

1-1-2012

Increasing Network Lifetime In Cluster Based Wireless Sensor Networks Via Fuzzy Logic

Rouzbeh Behrouz
Ryerson University

Follow this and additional works at: <http://digitalcommons.ryerson.ca/dissertations>



Part of the [Computer Sciences Commons](#)

Recommended Citation

Behrouz, Rouzbeh, "Increasing Network Lifetime In Cluster Based Wireless Sensor Networks Via Fuzzy Logic" (2012). *Theses and dissertations*. Paper 1269.

This Thesis is brought to you for free and open access by Digital Commons @ Ryerson. It has been accepted for inclusion in Theses and dissertations by an authorized administrator of Digital Commons @ Ryerson. For more information, please contact bcameron@ryerson.ca.

INCREASING NETWORK LIFETIME IN CLUSTER BASED WIRELESS SENSOR NETWORKS VIA FUZZY LOGIC

by

Rouzbeh Behrouz

BSc. in Computer Science, Ryerson University, Canada, 2004

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

© Rouzbeh Behrouz, 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.

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 of the purpose of scholarly research.

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

ROUZBEH BEHROUZ

INCREASING NETWORK LIFETIME IN CLUSTER BASED WIRELESS SENSOR NETWORKS VIA FUZZY LOGIC

Rouzbeh Behrouz

Master of Science, Computer Science, 2012

Ryerson University, Toronto, Ontario, Canada

ABSTRACT

Energy efficient operation is a critical issue that has to be addressed with large-scale wireless sensor networks deployments. Cluster-based protocols are developed to tackle this problem and Low Energy Adaptive Clustering Hierarchy (LEACH) is one of the best-known protocols of this type. However, certain aspects of LEACH offer room for improvement. One such aspect is the arrangement of wireless sensor network with the fixed base station location. In this thesis we propose Fuzzy Logic for Mobile Base Station (FLMBS) protocol that is based on LEACH but uses a Fuzzy Inference System driven approach to adjust the location of the base station. FLMBS produces reasonable improvement over LEACH in a network area greater than $1000 \times 1000 \text{ m}^2$. We also compare FLMBS to another protocol that uses Particle Swarm Optimization in order to find the location of base station. PSO outperforms FLMBS in $1000 \times 1000 \text{ m}^2$ but the difference is marginal, especially considering the larger complexity of PSO compared to FLMBS.

ACKNOWLEDGEMENTS

I am heartily thankful to my supervisor, Dr. Abdolreza Abhari, for his professional guidance and suggestions throughout my thesis studies.

I would also acknowledge Dr. Hooman Tahayori for his advice and his willingness to share his bright thoughts about fuzzy logic with me.

Furthermore, I would like to thank my wife and my parents for their love and encouragement over the years.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of this thesis, as well as expressing my apology that I could not mention personally one by one.

TABLE OF CONTENTS

AUTHOR’S DECLARATION.....	ii
ABSTRACT	iii
ACKNOWLEDGEMENTS.....	iv
LIST OF TABLES	ix
LIST OF FIGURES	x
LIST OF ABBREVIATIONS AND SYMBOLS USED	xiii
LIST OF ABBREVIATIONS.....	xiii
LIST OF SYMBOLS.....	xv
CHAPTER I.....	1
Introduction	1
1.1 Motivation.....	2
1.2 Objective and Scope	3
1.3 Thesis Contribution	4
1.4 Thesis Outline.....	5
CHAPTER II	7
Background and Related work.....	7
2.1 Wireless Sensor Network.....	7
2.1.1 WSN Hardware.....	10
2.1.2 WSN routing protocols	11

2.1.3 Cluster-based Wireless Sensor Networks	12
2.1.4 LEACH: Low-Energy Adaptive Clustering Hierarchy for Wireless Micro-sensor Networks.....	14
2.2 Swarm Intelligence	17
2.2.1 Particle Swarm Optimization	17
2.2.2 Particle Swarm Optimization Algorithm.....	19
2.2.3 Related Work.....	20
2.3 Fuzzy Logic.....	23
2.3.1 Crisp and Fuzzy Sets.....	23
2.3.2 Properties of Fuzzy Sets.....	25
2.3.3 Operations on Fuzzy Sets.....	26
2.3.4 Linguistic Variables and Linguistic Values	27
2.3.5 Fuzzy IF-THEN rules.....	27
2.3.6 Fuzzy Inference System	28
2.3.7 Related Work.....	31
CHAPTER III	33
Improving LEACH Protocol Using Fuzzy Logic Approach.....	33
3.1 FLMBBS Protocol Overview	33
3.2 FIS Design.....	35
3.3 FIS Parameters and Rules	35
3.4 Determination of CH priority value.....	40
3.4.1 Step 1: Input of Crisp Value and Fuzzification	40

3.4.2 Step 2: Rule Evaluation.....	44
3.4.3 Step 3: Aggregation of the Rule Outputs	46
3.4.4 Step 4: Defuzzification.....	46
3.5 BS Movement Example	48
CHAPTER IV	54
Simulation and Results.....	54
4.1 Network Model and Assumptions	54
4.2 First Group of Experiments (FLMBS vs LEACH)	55
4.2.1 Experiment 1.....	58
4.2.2 Experiment 2.....	62
4.2.3 Experiment 3.....	68
4.2.4. Experiment 4.....	71
4.3 Second Group of Experiments (FLMBS vs LEACH vs PSO).....	74
4.3.1 Experiment 1.....	75
4.3.2 Experiment 2.....	79
4.3.3 Experiment 3.....	82
4.4 Summary	85
CHAPTER V	86
Conclusions	86
5.1 Future work	88
Appendix A: Triangular and Trapezoidal MFs	90

Appendix B: Distance and Energy MF for a Larger Grid.....	92
References	93

LIST OF TABLES

Table 3.1: Fuzzy inference system IF-THEN rules	40
Table 3.2: Fuzzy inference system IF-THEN rules (Example)	46
Table 3.3: Priority value for different cluster-heads	49
Table 3.4: Example parameters.....	50
Table 4.1: Input parameters (First group of experiments).....	57
Table 4.2: Experiment 1: FND metrics for FLMBS and LEACH.....	61
Table 4.3: Experiment 2: FND metrics for FLMBS and LEACH.....	65
Table 4.4: Experiment 2: Energy dissipation after each round.....	68
Table 4.5: Experiment 3: FND metrics for FLMBS and LEACH.....	71
Table 4.6: Experiment 3: Energy dissipation after each round.....	73
Table 4.7: Experiment 4: FND metrics for FLMBS and LEACH.....	75
Table 4.8: Input parameters (Second group of experiments).....	77
Table 4.9: Experiment 1: FND metrics for FLMBS, LEACH and PSO	79
Table 4.10: Experiment 1: Energy dissipation after each round.....	80
Table 4.11: Experiment 2: FND metrics for FLMBS, LEACH and PSO	82
Table 4.12: Experiment 2: Energy dissipation after each round.....	83
Table 4.13: Experiment 3: FND metrics for FLMBS, LEACH and PSO	86

LIST OF FIGURES

Figure 2.1: WSN overview	9
Figure 2.2: General hardware architecture of a sensor node	10
Figure 2.3: Cluster based architecture for WSN.....	12
Figure 2.4: Radio energy dissipation model	15
Figure 2.5: A basic fuzzy set	24
Figure 2.6: Fuzzy Sets Operations	26
Figure 2.7: FIS structure	28
Figure 3.1: Fuzzy Inference System for FLMBS.....	35
Figure 3.2: Membership Function of Initial Energy	37
Figure 3.3: Membership Function of Distance to the BS.....	37
Figure 3.4: Membership Function to Data Transmission Rate.....	37
Figure 3.5: Membership Function of Priority.....	38
Figure 3.6: Fuzzification of crisp input Initial Energy (0.23J).....	42
Figure 3.7: Fuzzification of crisp input distance (45m).....	43
Figure 3.8: Fuzzification of the crisp input Rate (3.2kbps).....	44
Figure 3.9: Output evaluation of fuzzy IF-THEN rules	47
Figure 3.10: Centroid point.....	49
Figure 3.11: The position of the given CHs and the initial BS position	51
Figure 3.12: Vectors between CHs and BS	51
Figure 3.13: Resulting vectors	53

Figure 3.14: Net movement vector.....	54
Figure 4.1: Experiment 1: The number of alive nodes for an area 100m x 100m with 100 SNs and BS at (50,150).....	61
Figure 4.2: Experiment 1: FND values for both FLMBS and LEACH	62
Figure 4.3: Experiment 2: The number of alive nodes for an area 100m x 100m with 100 SNs and BS at (50,50).....	64
Figure 4.4: Experiment 2: FND values for both FLMBS and LEACH	65
Figure 4.5: Experiment 3: The number of alive nodes for an area 1000m x 1000m with 100 SNs and BS at (500,500).....	70
Figure 4.6: Experiment 3: FND values for both FLMBS and LEACH	72
Figure 4.7: Experiment 4: The number of alive nodes for an area 1000m x 1000m with 200 SNs and BS at (500,500).....	74
Figure 4.8: Experiment 4: FND values for both FLMBS and LEACH	75
Figure 4.9: Experiment 1: The number of alive nodes for an area 100m x 100m with 100 SNs and BS at (50,50).....	78
Figure 4.10: Experiment 1: FND values for both FLMBS, LEACH and PSO	79
Figure 4.11: Experiment 2: The number of alive nodes for an area 1000m x 1000m with 100 SNs and BS at (500,500)	82
Figure 4.12: Experiment 2: FND values for both FLMBS, LEACH and PSO	83
Figure 4.13: Experiment 3: The number of alive nodes for an area 1000m x 1000m with 200 SNs and BS at (500,500)	85
Figure 4.14: Experiment 3: FND values for both FLMBS, LEACH and PSO	86

Figure 4.15: MF for energy in 1000 x 1000 m ² grid.....	94
Figure 4.16: MF for distance in 1000 x 1000 m ² grid.....	94

LIST OF ABBREVIATIONS AND SYMBOLS USED

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimization
ADC	Analog to Digital Conversion
BS	Base Station
CH	Cluster Head
COG	Center Of Gravity
DC	Direct Communication
FIS	Fuzzy Inference System
FL	Fuzzy Logic
FLMBS	Fuzzy Logic for Mobile Base Station
FND	First Node Death
GPS	Global Positioning System
IP	Internet Protocol
LEACH	Low-Energy Adaptive Clustering Hierarchy
LND	Last Node Death
MATLAB	Matrix Laboratory
MF	Membership Function
MTE	Minimum Transmission Energy

MAC	Medium Access Control
PSO	Particle Swarm Optimization
PEGASIS	Power-Efficient Gathering in Sensor Information Systems
SI	Swarm Intelligence
SN	Sensor Node
TCP	Transmission Control Protocol
TDMA	Time-Division Multiple Access
TEEN	Threshold sensitive Energy Efficient sensor Network
UDP	User Datagram Protocol
WSN	Wireless Sensor Network

LIST OF SYMBOLS

$\mu_{\hat{A}}(x)$	The degree of membership function x
p	Probability of a node to become a CH in LEACH algorithm
r	The number of the current round in LEACH protocol.
$E_{Tx}(l, d)$	Energy consumed in transmitting an l -bit packet d meters.
$E_{Rx}(l)$	Energy consumed in receiving an l -bit packet.
E_{elec}	Electronics energy consumed in transmitting an l -bit packet
ϵ_{mp}	Energy dissipation value to run the amplifier for far distance (multipath)
ϵ_{fs}	Energy dissipation value to run the amplifier for close distance (free-space)
l_{ij}	Lifetime of the j -th CH for the i -th particle in PSO algorithm.
$e(0)_j$	Initial Energy of j -th CH
d_j	Euclidean distance between j -th CH and BS
r_j	Data Transmission rate for the j -th CH
$pBest$	Best solution it has achieved so far by each particle in PSO algorithm
$gBest$	Best value obtained so far by any particle in the swarm.
a_{j1}	Distance independent parameter in PSO algorithm.
a_{j2}	Distance dependent parameter in PSO algorithm.
c_1	Acceleration constant for moving the particles to $pBest$
c_2	Acceleration constant for moving the particles to $gBest$

CHAPTER I

Introduction

Wireless Sensor Network (WSN) is comprised of a collection of small size wireless sensor nodes that are low in cost, low in power consumption and also multifunctional [1,2]. Sensor nodes (SNs) collaborate together in order to achieve three common goals; monitoring the environment, processing data, and communicating wirelessly over short distances. Sensor nodes in WSN gather data about the conditions in which they are located and transform their data into electronic signals. The electronic signals are transmitted as radio waves to the base station (BS). In WSN the BS can be either a mobile or a fixed node that connects the sensor network to other types of network such as Internet or satellite where a reported data is accessible to the user [3]. One of the primary goals in designing WSNs is to prolong the lifetime of the network by reduction of energy consumption, as it is costly to change or replace exhausted batteries [4]. Routing is regarded as one of the most difficult issues in WSN because of the natural uniqueness that differentiate WSNs from other similar types of networks such as ad hoc or cellular wireless networks [5]. Global addressing scheme in WSN is almost impractical due to the large number of nodes, therefore traditional IP based protocols such as TCP and UDP are not applicable [6]. Also, SNs are very limited in terms of energy, processing, and storage capacities. Thus, they require precise resource management. Routing protocols in

WSN generally is divided into three different groups, Direct Communication (DC), Minimum Transmission Energy (MTE), and Cluster-based or Hierarchical routing protocols [7]. Cluster-based routing is mainly considered as a two-layer architecture where one layer is engaged in cluster head (CH) selection and the other layer is responsible for routing. A CH in hierarchical routing is a node that is responsible for gathering data from some of the other nodes in the cluster, aggregating this data and sending the aggregated data to the BS.

LEACH (Low Energy Adaptive Clustering Hierarchy) [7] is often considered as the most popular routing protocol that uses cluster-based routing in order to minimize energy consumption. In this thesis we propose an improvement on the LEACH protocol called FLMBBS that is capable of outperforming the LEACH protocol in terms of minimizing energy consumption.

1.1 Motivation

Wireless Sensor Networks is an active area of research in computer science and telecommunication today. WSNs enjoy widespread usage in both social and industrial settings. They are currently being employed in a variety of applications ranging from military surveillance to medical monitoring. For example WSN is used to enhance the military surveillance system by alarming military commands and also by controlling events of interest in hostile areas [8]. The events of interest could be the occurrence of suspicious activities, armed individuals, and moving vehicles or other defined activities. WSN must enquire the present position of an object with acceptable confidence to successfully detect, track and

classify the collected data. Furthermore, WSNs have been applied to many health care applications including controlling patient's health status, diagnostics, and detection of doctors and patients and also drug administration in hospitals [9]. Due to important nature of these applications, a high degree of precision is required during the design stage of the applications. Nonetheless, there are many challenges that are yet to be resolved. One of the most apparent challenges would be designing an energy efficient WSNs that has recently been given much attention. Since sensor nodes in WSN applications are operated through use of battery life, there are requirements for innovative procedures that would terminate roots of energy inefficiencies that result in reduction of network life time.

