well as their location from the collected simulation data and presents the energy level of each node at each round of simulation. The energy levels are represented by the colour and size of cubes in 3D scene. Moreover, to make the visualization tool easier to understand the number of rounds and the residual energy data of each round are also displayed in the viewing window.

5.1 Future work

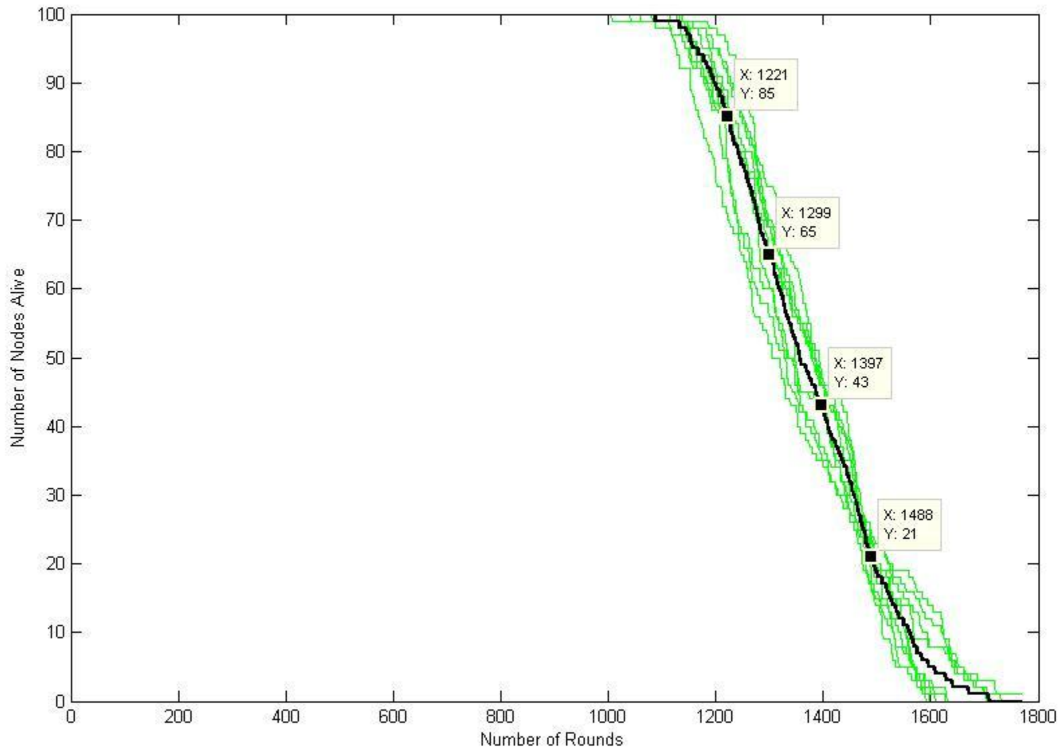
One possibility for improving the current algorithm is to consider the residual energy of each node during the cluster head selection phase. During the cluster head selection phase, if we reduce the probability of nodes with less residual energy the energy distribution in the network would be more balanced. Another work that could be done in order to prolong network lifetime is to consider residual energy while selecting cluster heads and increase the probability of the nodes which have more energy level.

A further consideration would potentially also account for the mobility of the sensor nodes. Moreover, some other scenarios such as reformation of clusters after death of a node or entry of a new node to the system could be considered in the future work.

APPENDIX I.

Experiment 1.

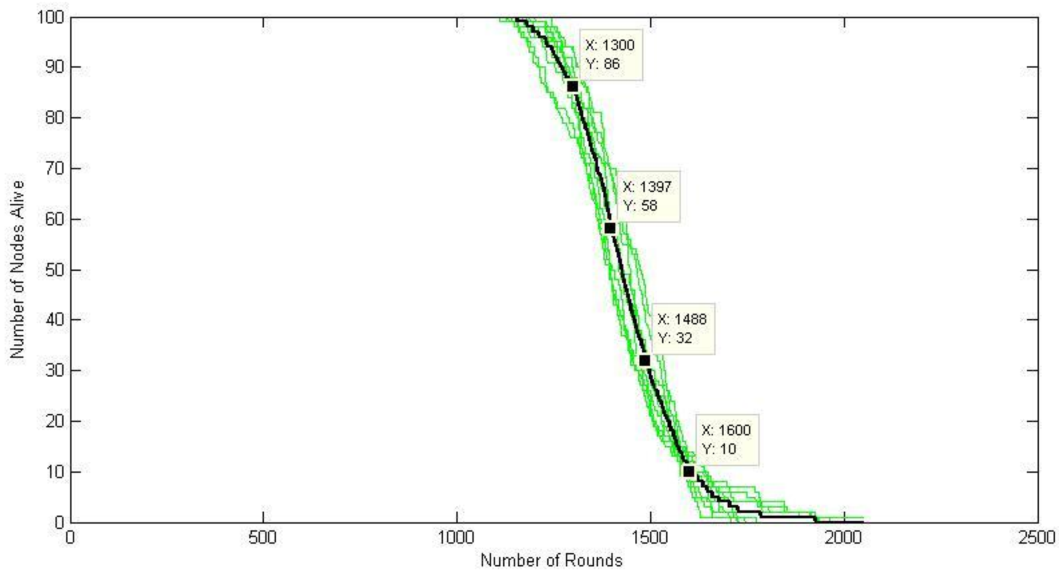
LEACH - 10 Experiments shown by green lines and the average is shown by a black line. The average is used in Figure 4.1 by red line.



Number of Nodes Alive for random rounds in 10 experiments in LEACH

Round	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**	$\boldsymbol{E_{Ave}}$
1221	96	85	89	79	92	78	87	71	85	91	85
1299	75	61	69	56	69	52	66	60	70	69	65
1397	48	37	43	35	46	35	47	42	46	46	43
1488	21	25	17	22	24	23	19	24	17	22	21

WEEC- 10 Experiments shown by green lines and the average is shown by a black line. The average is used in Figure 4.1 by black line.

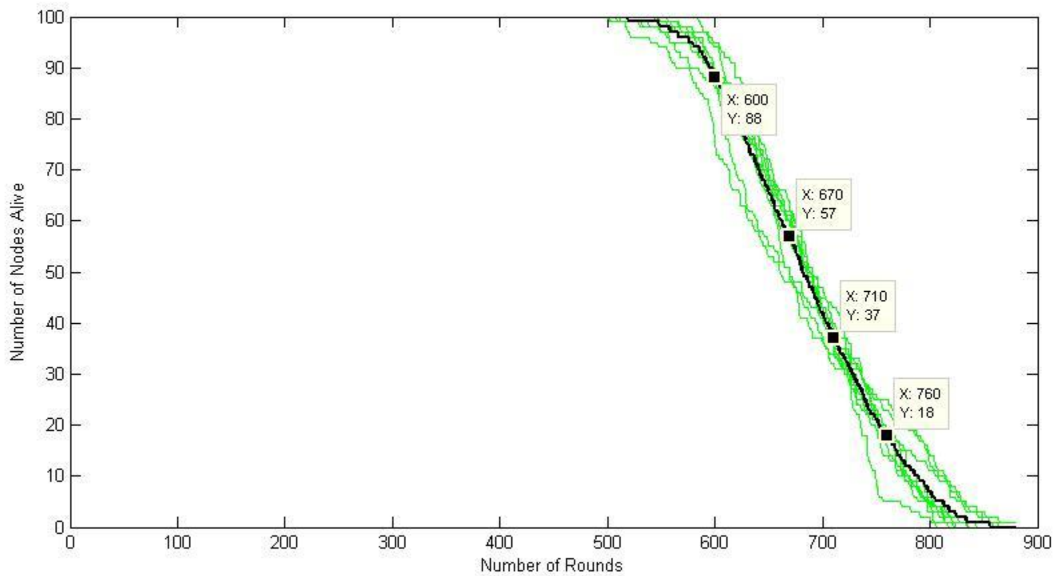


Number of Nodes Alive for random rounds in 10 experiments in WEEC

Round	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**	E_{Ave}
1300	94	83	87	78	93	76	88	82	87	90	86
1397	70	49	57	48	69	50	60	51	58	70	58
1488	40	29	34	31	33	24	27	26	35	45	32
1600	13	11	8	14	12	10	8	8	7	13	10

Experiment 2.