1.2 Objective and Scope

The primary objective of this thesis is to improve upon the LEACH protocol [7] by purposing a novel algorithm in which the BS is mobile and adjusts its position according to the needs of the CHs. This shortens the CH-to-BS transmission distances, thereby minimizing the energy consumption of the CHs, which constitute the majority of the energy consumption of the entire network. Hence the network's lifetime is extended and this is equivalent to increasing the energy efficiency of the WSN. This research was encouraged by the fact that LEACH occupies an incredibly important position in the area of wireless sensor networks [10]. The majority of the hierarchical routing algorithms, planning to prolong network lifetime, have been derived from LEACH.

In WSNs, sensing the environment, data computation and wireless communication are the three major events that consume power. Lowering the communication cost is considered as the primary concern while improving the energy cost of sensing and computation events are secondary concerns in designing such networks. Thus the cost of data communication is greater than the cost of sensing and computing the data in WSNs [11]. Hence, the primary objective of the current work is to minimizing the communication distance by finding the optimal position of BS and moving the BS towards it in order to minimize the communication cost. In hierarchical or cluster-based architecture, CHs consume more energy than the other nodes in the network. This is due to the fact that CHs deliver the task of communicating with the members within their clusters, gathering and aggregating the data from those members and sending the aggregated data to the BS. Usually the BS is located far from the nodes and sending data to it is very energy consuming as transmission distance is a prominent factor in the energy consumption of communication.

1.3 Thesis Contribution

The major contribution of this thesis can be structured as follows:

- In this thesis we propose a fuzzy-logic-driven approach for moving the BS within the LEACH protocol. This involves 2 parts:
 1. We present a fuzzy inference system that assigns a priority value to each CH based on that CH's residual energy, distance to BS, and data transmission rate.

2. Using the calculated priority values, the BS adjusts its position relative to all the CHs.
- We name the LEACH protocol modified with our above approach FLMBBS. We compare the performance of FLMBBS with that of the original LEACH protocol, in terms of overall network lifetime and also in terms of CH energy dissipation. From our experimental results, we can see that FLMBBS is capable of outperforming LEACH.
 - We also compare our proposed protocol with another protocol that is a modified version of LEACH using a mobile-BS strategy. This other protocol makes use of particle swarm optimization (PSO).

1.4 Thesis Outline

The subsequent parts of the thesis are organized as follows:

CHAPTER 2: this chapter starts with an overview of WSNs and Cluster based architecture; next, particle swarm optimization is introduced and how it optimizes the BS location is explained. Finally, the fuzzy logic system is presented followed by review of an example from literature related to our work.

CHAPTER 3: this chapter presents in detail the fuzzy inference system and the movement strategy in the proposed FLMBBS protocol. FIS design for the proposed protocol, and input and

output parameters are explained. An example of the FIS calculation of CH priority is given, followed by an example of our movement strategy.

CHAPTER 4: this chapter presents the results from two groups of simulation experiments involving the proposed protocol. In the first group of experiments, the performance in terms of network lifetime and CH energy consumption, of FLMBS is compared with that of the LEACH protocol under different scenarios. In second group of experiments the performance of FLMBS is compared to that of PSO.

CHAPTER 5: this chapter concludes the thesis by summarizing the main ideas and some directions for future research.

CHAPTER II

Background and Related work

This Chapter describes relevant background knowledge and related work for readers to easily understand the proposed protocol and the methodology and analysis of our experiments, which are to be presented and discussed in Chapter 3 and 4. In Section 2.1, wireless sensor networks and their significant characteristics are discussed. Section 2.2 presents swarm intelligence and how the optimal position of the base station is obtained using the particle swarm optimization method. Finally, section 2.3 addresses Fuzzy Logic, as well as the structure and operations of Fuzzy Inference Systems.

2.1 Wireless Sensor Network

Wireless sensor network consists of large number of wireless sensor nodes located over a geographic area. The “wireless sensor node” term is for devices that use low power and are equipped with one or more sensors, a radio unit, power supply, processor and an optional actuator. The sensor node can have sensors for the detection and measurement of thermal, mechanical, optical, magnetic, chemical or biological signals. In a basic WSN, the integrated

radio unit in a sensor node sends the data collected to the base station. The base station is normally located far from the sensor nodes and acts as a gateway between the network and subsequent communication centers.

A general structure of WSN is presented in Figure 2.1. The WSN can be structured or unstructured [12]. A basic wireless sensor network requires very little infrastructure. In one such network, nodes can be deployed in an ad hoc fashion. The network is not attended after deployment and does monitoring and reporting on its own. However, the sensor network deployed to obtain data from the environment may require a large number of sensor nodes, numbering thousands to tens of thousands depending on the area to be covered. Due to large number of nodes the management of network becomes difficult and complex structure is required. The structured wireless sensor network has planned deployment of sensor nodes, and this means that fewer nodes are required to cover the area compared to an unstructured network. Cost of maintenance and management is reduced.

The wireless sensor network nodes have limitations in terms of limited power available for working, low bandwidth, limited processing capabilities, small range and limited data storage. The network design is based on the environment of operation. Thus, network topologies, the schemes of deployment are decided on a case-to-case basis. Normally, small numbers of nodes are sufficient for indoor coverage whereas outdoor coverage requires large numbers of nodes. For inaccessible areas only ad hoc deployment is used. Ad hoc deployment is also used when the number of nodes ranges from 100s to 1000s.

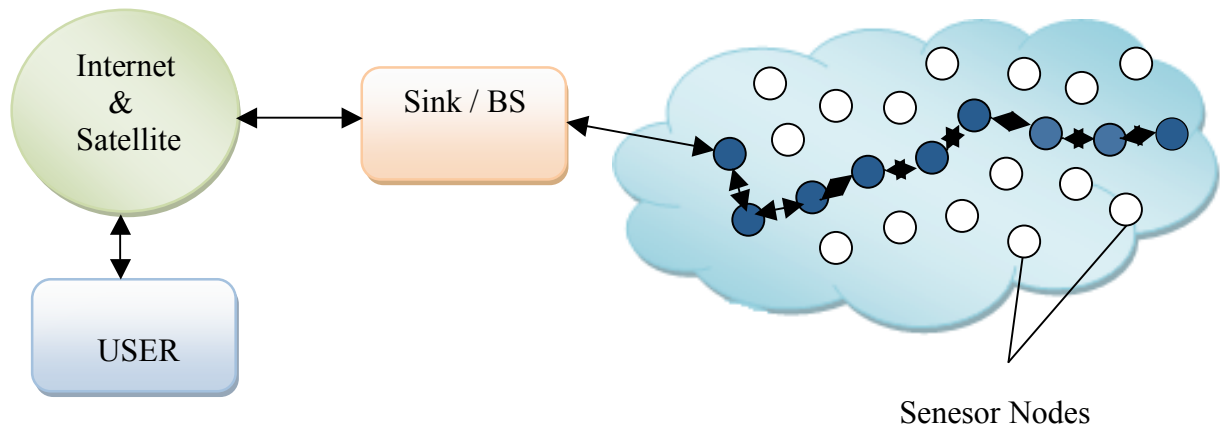


Figure 2.1: WSN overview [13]

The protocol stack of sensor network is extremely similar to the protocol stack of the traditional ad-hoc networks, with the following layers: Application, Transport, Network, Data Link, and Physical [14].

- *Application layer*: The application layer is responsible for user interface and data processing.
- *Transport layer*: This layer specifies the methodology for reliable packet transportation.
- *Network layer*: The network layer's function is to take care of addressing and forwarding packets.
- *Data link layer*: The data link layer's function is data streams multiplexing, error control, frame detection and ensuring reliable connections.
- *Physical layer*: The physical layer functions are to define frequency in use, signal characteristics such as modulation scheme and encryption.

2.1.1 WSN Hardware

The general architecture and the major components of a wireless sensor device (node) are illustrated in Figure 2.2. A wireless sensor device is generally composed of four basic components: a *sensing unit*, a *processing unit*, a *transceiver unit* and a *power unit* usually in the form a battery.

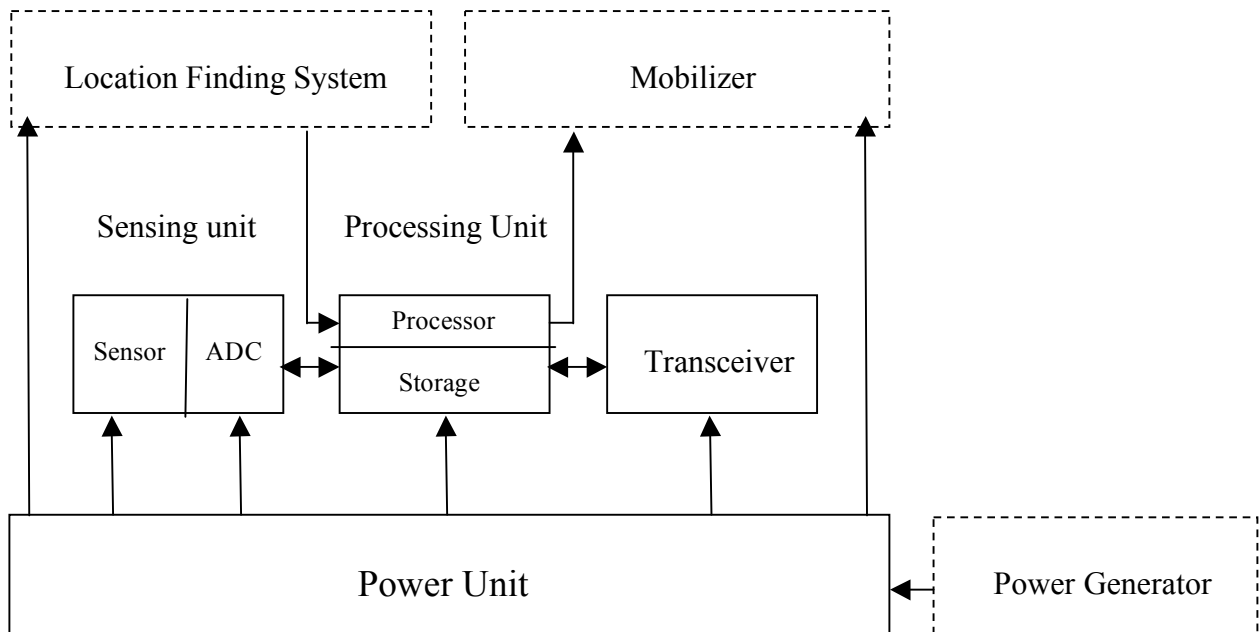


Figure 2.2: General hardware architecture of a sensor node [15]

Each sensing unit comprises of sensor(s) for sensing environment and analog-to-digital converter (ADC) [15]. Nodes transmit their sensed data if certain pre-defined conditions are met. The environmental signal is received in the form of an analog signal by the sensor and

then is converted into a digital signal by the ADC. The Processing unit consists of a microcontroller or in some applications a microprocessor and is responsible for analyzing the attributes of the sensed data by using digital signals. The Transceiver is for connecting the nodes and the BS through a radio transmitter. Lastly, the power unit is usually a battery.

Based on different applications, there might be extra components such as localization unit, energy producer, position changer, etc. These components are shown in Figure 2.2 by the dashed boxes.

2.1.2 WSN routing protocols

Many new energy saving protocols distinctively designed for sensor networks, are results of the recent advancements in WSN. Wireless communication is considered the primary component of energy consumption in WSN [16, 17]. So particular attention was given to the routing protocols, which can vary contingent on the application and network architecture.

The routing protocols in WSNs are broken down into three categories. First, direct communication (DC), which is the simplest protocol, where sensor nodes send data directly to the BS. The second category involve Minimum Transmission Energy (MTE) protocols, where nodes route data to the base station through intermediate nodes, each node acting as a router for the other nodes. The third and perhaps most interesting category are made up of clustering protocols. Hierarchical or cluster-based routing, originally presented in wire-line networks, are recognized techniques with particular advantages related to scalability and efficient

communication. Cluster-based routing has been shown to be more effective than DC and MTE and is hence focused on in this work.

2.1.3 Cluster-based Wireless Sensor Networks

In WSN, some of the issues are very critical and complex, such as energy-efficient operation, latency, channel disputation and management. In particular, in large-scale sensor networks, the nodes that reside far away from the BS either have to count on significant amount of intermediate nodes or use high transmission power to forward their sensed data. Various algorithms suggest solutions to the above issue, which is based on the decomposition of the entire network into smaller groups called clusters [18, 19, 20, 21, 22, 23, and 24]. In general, neighboring nodes are grouped into the same cluster and a cluster-head as shown in Figure 2.3, manages each cluster.

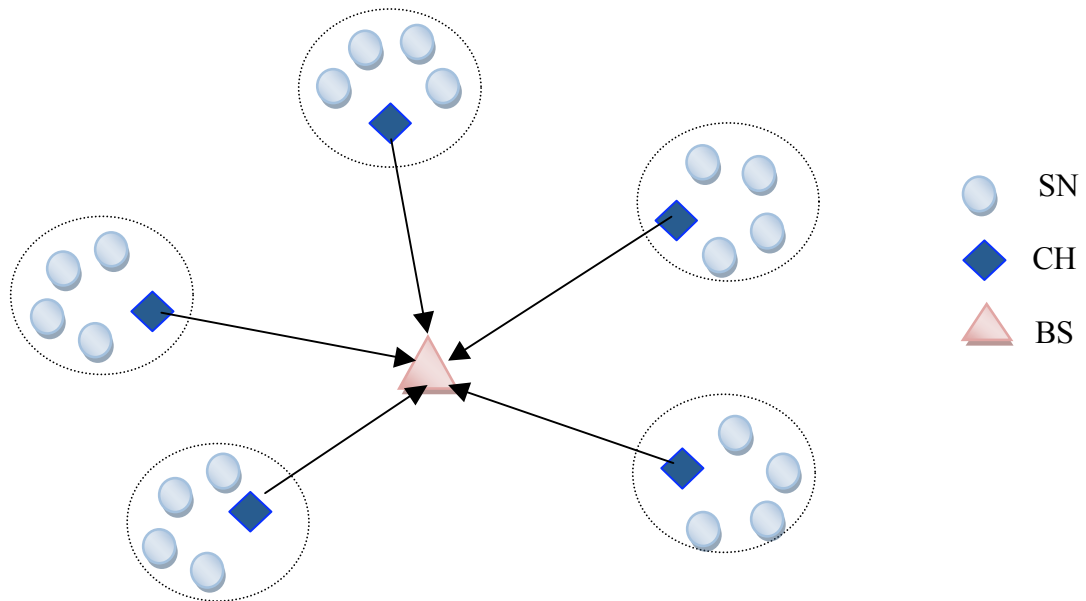


Figure 2.3: Cluster based architecture for WSN

The cluster-head acts as a local base-station and it is responsible for collecting the sensed data from the member nodes of that cluster. The cluster head forwards the collected data to the other cluster-heads or directly to the sink/BS. Communication among cluster-heads can be via either single or multi hops. The cluster-heads are responsible for managing both inter-cluster and intra-cluster communication [25].

Clustering has advantages and disadvantages. Clusters can decrease the power consumption of a WSN, thus boosting the lifetime of the network. Nodes inside a cluster are only required to broadcast to its cluster-head, and this decreases each node's connection variety. This also permits the spatial reuse of communication channels while decreasing collisions. By aggregating data, the number of messages that flow through the network can be lowered. Another important feature of clustering is the rotation cluster-head roles among the sensor nodes in order to not drain the battery of a single node (as the CH consumes the most energy among all nodes in a cluster).