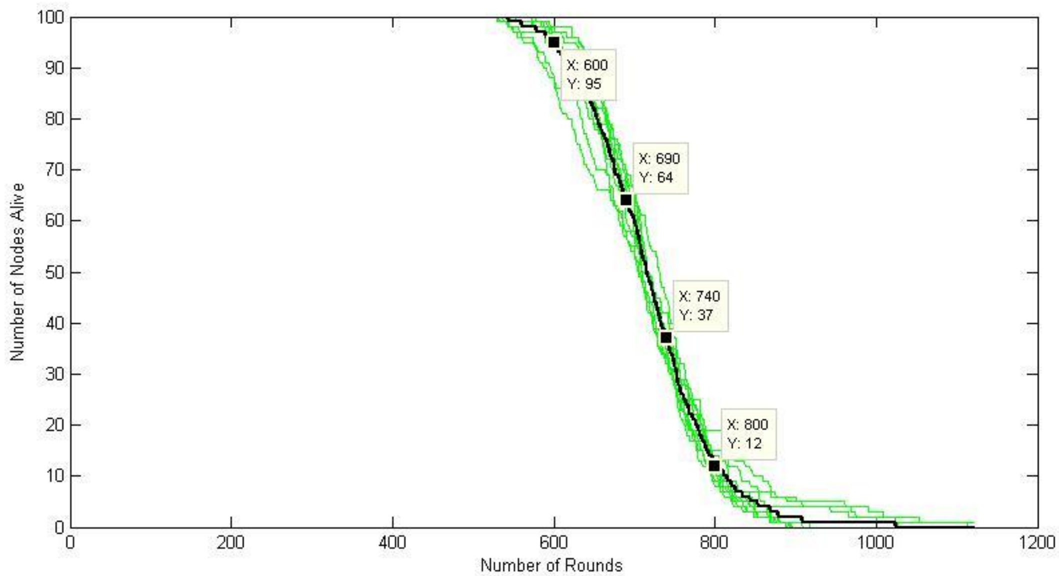
LEACH- 10 Experiments shown by green lines and the average is shown by a black line. The average is used in Figure 4.4 by red line.



Number of Nodes Alive for random rounds in 10 experiments in LEACH

Round	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E_{Ave}
600	94	90	87	95	89	86	87	89	76	89	88
670	49	60	48	60	62	57	62	59	49	61	57
710	34	43	32	36	40	36	37	39	38	39	37
760	25	19	19	19	18	5	16	17	23	14	18

WEEC- 10 Experiments shown by green lines and the average is shown by a black line. The average is used in Figure 4.4 by black line.

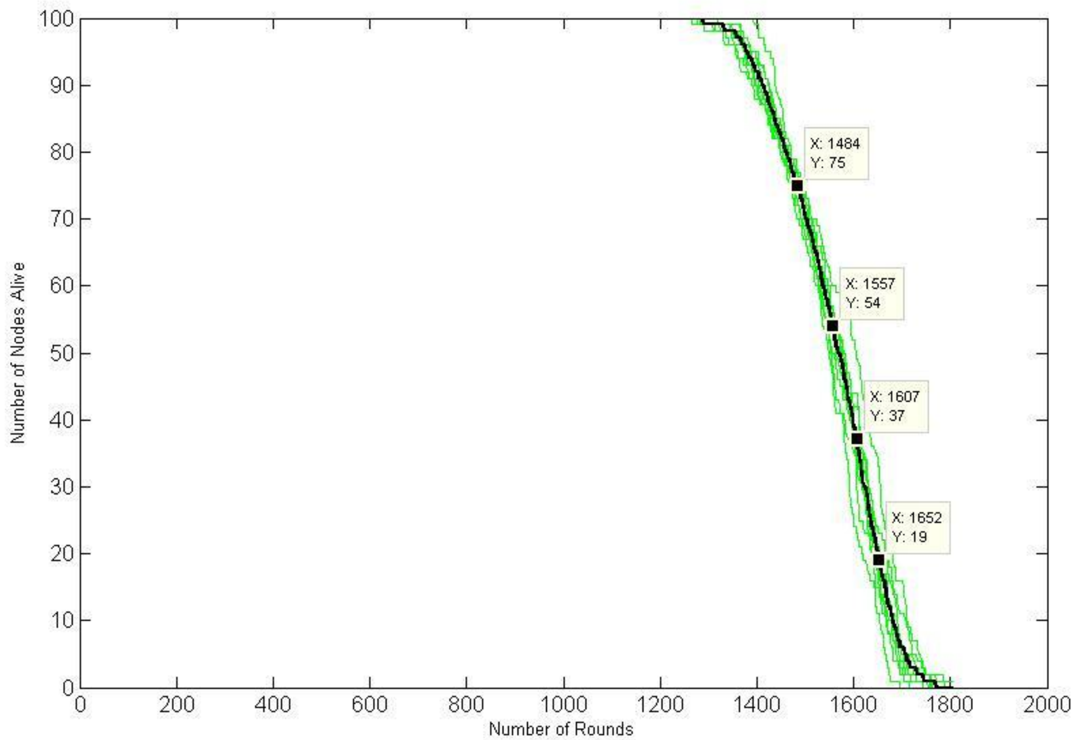


Number of Nodes Alive for random rounds in 10 experiments in WEEC

Round	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**	$\boldsymbol{E_{Ave}}$
600	96	98	88	96	97	95	94	98	88	96	95
690	58	67	58	66	66	60	70	70	57	68	64
740	38	35	33	38	33	32	37	42	34	45	37
800	19	12	14	9	13	11	15	8	11	12	12

Experiment 3.

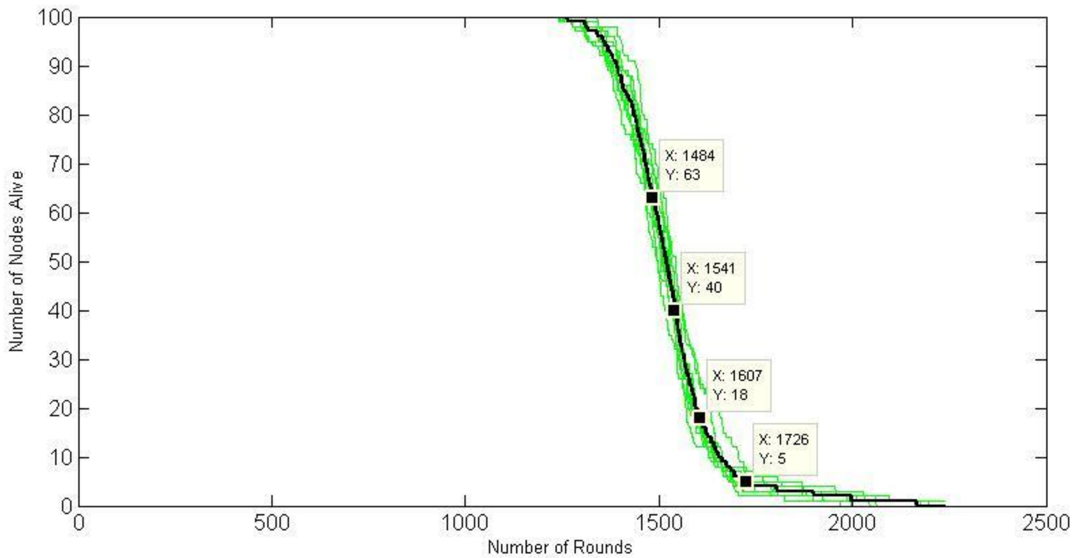
LEACH - 10 Experiments shown by green lines and the average is shown by a black line. The average is used in Figure 4.9 by red line.



Number of Nodes Alive for random rounds in 10 experiments in LEACH

Round	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**	$\boldsymbol{E_{Ave}}$
1484	75	77	77	73	77	70	72	76	76	74	75
1557	54	48	50	46	56	57	56	56	60	53	54
1607	42	29	34	23	36	44	37	34	49	38	37
1652	21	17	15	11	21	23	17	15	33	20	19

WEEC - 10 Experiments shown by green lines and the average is shown by a black line. The average is used in Figure 4.9 by black line.