Cluster-head selection can be based on different parameters such as node ID [26], node degree [27], residual energy [28], or probabilistically methods [29, 7]. One of the simplest selection methods is Max-Min d-clustering [23], but this solution is not directly applicable because it is not energy aware. On the other hand, LEACH [7] is a well-known cluster-based solution that is relatively simple to implement yet achieves longer network life time by selecting cluster-heads based on residual energy of the nodes and data aggregation.

2.1.4 LEACH: Low-Energy Adaptive Clustering Hierarchy for Wireless Micro-sensor Networks

Heinzelman et al. [7], proposed a new adaptive clustering algorithm in which the nodes organize themselves into clusters using a distributed algorithm. The basic idea of LEACH is to create clusters of sensor nodes based on the strength of the received signals and use the cluster heads as routers to the BS. Since data communication to the BS is the primary source of the energy consumption, the roles of the cluster-heads rotate among the sensor nodes. This strategy addresses the problem of traditional clustering protocols in which cluster heads are preset during the network lifetime. The operation of LEACH is broken down into rounds and each round consists of a *setup phase* and a *steady state phase*.

In the setup phase, the clusters are organized and CHs are selected. At the beginning of a given round, each node selects a random number between 0 and 1 and compares it to the threshold $T(n)$ shown in formula (2.1) and if the number is lower than a threshold, the node becomes a cluster head.

$$T(n) = \begin{cases} \frac{p}{1 - p(r \bmod \frac{1}{p})} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases} \quad (2.1)$$

Where p is the desired percentage of cluster heads, r is the current round, and G is the set of nodes that have not been cluster heads in the last $1/p$ rounds. At the beginning of each round, selected CHs broadcast an advertisement message to all the nodes in the network, informing their new status. After receiving this message, each of the non-CH nodes can determine to which cluster they belong to based on the strength of the received signal. Then, according to the number of nodes in a given cluster, that cluster's CH generates a TDMA schedule, and broadcasts a transmission time window to its member nodes.

Now the steady state phase begins. Nodes in each cluster can start sensing the data and transmitting sensed data to their own CH during the allocated transmission time. The CH node conducts the data fusion, aggregating, compressing and then sending the aggregated data to the BS. Since the BS is usually far away from the field, communicating to the BS will consume plenty of the CH's energy. Once the allocated transmission time is over, the steady state phase ends and the network goes back into the setup phase and starts another round, beginning with selection of new CH's. The radio hardware dissipation model assumed in LEACH [30] is a simple radio model where the transmitter dissipate energy to run the radio electronics and the power amplifier, and the receiver dissipates energy to run the radio electronics, as shown in Figure 2.4.

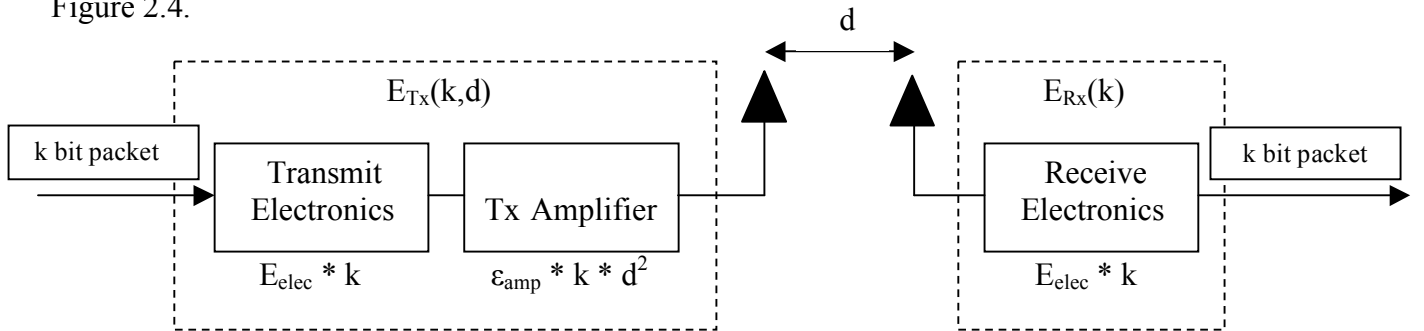


Figure 2.4: Radio energy dissipation model [30].

In their approach they use two different radio models: The free space model and the multi-path fading channel model. When the distance between the transmitter and receiver is less than threshold value d_0 , the algorithm adopts the free space (*fs*) model (d^2 power loss). Otherwise the algorithm adopts the multi-path (*mp*) fading channel model (d^4 power loss). Thus, to transmit an l -bit message a distance d , the radio expands

$$E_{Tx}(l, d) = \begin{cases} lE_{elec} + l\epsilon_{fs}d^2 & d < d_0 \\ lE_{elec} + l\epsilon_{mp}d^4 & d \geq d_0 \end{cases} \quad (2.3)$$

and to receive this message, the radio expends:

$$E_{Rx}(l) = E_{Rx} - elec(l) = lE_{elec}. \quad (2.4)$$

Where l in here represents the number of bits, E_{elec} is the energy dissipation to run the radio electronics, ϵ_{fs} and ϵ_{amp} are the energy dissipation values to run the amplifier for close and far distances respectively.

2.2 Swarm Intelligence

Swarm intelligence (SI) [31] is identified as the group of intelligence that derived from a collection of simple units, generally named agents. These units enter into some type of interactions with each other, sense the data and change their environment locally. Moreover, they exhibit complicated, evolving performance that is robust with respect to the failure of individual units. The majority of SI models are inspired from swarm of bees, flocks of birds, termites, ant colonies, or fish schools.

In computational intelligence, there are two well-known swarm inspired techniques: Ant colony optimization (ACO) [32, 33] and particle swarm optimization (PSO) [34, 35]. ACO was applied successfully in many discrete optimization problems motivated by the behaviors of ants, however we are not considering this algorithm in this thesis. The particles swarm concept is described in the next subsection and it is implemented in our experiments in chapter 4.

2.2.1 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is an metaheuristic algorithm which was purposed by Dr. Russell Eberhart and Dr. James Kennedy in 1995 [34, 35]. PSO was encouraged by social behavior of bird flocking or fish schooling, which use SI to solve an optimization problem by iteratively trying to improve a candidate solution. PSO creates a population of simple agents called particles. These particles interact with one another and the environment, while learning

from their own experiences. Usually there is no centralized control system dictating how individual particle will behave. It is the local interactions between particles that lead to the emergence of global behavior.

The PSO, can be effectively applied as an optimization method in n-dimensional space simulations although it was originally designed for simulating the movement of the swarm in 2-dimensional space [36]. In PSO each particle has its own position x and velocity v . Particles flow through the search space, and are evaluated according to some fitness criterion after each iteration. In every iteration, each particle in the swarm updates its status by following the two “best” position values, i.e. values that give the greatest fitness. The first position value is the value giving rise to the best fitness that particle has achieved so far ($pBest$), and the second is the value that gives rise to the best fitness obtained so far by any particle in the population ($gBest$).

The velocity of each particle, which is influenced directly by $pBest$ and $gBest$ values, is evaluated through the following equation:

$$V_{id}^{new} = w \times V_{id}^{old} + c_1 \times Rand_1() \times (pBest_{id} - x_{id}) + c_2 \times Rand_2() \times (gBest_{id} - x_{id}) \quad (2.5)$$

In the above equation, V_{id}^{old} is a numerical representation of the current velocity of the particle at the d -th dimension. Also V_{id}^{new} presents the new velocity derived for the i -th particle at the d -th dimension. w , which is the inertia weight, starts a global search if the inertia weight is large and if the inertia weight is small, it only facilitates a smaller local search. c_1 and c_2 are two constant numbers used to represent the acceleration of the particles movement to $pBest$ and

gBest respectively. $Rand_1()$ and $Rand_2()$ are two randomly generated numbers that must range from 0 to 1. And finally x_{id} states the current position of the i -th particle. The new position of the particle (x_{id}^{new}) is derived, by adding the new velocity calculated (V_{id}^{new}) and the current position of the particle x_{id}^{old} .

$$x_{id}^{new} = x_{id}^{old} + V_{id}^{new} \quad (2.6)$$

Next section explains the PSO algorithm in detail and shows how this algorithm optimizes the problem by updating the particle's state.

2.2.2 Particle Swarm Optimization Algorithm

Let S be the number of particles in the swarm, each having a position x and velocity v in the search space the PSO algorithm is as follows [34]:

Algorithm 2.1: PSO Algorithm

- For each particle $i=1, \dots, S$ do:
 - randomly initialize the particle's position x .
 - initialize the particle's best know position(pBest) to its initial position.
 - If $f(gBest)_i < f(pBest)_i$ update the swarm's best know position. $gBest_i \leftarrow pBest_i$

- Initialize the particle's velocity v .
 - Until the termination condition is reached repeat:
 - Generate random number between 0 to 1 for $\text{Rand}_1()$ and $\text{Rand}_2()$.
 - Update the particle's velocity based on Equation (2.5).
 - Update the particle's position based on Equation (2.6).
 - If $f(\text{pBest}_i) < f(x_i)$
 - Update the particle's best know position $\text{pBest}_i \leftarrow x_i$
 - If $f(\text{gBest})_i < f(\text{pBest}_i)_i$ update the swarm's best know position. $\text{gBest}_i \leftarrow \text{pBest}_i$
 - Now gBest holds the best found solution
-

2.2.3 Related Work

Hong et al. [37], attempted to minimize power consumption in a cluster-based routing protocol by making the BS mobile, in order to prolong the network lifetime. The protocol they considered involved fixed clusters and fixed CHs. In this setup, the energy levels and data transmission rates of the CHs might be different. Only CH's energy was considered. Once a CH depletes its energy the whole network shuts down.

The authors used a PSO algorithm to search for the optimal location of BS. Their algorithm starts by randomly generating a set of particles and allocating each particle with an initial velocity for changing its state. Each particle represents a potential BS location. After deploying the particles in the search space, lifetime of each CH for each particle position was calculated

(by treating that particle as the BS). They used the following equation in order to calculate the lifetime l_{ij} of the j -th CH for the i -th particles.

$$l_{ij} = e_j(0) / r_j(a_{j1} + a_{j2}d_{ij}) \quad (2.7)$$

In the above equation $e_j(0)$ and r_j are representing the initial energy and data transmission rate for the j -th CH respectively. a_{j1} and a_{j2} are a independent and dependent distance parameters respectively. d_{ij} is also for representation of the calculated Euclidean distance from the i -th particle to the j -th CH. Finally the fitness value for the i -th particle is calculated as follows:

$$fitness(i) = \underset{j=1}{\overset{m}{Min}} l_{ij} \quad (2.8)$$

In calculation of fitness value, m , stands for the total number of CHs. In every iteration, the particle with the minimum lifetime is chosen as the fitness value for the current round. The lifetime becomes longer as soon as the fitness value become larger and this means the corresponding BS position is getting better. At the end of every iteration the corresponding $pBest$ value of each particle is compared with the fitness value of each particle for the current round. If the $pBest_i$ value of the particles is less than the fitness value of that particle, $pBest_i$ value is updated with the current fitness value. The $gBest$ value is then derived from selecting the best $pBest_i$ among all the particles. Based on the velocity calculated by equation (2.2), which we explained earlier, the particles change their states and they also update their location based on the equation (2.3). When the termination conditions are met, the final $gBest$ presents

the optimal location of BS that is achieved so far. The following algorithm represents their proposed approach.

Algorithm 2.2: Finding the optimal location of BS using PSO

INPUT: a set of CHs, each CH_j with its location (x_j, y_j) , data transmission rate r_j , initial energy $e_j(0)$, parameters a_{j1} and a_{j2} ,

OUTPUT: a BS location that will cause a nearly maximal lifetime in the whole system.

- Initialize the fitness value of all $pBest$ and $gBest$ to zero.
 - Randomly generate a group of n particles in a two-dimensional space.
 - Randomly generating an initial velocity for each particle.
 - Calculate the lifetime l_{ij} of the j -th CH for the i -th particle by using equation (2.4)
 - Calculate the lifetime of the whole sensor network for the i -th particle as its fitness value($fitness_i$) by using equation (2.5)
 - Set $pBest_i$ as the current i -th particle if the value of $fitness(i)$ is larger than the current fitness value of $pBest_i$.
 - Get the $gBest$ as the best $pBest$ among all the particles.
 - $fitness\ of\ pBest_k = \max_{i=1}^n fitness\ of\ pBest_i$
 - and set $gBest = pBest_k$
 - Update the velocity of the i -th particle by using equation (2.2).
 - Update the position of the i -th particle by using equation (2.3).
 - Repeat steps until the termination conditions are satisfied.
-

2.3 Fuzzy Logic

Fuzzy logic (FL) is a type of many-valued logic that has been used in sensor networks and has contributed to improvements in network efficiency. FL deals with reasoning that is inexact rather than fixed and precise. Professor Lotfi Zadeh introduced the concept of FL in the mid 1960's. [38, 39, 40]. Fuzzy logic has introduced many methods for representing and inferring from uncertain or incomplete knowledge. Fuzzy concepts cannot be modeled by a simple set inclusion operator \in , but there is a degree of membership. In other words, while classical Boolean logic has just two values, true (always presented numerically as 1) or YES, and false (always presented numerically as 0), or NO, Fuzzy Logic extends these two values to values between 0 and 1 using the concept of degrees of membership.

In the following subsections, we explain crisp and fuzzy sets, properties of fuzzy sets, operations on fuzzy sets, linguistic variables, fuzzy IF-THEN rules, fuzzy inference systems, and followed by an example from literature of a fuzzy inference system at work.

2.3.1 Crisp and Fuzzy Sets

Let A be a *crisp set* defined over the universe of discourse X . Then for any element x in X , either x is a member of A or not. Crisp sets are also known as classical sets. Universe of discourse is set of all possible elements that can come into consideration.

The membership function f_A of an element x for a crisp set A is defined as follows:

$$f_A(x) : \rightarrow \{0,1\},$$

$$f_A(x) = \begin{cases} 1 & \text{if } x \in A, \\ 0 & \text{otherwise} \end{cases}$$

Therefore, if x belongs to the set A , for any element x of universe of discourse X , membership function $f_A(x)$ is equal to 1, and if x is not a member of set A , then it is equal to 0.

In contrast to Crisp sets, the elements of the *fuzzy sets* belong to a subordinate fuzzy set with a specific degree of membership. [41]. For any element x of universe X , if x belongs to set A , membership function $\mu_A(x)$ equals the degree to which x belongs to set A as shown in Figure 2.5. If x is not a member of set A the membership function $\mu_A(x)$ is equal to zero. The main difference between crisp sets and fuzzy set is that the elements in fuzzy sets can have partial membership with respect to a set. The membership function $\mu_A(x)$ of an element x for a fuzzy set A is defined as follows:

$$\mu_A(x) : X \rightarrow [0,1], \text{ where } \begin{array}{ll} \mu_A(x) = 1 & \text{if } x \text{ is a full member of set } A \\ \mu_A(x) = 0 & \text{if } x \text{ is not a member of set } A \\ 0 < \mu_A(x) < 1 & \text{if } x \text{ is partial member of set } A \end{array}$$

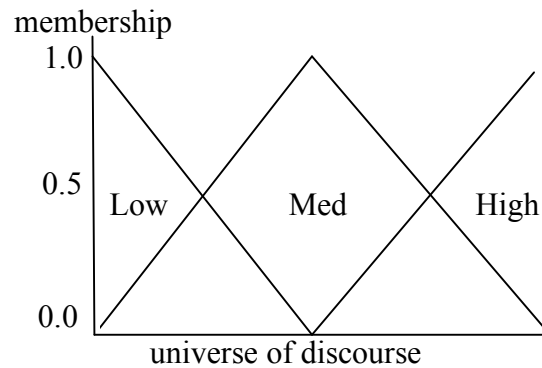


Figure 2.5: A basic fuzzy set

The elements in fuzzy sets can have an inclusive degree of membership that ranges from 0 to 1. In Figure 2.5 the membership functions (MFs) are illustrated by a triangular function but Gaussian, Sigmoid and other types of linear functions can also be applied to characterize the fuzzy sets [42]. Non-linear functions can also be used but it will increase the computational complexity of the algorithm.

2.3.2 Properties of Fuzzy Sets

Fuzzy sets and crisp sets both share similar properties. Classical or crisp sets is a unique case of fuzzy sets in which membership values are a subset of the interval [0,1], The following rules, which are common in crisp set theory, also apply to fuzzy set theory. In here \tilde{A}, \tilde{B} and \tilde{C} represents three random fuzzy sets [43].

$$\begin{aligned} \text{Commutativity: } \tilde{A} \cup \tilde{B} &= \tilde{B} \cup \tilde{A} \\ \tilde{A} \cap \tilde{B} &= \tilde{B} \cap \tilde{A} \end{aligned}$$

$$\begin{aligned} \text{Associativity: } \tilde{A} \cup (\tilde{B} \cup \tilde{C}) &= (\tilde{A} \cup \tilde{B}) \cup \tilde{C} \\ \tilde{A} \cap (\tilde{B} \cap \tilde{C}) &= (\tilde{A} \cap \tilde{B}) \cap \tilde{C} \end{aligned}$$

$$\text{Distributivity: } \tilde{A} \cup (\tilde{B} \cap \tilde{C}) = (\tilde{A} \cup \tilde{B}) \cap (\tilde{A} \cup \tilde{C})$$

$$\begin{aligned} \text{Idempotency: } \tilde{A} \cup \tilde{A} &= \tilde{A} \\ \tilde{A} \cap \tilde{A} &= \tilde{A} \end{aligned}$$

$$\text{identity} : \tilde{A} \cup \emptyset = \tilde{A} \quad \tilde{A} \cap X = \tilde{A}$$

$$\tilde{A} \cap \emptyset = \emptyset \quad \tilde{A} \cup X = X$$

$$\text{Transitivity} : \tilde{A} \subseteq \tilde{B} \subseteq \tilde{C} \quad \text{Then} \quad \tilde{A} \subseteq \tilde{C}$$

2.3.3 Operations on Fuzzy Sets

There are three major operations on fuzzy sets, which are complement, intersection, and union.

Figure 2.6 is a visual presentation of the different operations performed on fuzzy sets. Let \tilde{A}, \tilde{B} be two fuzzy sets defined on the universe of discourse X . for a given element x of the universe, the function-theoretic operations of union, intersection, and complements are defined as follows:

$$\text{Union} : \mu_{\tilde{A} \cup \tilde{B}}(x) = \mu_{\tilde{A}}(x) \vee \mu_{\tilde{B}}(x)$$

$$\text{Intersection} : \mu_{\tilde{A} \cap \tilde{B}}(x) = \mu_{\tilde{A}}(x) \wedge \mu_{\tilde{B}}(x)$$

$$\text{Complement} : \mu_{\tilde{A}}(x) = 1 - \mu_{\tilde{A}}(x)$$

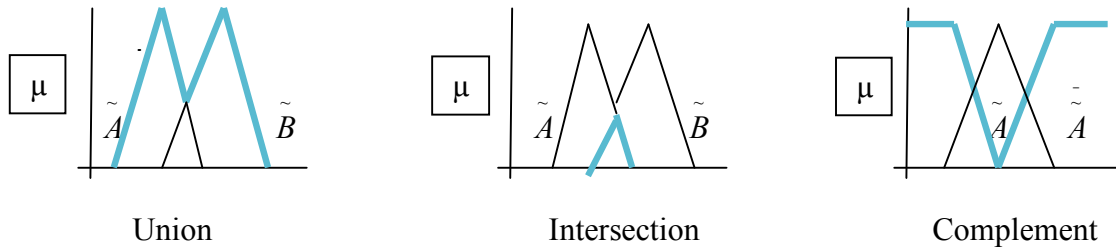


Figure 2.6: Fuzzy Sets Operations

Any fuzzy set \tilde{A} defined on the universe X is a subset of the universe. Also by definition the null set has membership 0 and x in X has membership 1. Note that the null set and the whole set are not fuzzy sets.

2.3.4 Linguistic Variables and Linguistic Values

A linguistic variable is a fuzzy variable [44]. Though variables in mathematics normally deals with numerical values, in fuzzy logic applications, non-numeric linguistic variables are usually used to facilitate the expression of rules and facts. The values of linguistic variables, or simply linguistic values, are words or sentences in a natural or artificial language [45]. For example, “distance is far” implies that the linguistic variable *distance* takes the linguistic value *far*.

2.3.5 Fuzzy IF-THEN rules

Once the linguistic variables and values are defined, the rules of the fuzzy system can be formulated. These rules map the fuzzy inputs to fuzzy outputs. Fuzzy rules, or IF-THEN rules, are statement(s) that consists of three parts: antecedent, fuzzy proposition and consequence(s). One antecedent may contain more than one of the (AND) or (OR) operators. We can express the fuzzy IF-THEN rule in the following statement:

if x is A and/or y is B ***then*** z is C

Where A, B, and C are the linguistic values, while x, y, and z are the linguistic variables.

2.3.6 Fuzzy Inference System

A Fuzzy Inference System (FIS) can be defined as a process of mapping from a given input to an output, using the theory of fuzzy sets. The process of FIS involves all of the pieces that have been described in the previous subsections: Membership Functions, Logical Operations, and IF-THEN Rules. The fundamental FIS structure is composed of four functional blocks as shown in Figure 2.7.

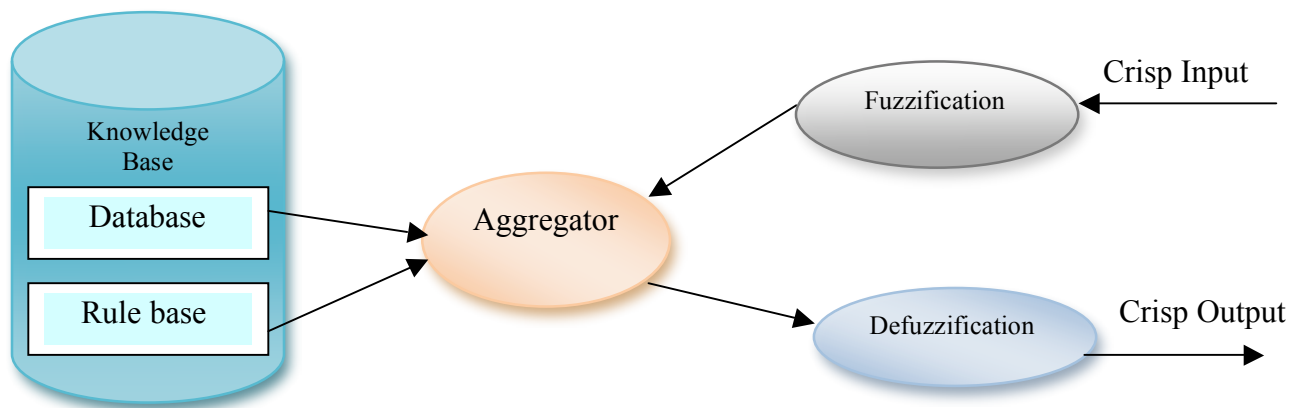


Figure 2.7: FIS structure [46]

- Fuzzification: converts the crisp input to the fuzzy sets.
- Aggregator: performs the inference operation based on the rules.
- Knowledge-base: 1. Database : defines the MFs
2. Rulebase : IF-THEN rules.

- Defuzzification: converts the fuzzy sets into a crisp value by using inference engine

The Mamdani [44] and Sugeno [46] are two popular types of FIS that have been successfully used in many applications. The difference between these two types of FIS lies in the aggregation of the output(s) and defuzzification. Some of the applications of the FIS are data classification, decision analysis, expert systems, times series predictions, robotics & pattern recognition. In this thesis we decided to use Mamdani fuzzy inference method, which is consider as the most commonly seen fuzzy methodology due to its simple structure of ‘min-max’ or “AND-OR” operations which we are going to explain in Step 3 and Step 4. The idea was purposed in 1975, by professor Ebrahim Mamdani at the university of London [44]. The procedure for implementing Mamdani-style fuzzy inference process is performed in four steps: fuzzification, rule evaluation, aggregate output(s), and finally defuzzification.

Step 1: Fuzzification:

Fuzzification comprises the process of transforming crisp input values into grades of membership function for linguistic values of fuzzy sets. The membership function is used to associate a grade to each linguistic value.

Step 2: Rule evaluation

Once that the input and output variables and the corresponding MFs are defined, we need to

design the rule- base of the fuzzy knowledge-base. The rule-base of FIS is composed of expert IF <antecedents> THEN <conclusion> rules. These rules transform the input variables to an output that informs of the projected outcomes. The number of rules depends on the number of linguistic variables and MFs. In Mamdani, this step is done by applying the AND operation on each rule.

Step 3: Aggregate output(s)

Because all decisions in FIS are dependent on examination of all the rules within that FIS, the Rules need to be bundled in a particular manner to make a decision. Aggregation method bundles the fuzzy sets represented by the output of each rule into a single fuzzy set. In Mamdani this aggregation is done by applying the OR operation on the output fuzzy sets.

Step 4: Defuzzification

Defuzzification is considered as the final step in the fuzzy inference process. This step involves generation of a crisp number from the single fuzzy set output founded by the aggregation method in Step 3. Centroid is considered as one of the most popular defuzzification methods [47]. Centroid defuzzification method is used to determine the point, which represents the center of gravity (COG) of the fuzzy set. The COG is a method of calculation that uses each subdivided area's center and assigns a particular weight based on the overall contribution to the whole to generate a weighted average point. The COG method is calculated as follows:

$$COG = \int \mu_A(x) * x dx / \int \mu_A(x) dx \quad (2.9)$$

Where, \int denotes an algebraic integration, and $\mu_a(x)$ is degree of membership function of set A

For completeness, we list the five defuzzification methods that are provided in the fuzzy logic toolbox [42]:

1. Centroid: centroid of area
2. Bisector: bisector of area
3. Mom: mean value of maximum.
4. Som: smallest (absolute) value of maximum.
5. Lom: largest (absolute) value of maximum.

2.3.7 Related Work

Ran et al. [48], improved the LEACH protocol by modifying the setup phase. In every round during the setup phase, instead of each node being given a random probability of becoming a CH as in LEACH, here each node's probability of becoming a CH is calculated using a FIS. The FIS takes 3 parameters as inputs (and thus 3 groups of membership functions): distance between node and BS, battery level of node, and node density (i.e. how many nodes are in a particular node's immediate environment). The output of the FIS is probability (of becoming CH). The FIS used was Mamdani-type, as the author identified it to be the simplest and most

commonly used. A total of 27 if-then rules were used, all gauged the following equation, which placed the most importance on remaining battery level:

$$\text{Probability} = \text{Batterylevel} * 2 + \text{nodedensity} + (2 - \text{distance}) \quad (2.10)$$

For example, one particular rule was: if battery level is low, and node density is medium, and distance is far, then probability is weak. The authors also specified the centroid method as the one used for defuzzification, in order to get the crisp value out of the aggregation obtained by the rules. Their simulation results showed that the FIS driven CH selection led to greater energy efficiency than in LEACH. Energy consumption was lower each round, and both the first node death and the last node death occurred later than in LEACH. Our model also relies on FIS heavily, but in every round instead of improving CH selection, we improved BS location.

CHAPTER III

Improving LEACH Protocol Using Fuzzy Logic Approach

In this chapter we present the design details of our model for a mobile BS. The sections of this chapter are organized as follows: in section 3.1 we describe the FLMBBS protocol and follow it with an explanation of the FIS design in section 3.2, all input and output parameters are discussed in section 3.3, a step by step method to calculate the priority of the CHs by using our FIS model with a simple example to clarify our design system work is discussed in section 3.4. Finally, we provide an example of the calculations for BS movement in 3.5.

3.1 FLMBBS Protocol Overview

In this section, the proposed FLMBBS protocol is presented in detail. Our protocol is largely identical to LEACH with the one important difference: after the setup phase where CH's are chosen and before the steady state phase where transmission takes place, the BS changes its location according to the locations and properties of the CH's selected in the current round. The movement strategy involves the usage of a Fuzzy Inference System (FIS), which in general is a system that uses fuzzy logic to map given inputs to output(s). The steps of our movement strategy are summarized as follows:

- Using a proposed FIS, a priority value is calculated for each CH, based on that CH's distance from the initial BS location, its residual energy and its firing rate. The details of the FIS are discussed in sections 3.2 to 3.4. In general, priority increases with distance and firing rate, while it decreases with residual energy.
- The BS moves towards each CH by a distance proportional to that CH's priority. This is accomplished by setting up a vector between the BS and each CH, with the vector's magnitude equal to that CH's priority multiplied by a constant step size (preset value, constant throughout one run of protocol). Subsequently all these BS-to-CH vectors are added together to obtain a *net* movement vector. This vector is then added to the current BS location to obtain the final BS location in this round. An example of BS movement is given in section 3.5.

Note that at the beginning of each round, the BS is moved back to its initial location (usually center of network area), since different CH's are chosen in different rounds. Also, before moving the BS, if the minimum CH-to-BS distance is smaller than the step size, then that minimum distance is taken as the step size for that round.

With our movement strategy, we essentially aimed to lower the transmission distance between the BS and each CH thus lowering energy dissipation of the CHs, with CHs in greater "urgency" benefiting more. Since CH energy dissipation represents a major portion of the total energy loss by the nodes, we expected our protocol to have improvement over LEACH in terms of energy dissipation within each round and also improvement in terms of overall network lifetime.

3.2 FIS Design

As noted previously in Chapter 2 section 2.3, the FIS structure is formed by four steps: fuzzification, knowledge base, aggregator, and defuzzification. In the proposed model the most frequent method is used which is called the Mamdani method. The structure of the purposed FIS model illustrated in Figure 3.1. The fuzzy inference in our model is used three parameters for the input and one parameter for the output.