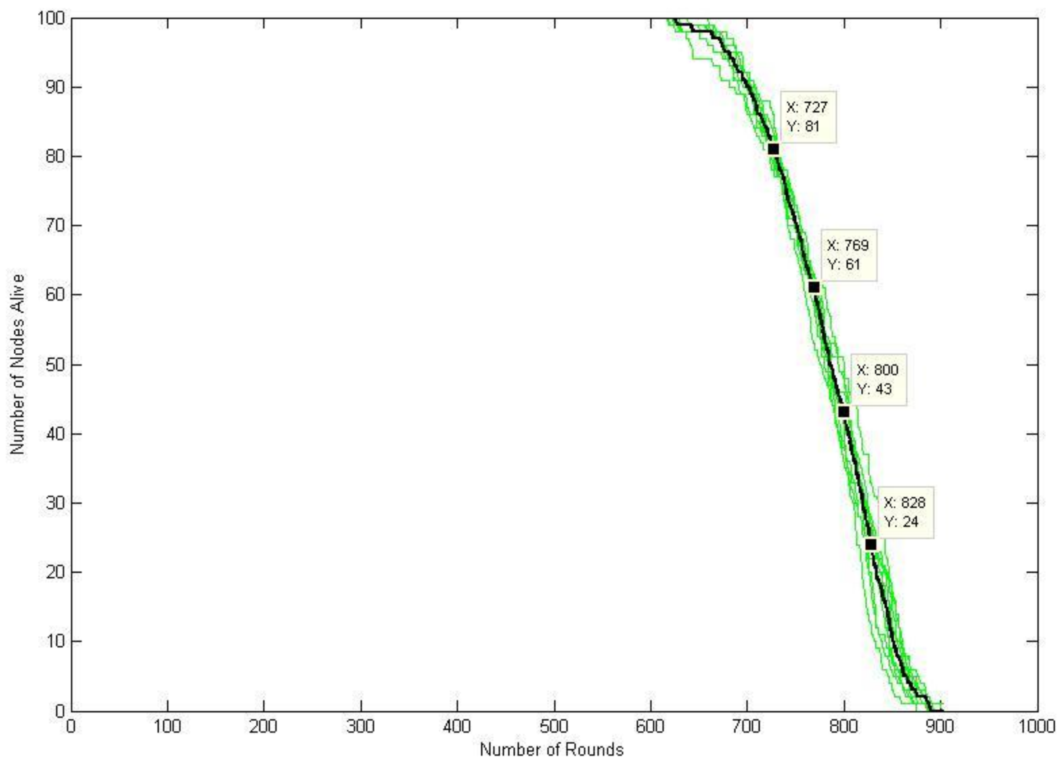


Number of Nodes Alive for random rounds in 10 experiments in WEEC

Round	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**	E_{Ave}
1484	64	59	61	53	73	69	58	61	69	64	63
1541	43	36	40	36	43	51	32	39	45	37	40
1607	18	15	17	12	17	24	15	16	25	17	18
1726	6	4	5	3	2	7	6	5	3	5	5

Experiment 4.

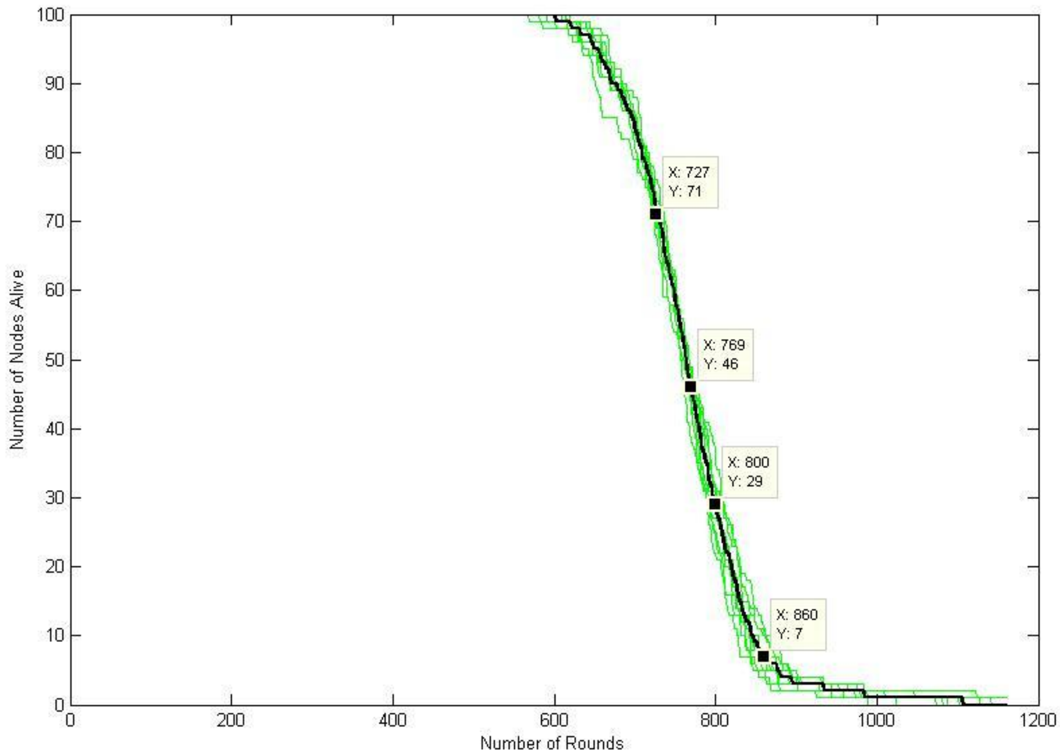
LEACH - 10 Experiments shown by green lines and the average is shown by a black line. The average is used in Figure 4.12 by red line.



Number of Nodes Alive for random rounds in 10 experiments in LEACH

Round	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**	E_{Ave}
727	82	83	80	80	79	83	81	86	78	80	81
769	61	62	53	63	58	63	61	60	62	62	61
800	48	38	39	49	41	42	46	35	41	50	43
828	26	20	17	28	25	26	27	12	22	33	24

LEACH - 10 Experiments shown by green lines and the average is shown by a black line. The average is used by Figure 4.12 by black line.



Number of Nodes Alive for random rounds in 10 experiments in WEEC

Round	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**	E_{Ave}
727	73	76	71	70	70	73	71	72	68	69	71
769	43	48	38	48	48	43	49	46	46	49	46
800	25	31	25	32	32	27	35	22	31	31	29
860	6	10	4	8	11	8	11	5	5	6	7

REFERENCES

[1] C.Y Chong, S. P. Kumar., Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE* , vol. 91, no. 8, pp. 1247- 1256, 2003.

[2] M. Ilyas and I. Mahgoub., Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems, *International Journal of Distributed Sensor Networks*, vol. 4, no. 4, pp. 369-369, 2008.

[3] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, *Computer Networks*, vol. 38, Issue 4, pp. 393-422, 2002.

[4] I. Demirkol, C. Ersoy, and F. Alagoz, MAC Protocols for Wireless Sensor Networks: A Survey, *IEEE Communication Magazine*, vol. 44, Issue 4, pp.115-121, 2006.

[5] J. N. Al-Karaki and A. E. Kamal. Routing techniques in wireless sensor networks: a survey," *Wireless Communications, IEEE* , vol.11, no.6, pp. 6- 28, 2004.

[6] W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *Proceedings of the 33rd Annual Hawaii International Conference on system Sciences,* vol.2, pp. 3005-3014, 2000.

[7] N. Behboudi and A. Abhari. , "A Weighted Energy Efficient Clustering (WEEC) for Wireless Sensor Networks. *Seventh International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, pp.146-151, 2011

[8] B. G Celler, T. Hesketh, W. Earnshaw and E. Ilsar. An instrumentation system for the remote monitoring of changes in functional health status of the elderly at home, *Proceedings of the 16th Annual International Conference of the IEEE-EMBS* , vol 2, pp.908-909, 1994.

[9] G. Coyle, L. Boydell and L. Brown. Home telecare for the elderly. Journal of Telemedicine and Telecare, vol 1, pp183-184, 1995.

[10] X. Li, L. Xu, H. Wang, J. Song, S. X. Yang. A Differential Evolution-Based Routing Algorithm for Environmental Monitoring Wireless Sensor Networks. Sensors, vol. 10, pp. 5425-5442, 2010.

[11] T. Gao, D. Greenspan, M. Welsh, R.R Juang and A. Alm. Vital Signs Monitoring and Patient Tracking Over a Wireless Network. *27th Annual International Conference of the Engineering in Medicine and Biology Society*, vol. 1, pp.102-105, 2005.

[12] K. Lorincz, D. Malan, T.R.F. Fulford-Jones, A. Nawoj, A. Clavel, V.Shnayder, G. Mainland, M. Welsh, S. Moulton. Sensor networks for emergency response: challenges and opportunities," *Pervasive Computing, IEEE* , vol.3, no.4, pp. 16- 23, 2004.

[13] M. Castillo-Effen, D.H. Quintela, R. Jordan, W. Westhoff, W. Moreno. Wireless sensor networks for flash-flood alerting. *Proceedings of the Fifth IEEE International Caracas Conference on Devices, Circuits and Systems*, vol. 1, pp. 142- 146, 2004.