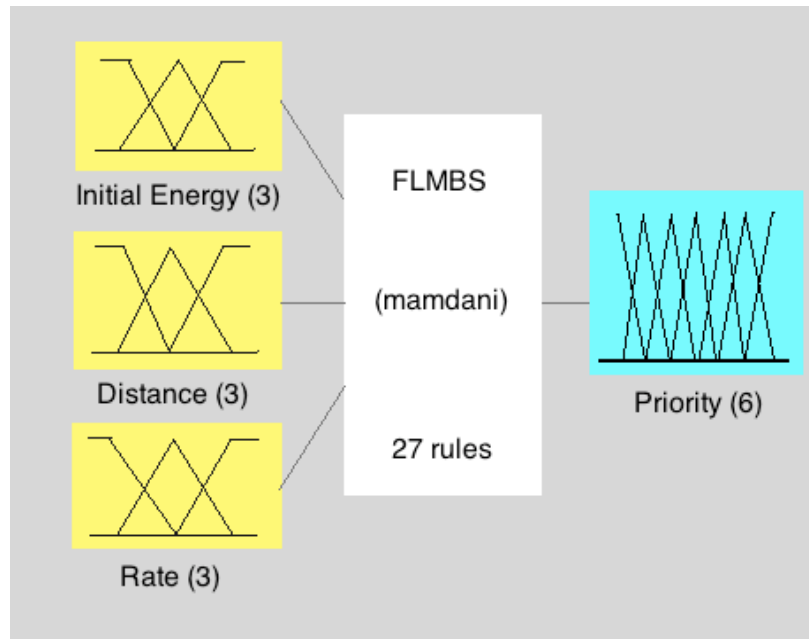


Figure 3.1: Fuzzy Inference System for FLMBs

3.3 FIS Parameters and Rules

In the proposed model, FLMBs, we considered three attributes of each CH (residual energy of

the CH ($e(0)$), distance between the CH and BS (d), and the data transmission rate (r) as an input into to our FIS. These attributes are chosen as input parameters because they are the factors directly influencing energy dissipation and network lifetime. The output of FIS is defined as priority, which determines how important each CH is to prolonging network lifetime.

We divide each of the three input linguistic variables used to represent these parameters into three levels: *low*, *adequate*, and *high* for energy level of the CH; and *close*, *medium*, and *far* for the distance to the BS; and *slow*, *average*, and *fast* for data transmission rate. As we mentioned earlier, many types of membership functions are available in the MATLAB Fuzzy Logic toolbox [42] including Triangle, Trapezoidal, Sigmoidal, Guassian, S-shape, and Z-shape. However, the triangle and trapezoidal membership functions are more useful than the other types because their degrees are more easily determined so, that is why we decided to use these two membership functions only. Formulas for triangle and trapezoidal membership functions are shown in appendix A. The three input membership functions are shown in Figures 3.2, 3.3, and 3.4. The linguistic values for the middle level of the three inputs (*adequate*, *medium*, *average*) are represented by triangle membership functions, while the other linguistic values are represented by trapezoidal membership functions. For the output linguistic variable, priority, we use seven linguistic values: *very low*, *low*, *med low*, *med*, *med high*, *high*, *very high*. All seven membership functions of output linguistic values are represented by triangular function as shown in Figure 3.5.

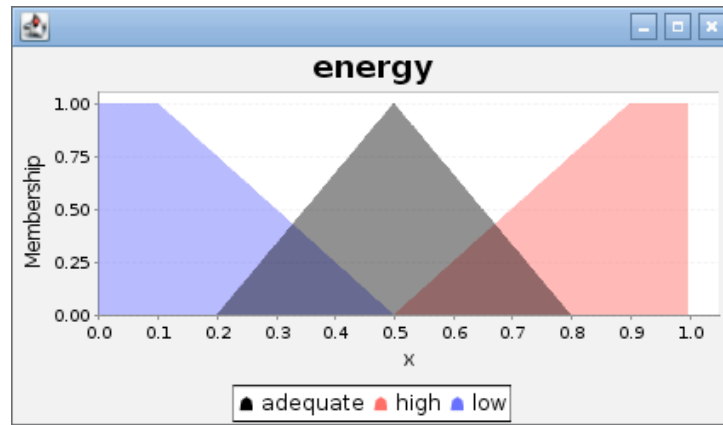


Figure 3.2: Membership Function of Initial Energy

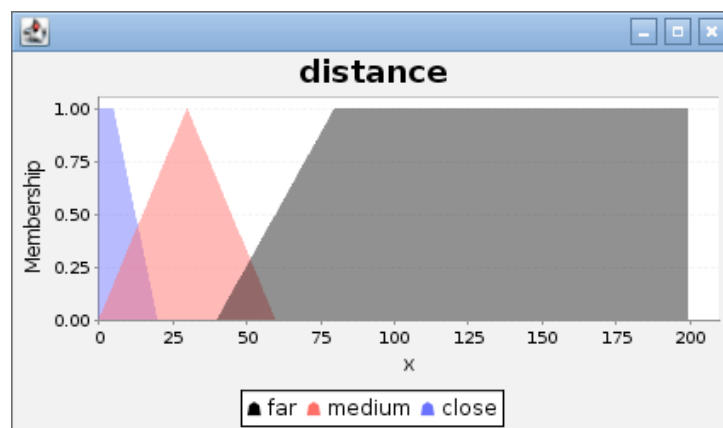


Figure 3.3: Membership Function of Distance to the BS

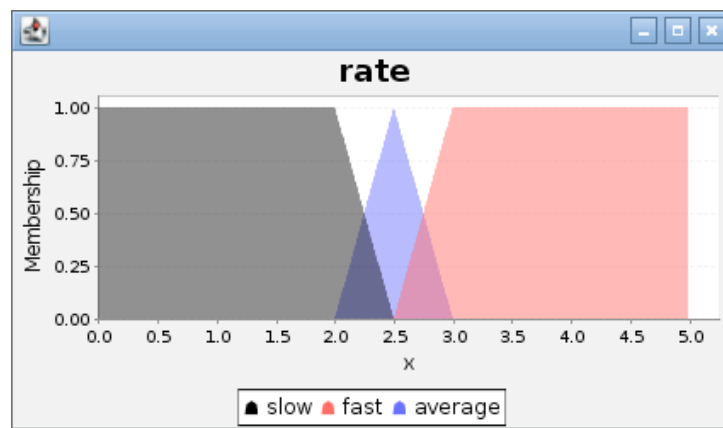


Figure 3.4: Membership Function to Data Transmission Rate

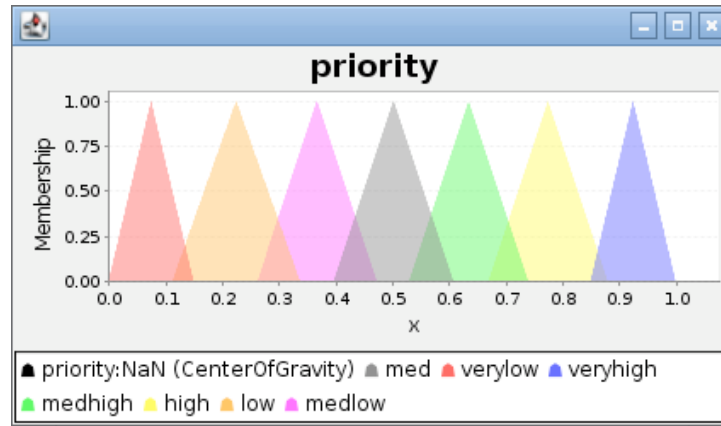


Figure 3.5: Membership Function of Priority

Since we have three parameters, each divided into three levels, we have $3^3=27$ possible priority values, each corresponding to an IF-THEN rule (Table 3.1). These rules fall between two extreme cases as shown next:

Case (1): if $e(0)$ is low and d is far and r is fast

Then priority is veryhigh

Case (2): if $e(0)$ is high and d is close and r is slow

Then priority is verylow

Table 3.1: Fuzzy Inference System IF-THEN rules

e(0)	d	r	Priority
Low	Far	Fast	VeryHigh
Low	Far	Average	VeryHigh
Low	Far	Slow	High
Low	Medium	Fast	MedHigh
Low	Medium	Average	MedHigh
Low	Medium	Slow	MedHigh
Low	Close	Fast	Med
Low	Close	Average	MedLow
Low	Close	Slow	MedLow
Adequate	Far	Fast	VeryHigh
Adequate	Far	Average	High
Adequate	Far	Slow	MedHigh
Adequate	Medium	Fast	MedHigh
Adequate	Medium	Average	Med
Adequate	Medium	Slow	MedLow
Adequate	Close	Fast	MedLow
Adequate	Close	Average	Low
Adequate	Close	Slow	VeryLow
High	Far	Fast	MedHigh
High	Far	Average	MedHigh
High	Far	Slow	Med
High	Medium	Fast	MedLow
High	Medium	Average	MedLow
High	Medium	Slow	MedLow
High	Close	Fast	Low
High	Close	Average	VeryLow
High	Close	Slow	VeryLow

Note that initially we intuitively determined the membership function values for each parameter. We then varied the values somewhat and tested the resulting different FIS's in our

experiments. We found that the FIS with the above membership function values gave the most consistent results.

3.4 Determination of CH priority value

This section describes how to use a popular Mamdani fuzzy inference technique that was described and explained in chapter 2. To show and clarify how to use FIS to determine CH priority value by the BS, a simple example is considered as below.

Assume the following scenario, a CH with initial energy level ($=0.25J$) and it is located at a distance (45m) from the BS. Also the data transmission rate to send the data is (3.2kbps). The following four steps provide details for the calculation of priority value in the FIS.

3.4.1 Step 1: Input of Crisp Value and Fuzzification

First we will forward our crisp value inputs, initial energy of the CH ($=0.25J$), distance to the BS ($=45m$), and CH data transmission rate ($=3.2kbps$) to our FIS.

The value of membership functions are derived from these three crisp inputs. This is done by finding the intersection point of the value of our parameters with the degree of the membership function. These degrees of membership functions are calculated as follows and they are illustrated in Figures 3.6, 3.7 and 3.8.

1. The membership function for energy is calculated as follows:

$$\mu_{low}(x) = \begin{cases} 1 & \text{if } x < 0.1 \\ (\frac{0.5-x}{0.4}) & \text{if } 0.1 \leq x \leq 0.5 \\ 0 & \text{if } x > 0.5 \end{cases}$$

$$\mu_{adequate}(x) = \begin{cases} 0 & \text{if } x < 0.2 \\ \frac{x-0.2}{0.3} & \text{if } 0.2 \leq x \leq 0.5 \\ (\frac{0.8-x}{0.3}) & \text{if } 0.5 \leq x \leq 0.8 \\ 0 & \text{if } x > 0.8 \end{cases}$$

$$\mu_{high}(x) = \begin{cases} 0 & \text{if } x < 0.5 \\ (\frac{x-0.5}{0.4}) & \text{if } 0.5 \leq x \leq 0.9 \\ 1 & \text{if } x > 0.9 \end{cases}$$

So if energy is equal to 0.25J then we have:

$$\mu_{low}(0.25) = (\frac{0.5-0.25}{0.4}) = 0.625$$

$$\mu_{adequate}(0.25) = (\frac{0.25-0.2}{0.3}) = 0.166$$

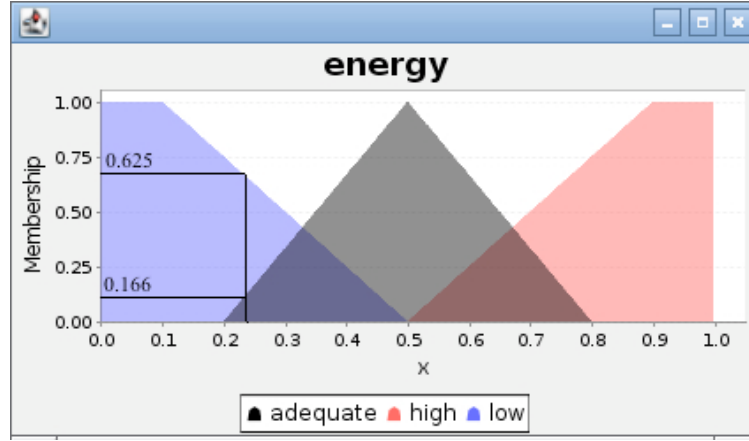


Figure 3.6: Fuzzification of crisp input Initial Energy (0.23J)

2. The membership function for distance is calculated as follows:

$$\mu_{close}(x) = \begin{cases} 1 & \text{if } x < 5 \\ \frac{20-x}{15} & \text{if } 5 \leq x \leq 20 \\ 0 & \text{if } x > 20 \end{cases}$$

$$\mu_{medium}(x) = \begin{cases} 0 & \text{if } x < 0 \\ \frac{x-0}{30} & \text{if } 0 \leq x \leq 30 \\ \frac{60-x}{30} & \text{if } 30 \leq x \leq 60 \\ 0 & \text{if } x > 60 \end{cases}$$

$$\mu_{far}(x) = \begin{cases} 0 & \text{if } x < 40 \\ \frac{x-40}{40} & \text{if } 40 \leq x \leq 80 \\ 1 & \text{if } x > 80 \end{cases}$$

So if distance is equal to 45m then we have:

$$\mu_{medium}(45) = \left(\frac{60-45}{30}\right) = 0.5$$

$$\mu_{far}(45) = \left(\frac{45-40}{40}\right) = 0.125$$

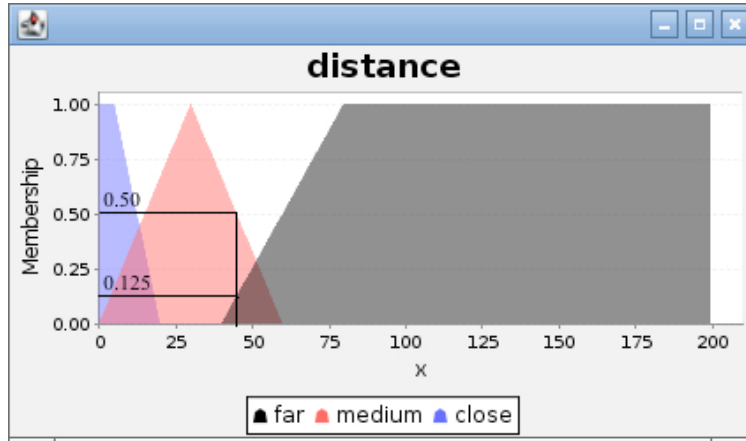


Figure 3.7: Fuzzification of crisp input distance (45m)

3. The membership function for rate is calculated as follows:

$$\mu_{slow}(x) = \begin{cases} 1 & \text{if } x < 2.0 \\ \left(\frac{2.5-x}{0.5}\right) & \text{if } 2.0 \leq x \leq 2.5 \\ 0 & \text{if } x > 2.5 \end{cases}$$

$$\mu_{average}(x) = \begin{cases} 0 & \text{if } x < 2.0 \\ \frac{x-2.0}{0.5} & \text{if } 2.0 \leq x \leq 2.5 \\ \left(\frac{3.0-x}{0.5}\right) & \text{if } 2.5 \leq x \leq 3.0 \\ 0 & \text{if } x > 3.0 \end{cases}$$

$$\mu_{fast}(x) = \begin{cases} 0 & \text{if } x < 2.5 \\ \left(\frac{x-2.5}{0.5}\right) & \text{if } 2.5 \leq x \leq 3 \\ 1 & \text{if } x > 3.0 \end{cases}$$

So if rate is equal to 3.2 then we have:

$$\mu_{fast}(3.2) = 1$$

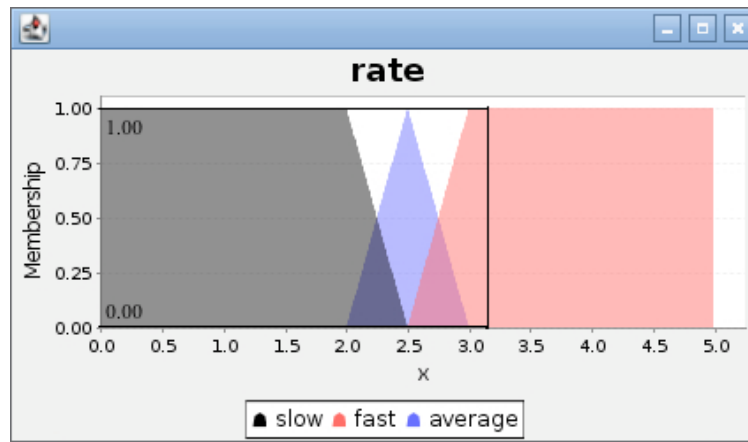


Figure 3.8: Fuzzification of the crisp input Rate (3.2kbps)

3.4.2 Step 2: Rule Evaluation

After the fuzzification step has been completed and the membership values have been obtained, we supply/feed these values to our IF-THEN rules to determine our new fuzzy output sets. We use the fuzzy operator (AND) to evaluate the rules. Basically, for each rule the minimum of the three membership values is used to obtain the priority membership value. See Table 3.2 below.