[14] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, vol. 40, pp. 102–114, 2002.

[15] G. Asada, M. Dong, T. S. Lin, F. Newberg, G. Pottie, W. J Kaiser and H. O. Marcy. Wireless integrated network sensor: Low power systems on a chip. *Proceedings of the 24th European Solid-State Circuits Conference*, pp. 9-16, 1998.

[16] E. Shih, S. H. Cho, N. Ickes, R. Min, A. Sinha, A. Wang and A. Chandrakasan. Physical Layer Driven Protocol and Algorithm Design for Energy-Efficient Wireless Sensor Networks. *Proceedings of the 7th annual international conference on Mobile computing and networking*, pp. 272-287, 2001.

[17] Crossbow Technology Inc. Wireless Sensor Networks product reference guide. http:// www.xbow.com, 2007. [Online].

[18] Freescale Semiconductor Inc. Freescale Sensor Products datasheet. http://www.freescale.com, 2007. [Online]

[19] J.M Rabaey, M.J. Ammer, J. J. L. da Silva, D. Patel, and S. Roundy. PicoRadio supports ad hoc ultra-low power wireless networking, *IEEE Computer*, vol. 33, no.7, pp.42-48, 2000.

[20] M. A. M. Vieira, J. C. N. Coelho, J. D. C. da Silva, and J.M. da Mata. Survey on wireless sensor network devices. *Proceedings of the ninth IEEE Conference on Emerging Technologies and Factory Automation.* vol.1, pp. 537- 544, 2003.

[21] W. Ye, J. Heidemann, and D. Estrin. An energy-efficient MAC protocol for wireless sensor networks. *Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies.* vol.3, pp. 1567- 1576, 2002.

[22] V. Cionca, T. Newe and V. Dadarlat. TDMA Protocol Requirements for Wireless Sensor Networks. *Second International Conference on Sensor Technologies and Applications*, pp.30-35, 25, 2008.

[23] T. van Dam and K. Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks. *In Proceedings of the First ACM Conference on Embedded Networked Sensor Systems*, pp. 171-180, 2003.

[24] L. van Hoesel and P. Havinga. A lightweight medium access protocol (LMAC) for wireless sensor networks. *In Proceedings of the 3rdInternational Symposium on Information Processing in Sensor Networks (IPSN)*, pp. 55–60, 2004.

[25] T. Nieberg, S. Dulman, P. Havinga, L. V. Hoesel and J. Wu. “Collaborative Algorithms for Communication in Wireless Sensor Networks” in Ambient Intelligence: Impact on Embedded Systems, T. Basten, M. Geilen and H. Groot, Springer US, pp. 271-294, 2004.

[26] S. Dulman, T. Nieberg, J. Wu, and P. Havinga, “Trade-off between traffic overhead and reliability in multipath routing for wireless sensor networks,” in *Proceedings of the Wireless Communications and Networking Conference*, vol.3, pp. 1918–1922, 2003.

[27] D. Braginsky and D. Estrin. Rumor routing algorthim for sensor networks. *In Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. *ACM*, pp. 22-31, 2002.

[28] S. Capkun, M. Hamdi, J. Hubaux, GPS-free positioning in mobile ad-hoc networks. *in processing of the 34th Hawaii International Conference on System Sciences*, pp. 3481–3490, 2001.

[29] Q. Fang, F. Zhao, and L. Guibas. Lightweight sensing and communication protocols for target enumeration and aggregation. *In Proceedings of the 4th ACM international symposium on Mobile ad hoc networking*, pp. 165-176, 2003.

[30] C. Karlof and D. Wagner. Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures. *Ad Hoc Networks*, vol. 1, pp. 293–315, 2003.

[31] K. Sohrabi, J. Gao, V. Ailawadhi and G. J. Pottie. Protocols for self-organization of a wireless sensor network. *IEEE Personal Communications*, vol. 7, no. 5, pp.16-27, 2000.

[32] N. Bulusu, D. Estrin, L. Girod and J. Heidemann. Scalable coordination for wireless sensor networks: self-configuring localization systems, *In processing of sixth International Symposium on Communication Theory and Applications*, 2001.

[33] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler. 2002. SPINS: security protocols for sensor networks. *Wireless Networks*, vol. 8, no. 5, pp.521-534, 2002.