Table 3.2: Fuzzy Inference System IF-THEN rules (Example for $e(0)=2.5J$, $d=45m$, $r=3.2kbps$)

$e(0)$	d	r	Priority
Low(0.625)	Far(0.125)	Fast(1)	VeryHigh(0.125)
Low(0.625)	Far(0.125)	Average(0)	VeryHigh(0)
Low(0.625)	Far(0.125)	Slow(0)	High(0)
Low(0.625)	Medium(0.50)	Fast(1)	MedHigh(0.50)
Low(0.625)	Medium(0.50)	Average(0)	MedHigh(0)
Low(0.625)	Medium(0.50)	Slow(0)	MedHigh(0)
Low(0.625)	Close(0)	Fast(1)	Med(0)
Low(0.625)	Close(0)	Average(0)	MedLow(0)
Low(0.625)	Close(0)	Slow(0)	MedLow(0)
Adequate(0.166)	Far(0.125)	Fast(1)	VeryHigh(0.125)
Adequate(0.166)	Far(0.125)	Average(0)	High(0)
Adequate(0.166)	Far(0.125)	Slow(0)	MedHigh(0)
Adequate(0.166)	Medium(0.50)	Fast(1)	MedHigh(0.125)
Adequate(0.166)	Medium(0.50)	Average(0)	Med(0)
Adequate(0.166)	Medium(0.50)	Slow(0)	MedLow(0)
Adequate(0.166)	Close(0)	Fast(1)	MedLow(0)
Adequate(0.166)	Close(0)	Average(0)	Low(0)
Adequate(0.166)	Close(0)	Slow(0)	VeryLow(0)
High(0)	Far(0.125)	Fast(1)	MedHigh(0)
High(0)	Far(0.125)	Average(0)	MedHigh(0)
High(0)	Far(0.125)	Slow(0)	Med(0)
High(0)	Medium(0.50)	Fast(1)	MedLow(0)
High(0)	Medium(0.50)	Average(0)	MedLow(0)
High(0)	Medium(0.50)	Slow(0)	MedLow(0)
High(0)	Close(0)	Fast(1)	Low(0)
High(0)	Close(0)	Average(0)	VeryLow(0)
High(0)	Close(0)	Slow(0)	Verylow(0)

3.4.3 Step 3: Aggregation of the Rule Outputs

After completing fuzzification and rule evaluation steps, the aggregation step will start. The aggregation is a process of the union of all the outputs obtained from applying all rules (27 rules in our FIS model). Since we are looking at aggregating all our rules we have used an (OR) fuzzy logic operator. The OR operator simply selects the maximum of our rule evaluation values, to generate the new aggregate fuzzy set that we will use in next step. The Figure 3.9 below illustrates the aggregation output of the rules.

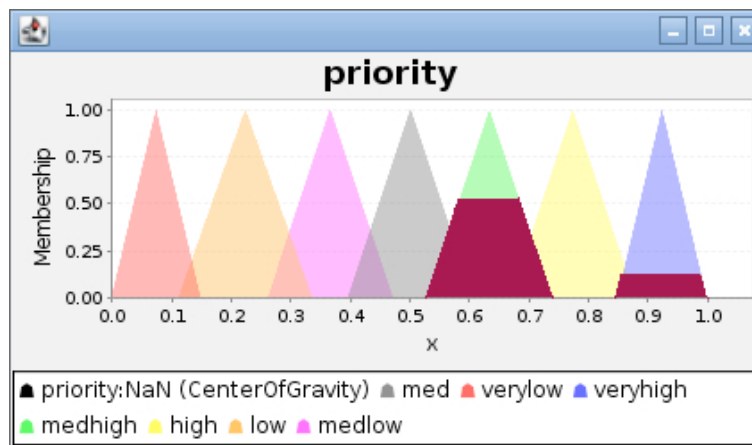


Figure 3.9: Output evaluation of fuzzy IF-THEN rules

3.4.4 Step 4: Defuzzification

Defuzzification is the last step, in fuzzy Inference system, where the priority for this CH is obtained. Since centroid is considered as one of the most popular defuzzification method and it

has been successfully applied in different applications [48, 49, 50] we decide to use it in this work. Centroid defuzzification method is used to find the final crisp number, which represents the CH priority value. By applying the values we got previously from step 3 in the equation (2.9) and calculating the algebraic integration, we determine the priority value for CH is approximately equal to (= 0.69). In here we show the calculation for COG.

$$\begin{aligned}
 \text{COG} &= \\
 &\left[\int_{0.53}^{0.58} \left(\frac{x-0.53}{2} \right) dx + \int_{0.58}^{0.69} (0.5) dx + \int_{0.69}^{0.74} \left(\frac{7.4-x}{2} \right) dx + 0.125 \int_{0.85}^{0.86} (x-0.85) dx + \int_{0.86}^{0.99} 0.125 x dx + 0.125 \int_{0.99}^{1.00} (1.0-x) dx \right] \\
 &\div \\
 &\left[\int_{0.53}^{0.58} \left(\frac{x-0.53}{2} \right) dx + \int_{0.58}^{0.69} (0.5) dx + \int_{0.69}^{0.74} \left(\frac{7.4-x}{2} \right) dx + 0.125 \int_{0.85}^{0.86} (x-0.85) dx + \int_{0.86}^{0.99} 0.125 dx + 0.125 \int_{0.99}^{1.00} (1.0-x) dx \right] \\
 &= 0.69
 \end{aligned}$$

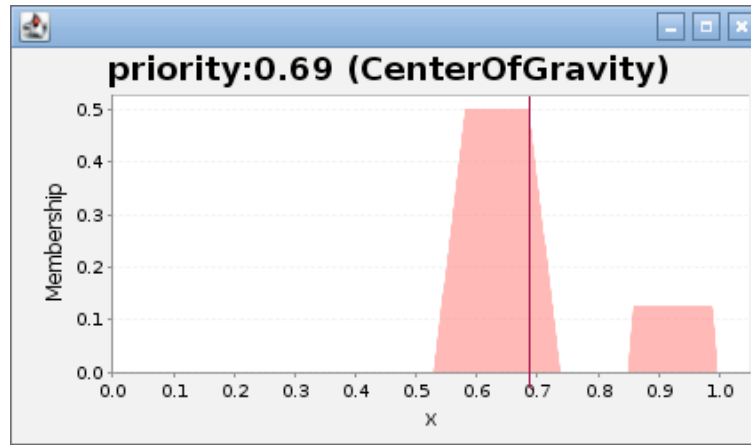


Figure 3.10: The Centroid point

COG returns the Centrer Of Area under the curve. The COG is the point along the x-axis about which this shape would balance [51]. We also provide some examples of priorities calculated for CH's with different conditions, using mamdani FIS and centroid defuzzification in Table 3.3.

Table 3.3: Priority value for different cluster-heads

NO.	Initial Energy	Distance to BS	Data Transfer Rate	Priority using FIS
1	0.2	45	4	0.6878
2	0.2	15	4	0.5796
3	0.2	128	3	0.9250
4	0.5	10	2	0.2115
5	0.8	80	4	0.6350
6	0.8	10	1	0.1324

However, we can use methods other than the centroid method to calculate the priority of the CHs. But, centroid is more useful and easier than other methods, as we mentioned previously.

3.5 BS Movement Example

A simple example in a two-dimensional space is demonstrated in here in order to show the BS movement strategy in our approach. In the following example, we consider four CHs and show their initial parameters in Table 3.4. The two-dimensional coordinates of CHs are represented

by the “Location” parameters, CHs data transmission rates are represented by the “Rate” parameters and the current energy for all the CHs are represented by the “Energy” parameters. Finally the “Priority” parameters represent the priority of all CHs calculated by using our fuzzy logic inference system. For simplicity we make the rate and energy of the CHs uniform, in other words the initial rate and energy parameters of all four CHs are the same. We also choose the step size to be equal to 2 and initial BS location to be at (5,5), thus giving a similar setup as what was in the experiments.

Table 3.4: Example parameters

Number	Location	Rate	Energy	Priority
N1	(2,1)	3	1	0.25
N2	(3,5)	3	1	0.24
N3	(5,8)	3	1	0.24
N4	(9,9)	3	1	0.26

In the following figure the positions of the given CHs are shown with the blue dots and the initial position of a BS is shown with the red square.

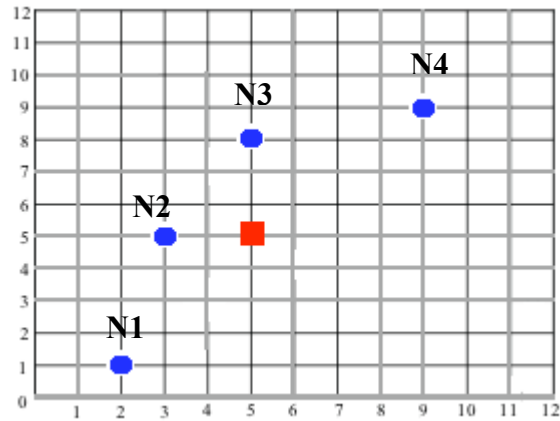


Figure 3.11: the position of the given CHs and the initial BS position

First the vectors between a BS and all the CHs must be calculated as shown in Figure 3.12, the calculation is done by subtracting the beginning point from the end point:

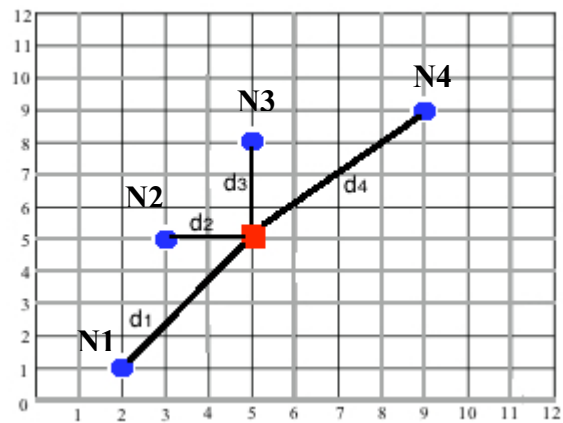


Figure 3.12: vectors between CHs and BS

$$V_1 = \langle (2-5), (1-5) \rangle = \langle -3, -4 \rangle$$

$$V_2 = \langle (3-5), (5-5) \rangle = \langle -2, 0 \rangle$$

$$V_3 = \langle (5-5), (8-5) \rangle = \langle 0, 3 \rangle$$

$$V_4 = \langle (9-5), (9-5) \rangle = \langle 4, 4 \rangle$$

Next step is to calculate the magnitude or the length of these vectors:

$$|d_1| = \sqrt{(2-5)^2 + (1-5)^2} = 5.00$$

$$|d_2| = \sqrt{(3-5)^2 + (5-5)^2} = 2.00$$

$$|d_3| = \sqrt{(5-5)^2 + (8-5)^2} = 3.00$$

$$|d_4| = \sqrt{(9-5)^2 + (9-5)^2} = 5.65$$

Now we can compute the unit vector by dividing each vector by its magnitude.

$$U_{B1} = \frac{\langle -3, -4 \rangle}{5} = \langle \frac{-3}{5}, \frac{-4}{5} \rangle = \langle -0.60, -0.80 \rangle$$

$$U_{B2} = \frac{\langle -2, 0 \rangle}{2} = \langle -1, 0 \rangle$$

$$U_{B3} = \frac{\langle 0, 3 \rangle}{3} = \langle 0, 1 \rangle$$

$$U_{B4} = \frac{\langle 4, 4 \rangle}{5.65} = \langle \frac{4}{5.65}, \frac{4}{5.65} \rangle = \langle 0.70, 0.70 \rangle$$

Next step is to multiply each unit vector by the corresponding CH's priority.

$$\langle -0.60, -0.80 \rangle \times 0.25 = \langle -0.15, -0.20 \rangle$$

$$\langle -1, 0 \rangle \times 0.24 = \langle -0.24, 0 \rangle$$

$$\langle 0, 1 \rangle \times 0.24 = \langle 0, 0.24 \rangle$$

$$\langle \frac{4}{5.65}, \frac{4}{5.65} \rangle \times 0.26 = \langle 0.18, 0.18 \rangle$$

And then multiply the result by the step size:

$$\langle -0.15, -0.20 \rangle \times 2 = \langle -0.3, -0.4 \rangle$$

$$\langle -0.24, 0 \rangle \times 2 = \langle -0.48, 0 \rangle$$

$$\langle 0, 0.24 \rangle \times 2 = \langle 0, 0.48 \rangle$$

$$\langle 0.18, 0.18 \rangle \times 2 = \langle 0.36, 0.36 \rangle$$

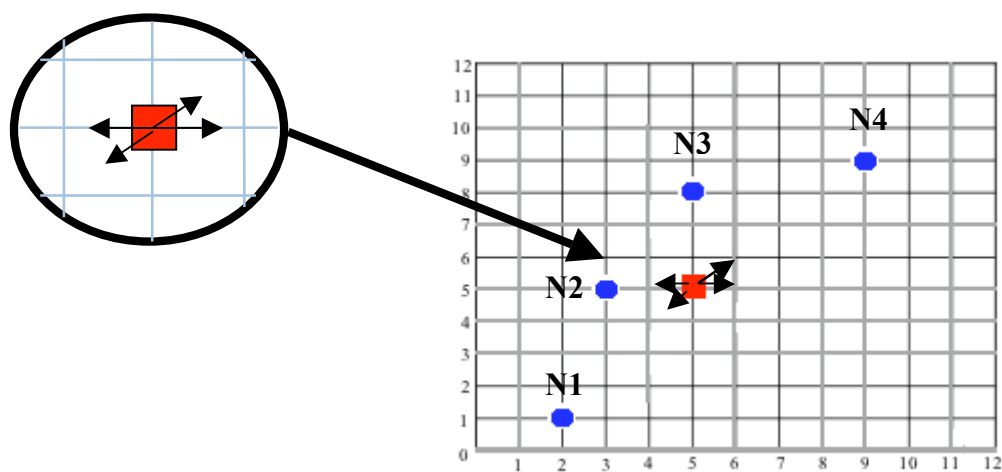


Figure 3.13: resulting vectors

Next we add the resulting vectors to produce a *net* vector as shown in Figure 3.14.

$$\langle -0.30, -0.40 \rangle + \langle -0.48, 0 \rangle + \langle 0, 0.48 \rangle + \langle 0.36, 0.36 \rangle = \langle -0.42, 0.44 \rangle$$

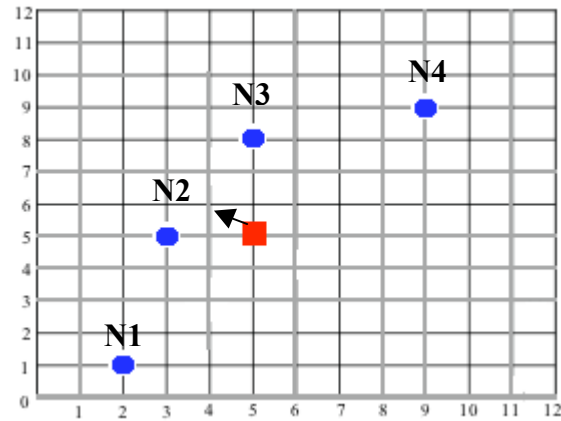


Figure 3.14: Net movement vector

Finally we can add the net movement vector to the current BS location to obtain the final BS location.

$$\langle 5, 5 \rangle + \langle -0.42, 0.44 \rangle = \langle 4.58, 5.44 \rangle$$

All these vector calculations are derived from the book called “Linear Algebra: Ideas and Applications“ [52].

CHAPTER IV

Simulation and Results

In this chapter, we performed two groups of experiments: in the first group of experiments we compared our proposed FLMBBS with the original LEACH protocol. Each experiment ran with a different scenario to show how the behavior of FLMBBS compared with that of LEACH, paying particular attention to network lifetime. Since the BS is static in the original LEACH protocol while it is mobile in ours, we decided to include another algorithm [31], PSO (Particle Swarm Optimization), which considers mobile BS in a cluster-based architecture, in order to evaluate the performance of our purposed FLMBBS against the performance of another mobile-BS algorithm. This occurred in the second group of experiments.

4.1 Network Model and Assumptions

We made some assumptions about the SNs and the underlying network model:

- There was a BS, which was mobile for FLMBBS and PSO, and static for LEACH.
- All nodes remained stationary after deployment.
- All nodes had the same initial energy.
- The nodes had different data transmission rates.
- Each node sensed the environment at a fixed rate and always had data to send.

- The BS was occupied with a GPS system to allow distances calculation [53].
- The energy expenditure model of the sensor nodes were based on the first order radio model, which we described in subsection 2.1.3.1.
- The nodes were deployed randomly in a 2-D plane with uniform random distribution.
- For simplicity, the time and energy taken for BS movement was negligible.

4.2 First Group of Experiments (FLMBS vs LEACH)

Here, we compared two cluster-based or hierarchical routing protocols in terms of network lifetime. The first was the well-known LEACH protocol, which was described by Heinzelman et al. [7]. The second was the FLMBS protocol established in this work. All experimental simulations were carried out in MATLAB [54]. MATLAB is a high performance language for technical computing. It integrates programming, computation and visualization in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notations. The name MATLAB stands for matrix laboratory. We had chosen MATLAB because it is powerful and enjoys widespread usage within this field. Four experiments with different parameters were performed in this group. The simulation parameters for each experiment are provided in Table 4.1 shown below.

In each experiment, FLMBS was compared to LEACH. The key differences between the experiments were the initial BS location (note that BS stays there for LEACH but moves during FLMBS), and the size of the network (number of nodes and size of area).

Between experiments 1 and 2, we got to see the effects of the initial BS location (outside the

network area vs. inside the network area). Experiments 2, 3 and 4 showed the scalability of FLMBs vs. LEACH. Since the locations of the SN's were generated randomly and the CH's within each round of the algorithms were selected randomly as well, we performed each experiment 10 times in order to account for the influences by the random factors

Table 4.1: Input Parameters (First group of Experiments)

Parameters	Experiment 1	Experiment 2	Experiment 3	Experiment 4
Size of the area	100m*100m	100m*100m	1000m*1000m	1000m*1000m
Init. BS location	(50,150)	(50,50)	(500,500)	(500,500)
Number of nodes	100	100	100	200
Initial energy	1J	1J	500J	500J
E_{elec}	50 nJ/bit/m ²	50 nJ/bit/m ²	50 nJ/bit/m ²	50 nJ/bit/m ²
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	5 nJ/bit/signal	5 nJ/bit/signal
Data packet size	2000 bits	2000 bits	2000 bits	2000 bits
Step size	20m	20m	200m	200m
Data rate	1-5	1-5	1-5	1-5

Note that we used differently scaled membership functions in our FIS and different step sizes for the different experiments. For instance, when we increased grid size from 100x100 to 1000x1000, the distance and energy membership functions were scaled proportionally and the step size scaled from 20 to 200. We chose step size to be 20% of the grid length after some trials with different step sizes (data not shown). We found that smaller step sizes gave poorer results but increasing the step size further did not result in much, if at all, improvement.

We used in FLMBBS the same basic network operation model as the LEACH protocol. However, our algorithm involved moving the BS each round after the CHs had been chosen, in order to lessen the energy dissipation of the cluster heads. Our initial strategy was to move the BS directly towards the CH with the highest priority (as calculated by the FIS). This strategy proved to be too simple and ineffective, which was understandable because it only took into account the needs of one cluster head. It was all too possible that this cluster head's energy dissipation had been improved but to the detriment of the other CH's, resulting in a worsening of the overall CH energy dissipation. Therefore, a new movement strategy was proposed and carried out. Here, instead of focusing on just one CH, the BS moved towards all CH's according to their priorities. The amount of distance BS moved towards any CH was equal to the priority of that CH multiplied by a constant step size. Thus for each CH, there was a unique BS-movement vector towards it. The final BS location in this round was calculated by adding all these movement vectors together. In the case when one or more CH's distance from BS was less than the step size, we took the minimum BS-to-CH-distance and used that as the step size. Note that at the beginning of each round, the BS location was reset to the default location at the center of the grid (this was also the static BS location in LEACH). This was done since different CH's were chosen each round.

The new strategy produced solid results. Improvement over static-BS algorithm (LEACH) could be seen by examining CH energy dissipation in one round or by looking at the network lifetime. The improvement became more significant in a bigger grid, i.e. distances were larger and played a much bigger role in the calculation of energy dissipation. This made sense because our strategy aimed to minimize energy dissipation by minimizing distance.

And finally in the steady state phase all the SNs send their fixed amount of data to their own CHs. Later the CHs aggregate these data and send the compressed data to the BS. Using this network operation mode allows the network lifetime metric to be measured in data collection rounds till the very first node runs out of its energy. Different authors define the network lifetime in different ways: some of them consider the network lifetime as number of rounds until the death of the first node (FND) and others consider the death of the last node (LND) to signify end of lifetime. For the work presented in this thesis, we will focus on the FND metric to test and analyze our algorithm.

4.2.1 Experiment 1

In the first experiment our field dimension is $100 \times 100 \text{ m}^2$ and we randomly distributed 100 sensor nodes in the search space. We chose 5% of the nodes as CH and located the BS outside the field at (50,150). For this experiment we allocated each SN with an initial energy of 1J and a random number between 1 and 5 for the data transmission rate. We used the same clustering strategy as LEACH protocol, so each SN selected a number between 0 to 1 and if the number was less than some predefined threshold, which we explained earlier in subsection 2.1.3.1 the node became a CH. The configuration parameters for this experiment are shown in Table 4.1. We implemented our algorithm based on the above network features and parameters using MATLAB.

Figure 4.1 shows the number of nodes alive for both algorithm and it is clear that our FLMBS outperformed the LEACH algorithm. In a typical run, the number of rounds before the first node died in the LEACH protocol and the FLMBS were 2566 and 2697 respectively. Nodes

began to die faster in the LEACH protocol than in FLMBs. Our algorithm also showed improvement in terms of the LND. Figure 4.1 shows that the number of rounds before the last node died in the network using the LEACH protocol is lower than the number of rounds before the last node died in the network using the FLMBs protocol (3476 and 3704 respectively).

As we explained earlier, we considered the network lifetime as the time elapsed until the first node died. Our proposed scheme FLMBs improved the network lifetime by 5.4% compare to the LEACH protocol. The percentage difference in the network lifetime was calculated using equation (4.1).

$$\text{Percentage Difference} = \left| \frac{r_{FLMBs} - r_{LEACH}}{r_{LEACH}} \right| \times 100 \quad (4.1)$$

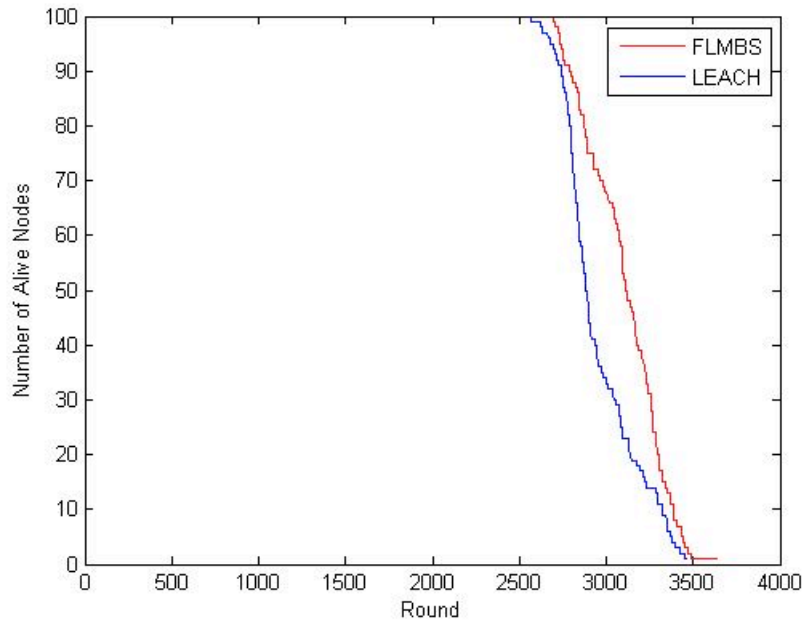


Figure 4.1: Experiment 1 – The number of alive nodes for an area 100m x100m with 100SNs and BS at (50,150)

Table 4.2: Experiment 1 - FND metrics for FLMBS and LEACH

Run	FLMBS FND	LEACH FND	Difference
1	2706	2564	142
2	2670	2626	44
3	2643	2493	150
4	2697	2566	131
5	2711	2558	145
6	2679	2498	181
7	2699	2634	65
8	2636	2487	149
9	2655	2483	172
10	2710	2532	178
Average	2680.60	2544.10	136.5

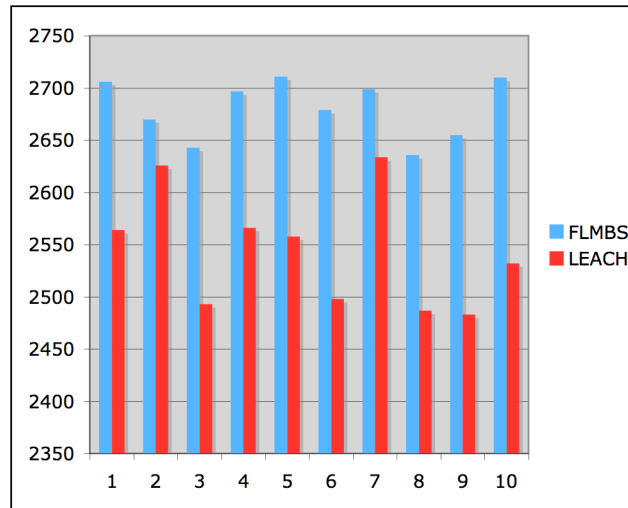


Figure 4.2: Experiment 1 - FND value for FLMBS and LEACH

Table 4.2 shows the FND result for both protocols for the ten runs. We observe that the

average value for FND in LEACH is 2544.10, whereas, in the FLMBBS, the average is 2680.60. The same data is plotted in Figure 4.2 for clarity.

T-test:

The t-test is often used to calculate the significance of observed differences between the means of two samples [55]. The t-test is generally used with scalar variables, such as length and width, and so on. The null hypothesis is that there are no significant differences between the means. For this experiment the two-sample t-test ($p=0.05$) was performed on the two sets of data and the null hypothesis was rejected. The calculated p-value was 6.4×10^{-6} . Hence there was a significant improvement from the proposed algorithm over LEACH on the 100x100 grid and with initial BS location at (50,150). All the t-tests were performed in MATLAB.

This experiment was not truly meaningful as it wasn't very fair. The BS remains far outside of the network field in LEACH but it moves into the field in FLMBBS. Even our initial strategy, where we simply moved the BS towards the CH with the highest priority, produced much improvement over LEACH. Essentially, as long as the CH gets moved to be among the SNs there shall be great improvement. This is an unrealistic scenario for comparison because surely in a real world situation if the BS can be moved inside of the network field it can also be set up there in the beginning for LEACH. This is why we choose the initial BS location (i.e. also the static BS location for LEACH) to be in the center of the network field for the subsequent experiments. We only performed experiment 1 for completeness, as we found papers [49,50] where authors claimed much improvement over LEACH using a mobile-BS protocol but in

fact had the static BS location for LEACH outside of the network field.

4.2.2 Experiment 2

All settings were exactly the same as in the previous experiment except this time we placed the BS initially inside the field at (50, 50). In this experiment we chose the initial energy of nodes to be 1J again and grid is $100 \times 100 \text{ m}^2$. Figure 4.3 shows the network lifetime for the LEACH and FLMBBS algorithms by applying the parameters shown in Table 4.1 in a typical run of the experiment.

Figure 4.3 shows the number of nodes alive for both algorithms. Since we consider $100 \times 100 \text{ m}^2$ search space the distance between the nodes and BS is low, so it is not very clear that our FLMBBS improves the number of round before the first node dies compare to LEACH algorithm.

Distance is the important descriptor in our FIS and plays a major role in our simulations. In the next section we will show how our method improves by increasing the size of the network area.