[34] Y. Yu, D. Estrin, and R. Govindan, “Geographical and Energy-Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks,” *UCLA Computer Science Department, UCLA-CSD TR-010023*, 2001

[35] W. R. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. *In Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pp. 174-185, 1999.

[36] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann and F. Silva. Directed diffusion for wireless sensor networking. *IEEE/ACM Transactions on Networking*,, vol.11, no.1, pp. 2- 16, 2003.

[37] O. Younis, M.Krunz, and S.Ramasubramanian. Node clustering in wireless sensor networks: recent developments and deployment challenges. *IEEE Network* , vol.20, no.3, pp. 20-25, 2006.

[38] K. Akkaya and M. Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, vol. 3, no.3, pp. 325-349, 2005.

[39] W. B. Heinzelman, A. P. Chandrakasan and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications,* vol. 1, no. 4, pp. 660- 670, 2002.

[40] S. S. Satapathy and N. Sarma. TREEPSI: tree based energy efficient protocol for sensor information. *IFIP International Conference on Wireless and Optical Communications Networks*, pp.4, 2006.

[41] K. T. Kim, C. H. Lyu, S. S. Moon and H. Y. Youn. Tree-Based Clustering (TBC) for Energy Efficient Wireless Sensor Networks. *The twenty-fourth IEEE International Conference on Advanced Information Networking and Applications Workshops*, pp.680-685, 2010.

[42] K. Kim and H. Youn. “Energy-driven Adaptive Clustering Hierarchy (EDACH) for wireless sensor networks” in Embedded and Ubiquitous Computing, L. Yan, B. Xiao, D. Kim, Y.Dai and L. Yang, *Springer Berlin*, pp.1098– 1107, 2005.

[43] A. Manjeshwar and D. P. Agarwal. TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. *In Proceedings of 15th International Parallel and Distributed Processing Symposium*, pp.2009-2015, 2001.

[44] XuYang Ding; Hai Zhao; Jian Zhu; Kuan Zhang; DaZhou Li; , "A Novel Localization Algorithm Based on RSSI for Wireless Sensor Networks," *7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*,, no., pp.1-4, 2011.

[45] T. S. Rappaport, Wireless Communications: Principles and Practice, Prentice-Hall, Englewood Cliffs, NJ, 1996.

[46] F. A. Aderohunmu, Energy Management Techniques in Wireless Sensor Networks: Protocol Design and Evaluation. MASc thesis, University of Otago, New Zealand, 2010.

[47] N. Behboudi and A. Abhari, Adaptive Clustering for Energy Optimization of Variant Base Station in Wireless Sensor Networks, *Spring Simulation Multiconference (SpringSim'11 Poster Track)*, 2011.

[48] I. Dietrich and F. Dressler, On the lifetime of wireless sensor networks, *ACM Transactions on Sensor Networks (TOSN)*, pp. 1-39 Vol.5, Issue 1, 2009.

[49]Y. Hu, D. Li, X. He, T. Sun and Y. Han. The Implementation of Wireless Sensor Network Visualization Platform Based on Wetland Monitoring. *Second International Conference on Intelligent Networks and Intelligent Systems, ICINIS '09*. pp. 224-227, 2009.

[50] Y. Yang, P. Xia, L. Huang, Q. Zhou, Y. Xu and X. Li. SNAMP: A Multi-sniffer and Multi-view Visualization Platform for Wireless Sensor Networks. *In proceedings* of *1ST IEEE Conference on Industrial Electronics and Applications*, pp.1-4, 2006.

[51] Jerry Zhao, "Measurement and Monitoring in Wireless Sensor Networks", Ph.D. Dissertation, University of Southern California, 2003.

[52] J. Blumentha, M. Handy, F. Golatowski and M. Haase. Wireless sensor networks - new challenges in software engineering", *In Proceedings of IEEE Conference on Emerging Technologies and Factory Automation*, vol.1, pp. 551-556, 2003.

[53] G. Smaragdakis, I. Matta and A. Bestavros, SEP: A Stable Election Protocol for clustered heterogeneous wireless sensor networks, *Second International Workshop on Sensor and Actor Network Protocols and Applications (SANPA 2004)*, 2004.

[54] Autodesk Maya, http://usa.autodesk.com/maya/ [Online]