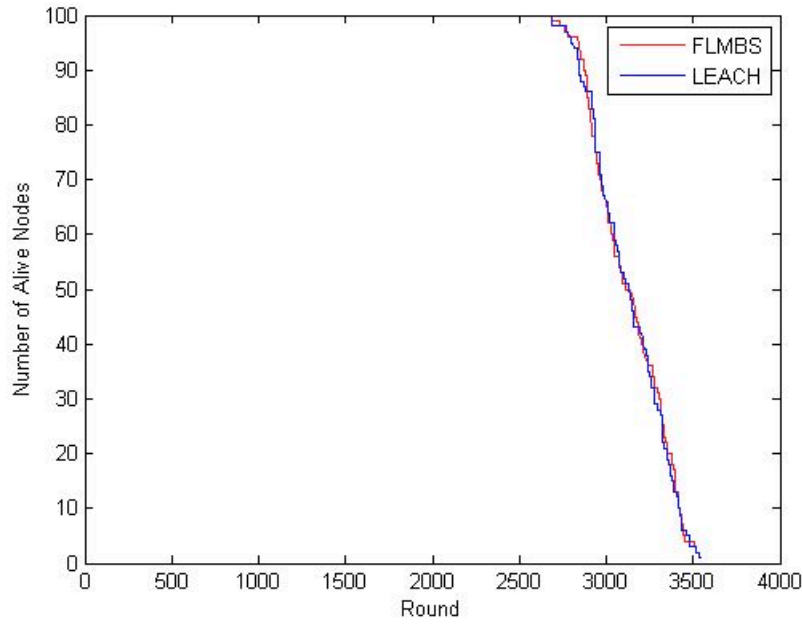


Figure 4.3: Experiment 2 - The number of alive nodes for an area 100m x100m with 100SNs and BS at (50,50)

Table 4.3: Experiment 2 - FND metrics for FLMBS and LEACH

Run	FLMBS FND	LEACH FND	Difference
1	2684	2712	-28
2	2695	2691	4
3	2655	2641	14
4	2677	2695	-18
5	2623	2644	-21
6	2658	2646	12
7	2652	2604	48
8	2589	2594	-5
9	2599	2638	-39
10	2712	2701	11
Average	2663.4	2656.6	6.8

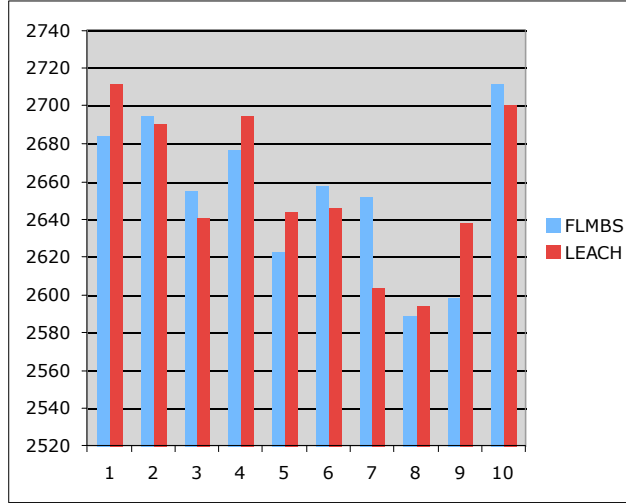


Figure 4.4: Experiment 2 - FND value for FLMBS and LEACH

Table 4.3 shows the FND result for both protocols for the ten runs. The average value for FND in LEACH is 2656.60, whereas, in the FLMBS, the average is 2663.40. There seems to be a tiny improvement for FLMBS over LEACH but the difference is not significant. The same data is plotted in Figure 4.4 for clarity.

We use equation 4.1 in order to calculate the percentage of improvement until the first node died. Our proposed scheme FLMBS improved the network lifetime by 0.3% compare to the LEACH protocol.

T-test:

For this experiment the two-sample t-test ($p=0.05$) was performed on the two sets of data and the null hypothesis was *not* rejected. Hence there was no significant improvement from the

proposed algorithm over LEACH on the 100x100 m² grid with BS at (50,50).

We could see the reason for this readily when we looked at the energy dissipation of CH's in one round. Here, in one round where all nodes were alive the same CH's were selected for both LEACH and FLMBBS (same network) and the energy dissipation was calculated for every CH using both protocols. The difference in energy dissipation as found using the two protocols was then calculated for each CH and an average difference was found (positive value meant FLMBBS had less energy dissipation). This difference was compared to the LEACH energy dissipation per CH to give a percentage difference. Finally, the average BS-to-CH distance was found for each protocol and the difference between the two averages was calculated. The results are tabulated below in Table 4.4.

As can be seen from the table below, in the 100x100 m² network area, there was minimal improvement in terms of energy dissipation of CH per round. Hence we did not see significant improvement in terms of network lifetime.

This minimal difference was due to the relative weight given to the BS-to-CH distance parameter in the calculation of CH energy dissipation. For the 100x100 m² grid, the broadcast distance (diagonal distance of grid) was greater than $d_0 = 87.7058\text{m}$ and the typical BS-to-CH distance was smaller than d_0 , and hence the following equation was used:

$$\begin{aligned}
 E1 = E0 - & \\
 & (ETX * ctrPacketLength + Emp * ctrPacketLength * (distanceBroad^4)) \\
 - & ((ETX + EDA) * packetLength + Efs * packetLength * (distance^2))
 \end{aligned} \tag{4.2}$$

Where E0 and E1 were the initial and final energies in this round respectively. DistanceBroad was the diagonal length of the grid and distance was BS-to-CH distance. In the 100x100 m² grid, where the BS default location was at (50,50), in a round where all nodes were alive, the BS-to-CH distance reduction by FLMBS (compared to LEACH) was typically less than 20m (data not shown), and the average distance reduction was less than 8m. Such a difference in distance would not produce much difference in energy calculation based on the above equation (4.2). For example, if we take a typical BS-to-CH distance of 40, and a typical distance reduction by FLMBS of 6, then the percentage difference in energy dissipation is roughly 0.6% (calculated using the equation above; percentage difference was higher in the table because the distance reduction given in table were averages).

Table 4.4: Experiment 2 – Energy Dissipation after each round

RUN	Average difference in energy dissipation (J)	LEACH energy dissipation per CH (J)	Percentage difference (%)	Average Distance Reduction (m)
1	2.0840e-004	0.0105	2.0133	12.4310
2	2.4725e-005	0.0043	0.5786	3.2088
3	8.0586e-005	0.0064	1.2542	4.4962
4	2.5860e-006	0.0049	0.0532	1.3015
5	1.3374e-004	0.0095	1.4200	11.3643
6	4.4667e-005	0.0049	0.9089	5.3649
7	1.7078e-004	0.0094	1.8245	10.5656
8	6.9632e-005	0.0187	0.3717	7.3500
9	7.7452e-005	0.0199	0.3901	7.3501
10	7.0926e-005	0.0067	1.0516	5.7895
Avg	8.8349e-005	0.0095	0.9866	7.0222

On the other hand, if we increased the grid size to 1000x1000 m², the broadcast distance (diagonal distance of grid) was greater than do= 87.7058m but now the typical BS-to-CH distance was also greater than do. Therefore a different equation was used (4.3) to calculate CH energy dissipation:

$$E1 = E0 - \left(ETX * ctrPacketLength + Emp * ctrPacketLength * (distanceBroad^4) \right) - \left((ETX + EDA) * packetLength + Emp * packetLength * (distance^4) \right) \quad (4.3)$$

The difference compared to the previous equation occurred in the second part. Efs= 1.0000e-010 had been replaced by Emp= 1.3000e-014 and distance now had power of 4 instead of 2. Even though the constant term was smaller, the change in power for the distance parameter (when distance was measured on the 1000x1000 m² grid) meant that overall changes in distance were now more significant with respect to the calculation of energy dissipation. If we simply scaled the values from the calculation example for the 100x100 m² grid, i.e. BS-to-CH distance became 400 and the distance reduction by FLMBs was 60, then the percentage difference in energy dissipation was now calculated to be about 3%. Although this was still not a huge difference, it was enough to produce a significant improvement in the network lifetime as seen in the next experiment.

Note that increasing the grid size further would not and did not have any additional major impact on the calculations, since the same equations were used as for the 1000x1000 m² grid.

The above gives important insight into why we did not perceive significant improvement over

LEACH on the 100x100 grid, but did perceive it on a larger grid.

4.2.3 Experiment 3

In the third experiment we used 1000 x 1000 m² field in order to evaluate the performance of our protocol on a larger grid. We chose the initial energy of the nodes to be 500J instead of 1J to fit the larger grid. Membership functions for distance and energy had been rescaled, and so was step size. The MFs for distance and energy in 1000 x 1000 m² grid is shown in appendix B. The default BS location was now at (500,500), while the number of nodes remained 100. Once again we ran the experiment for 10 times, and the results from a typical run are shown in the Figure 4.5.

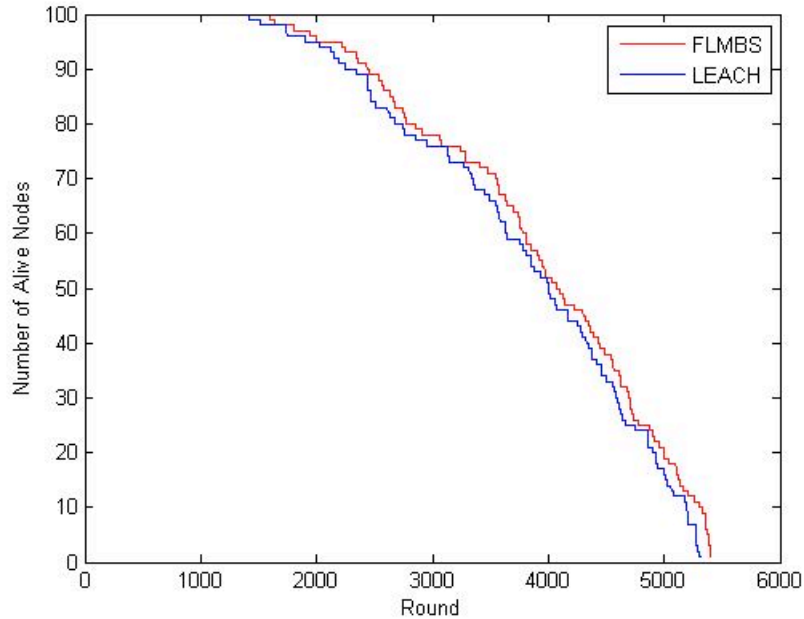


Figure 4.5: Experiment 3 - The number of alive nodes for an area 1000m x1000m with 100SNs and BS at (500,500)

From the figure it is clear that our FLMBS protocol improved the network lifetime significantly. The number of rounds before the first node died in the LEACH protocol vs. that in FLMBS were 1501 and 1554 respectively. The improvement over LEACH was now much more apparent. Our algorithm showed improvement in terms of the LND as well. The number of rounds completed before the last node dies in LEACH and FLMBS were 5310 and 5405 respectively.

Using the FND as the measure of network lifetime, FLMBS in this scenario improved the network lifetime by 5.4% compared to LEACH.

Table 4.5 shows the FND results for 10 runs of the experiment. The same data is plotted in Figure 4.6 for clarity.

Table 4.5: Experiment 3 - FND metrics for FLMBS and LEACH

Run	FLMBS FND	LEACH FND	Difference
1	1531	1537	-6
2	1602	1420	182
3	1492	1489	3
4	1554	1501	53
5	1461	1397	64
6	1509	1530	-21
7	1523	1473	50
8	1481	1465	16
9	1572	1434	138
10	1537	1515	22
average	1526.20	1476.10	50.1

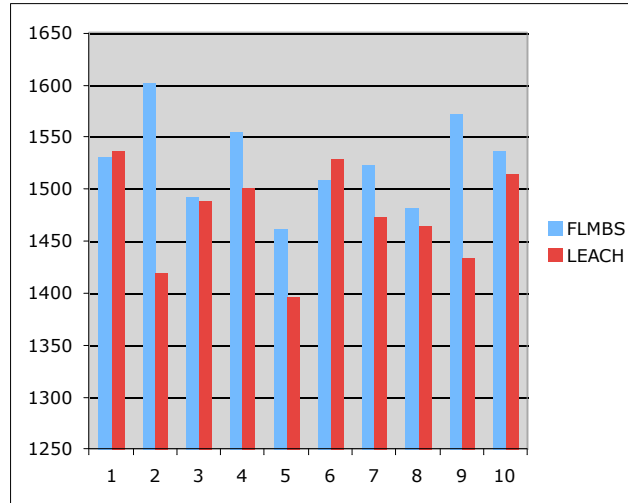


Figure 4.6: Experiment 3 - FND value for FLMBS and LEACH

T-test:

When run on the 1000x1000 grid, FLMBS produced significant improvement over LEACH in terms of round of first-death. The two-sample t-test ($p=0.05$) was performed and the null hypothesis was rejected. The calculated p-value was 0.037. The 5% delay in FND when using FLMBS represented a significant improvement in network lifetime.

Analysis of energy dissipation per round was performed again and as noted previously, in the 1000x1000 m² grid compared to the 100x100 m² grid, the improvement of FLMBS over LEACH was more marked, and this was the basis for the significant improvement in overall network lifetime.

Table 4.6: Experiment 3 – Energy Dissipation after each round

RUN	Average difference in energy dissipation (J)	LEACH energy dissipation per CH (J)	Percentage difference (%)	Average Distance Reduction (m)
1	0.3533	12.4457	2.9213	28.5065
2	1.5064	15.2321	10.9749	176.9541
3	0.1956	12.5047	1.5894	21.6298
4	0.5620	23.5190	2.4480	72.6640
5	0.4515	12.0621	3.8890	90.0188
6	0.3778	12.2496	3.1827	54.8081
7	0.6172	12.4216	5.2287	80.0338
8	0.7188	15.4484	4.8799	228.1422
9	0.3239	13.2179	2.5121	73.5275
10	0.1823	13.0431	1.4174	60.3266
Avg	0.5289	14.2144	3.9043	88.6611

4.2.4. Experiment 4

In the fourth experiment we evaluated the scalability of our model by increasing the number of nodes and enlarging the size of the field. For this experiment we randomly deployed 200 nodes in $1000 \times 1000 \text{ m}^2$. We located the BS in the middle of the field at position (500,500). Increasing the number of nodes in this experiment allowed us to study the effect of node density on FLMBS behavior. In Figure 4.7 we can see that the number of nodes still alive for both algorithms in a typical run of the experiment; our FLMBS performance was better than

the LEACH performance.

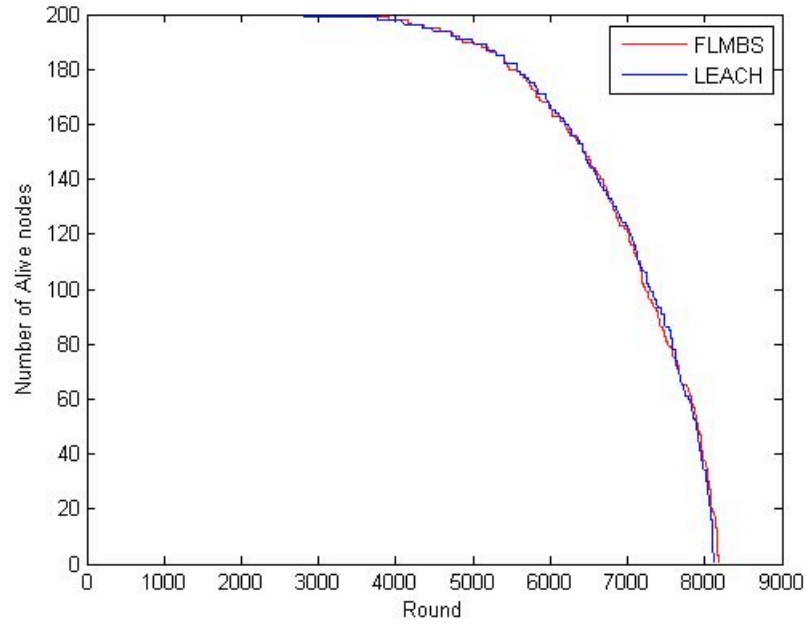


Figure 4.7: Experiment 4 - The number of alive nodes for an area 1000m x1000m with 300SNs and BS at (500,500)

Moreover, to make the results more clear we present the FND metric and plot it for both protocols to show the differences. From the Table 4.7 below we can determine that the average FND for ten independent runs for LEACH is 2821.5, whereas in the FLMBBS the average is 3029.

In this experiment the FLMBBS algorithm improved the network lifetime by 7.35% compared to LEACH protocol.

Table 4.7: Experiment 4 - FND metrics for FLMBS and LEACH

Run	FLMBS FND	LEACH FND	Difference
1	3035	2776	259
2	2965	2843	122
3	3009	2816	193
4	2978	2775	203
5	3152	2867	285
6	3101	2841	260
7	3023	2805	218
8	3079	2817	262
9	2994	2798	196
10	2954	2877	77
average	3029	2821.5	207.5

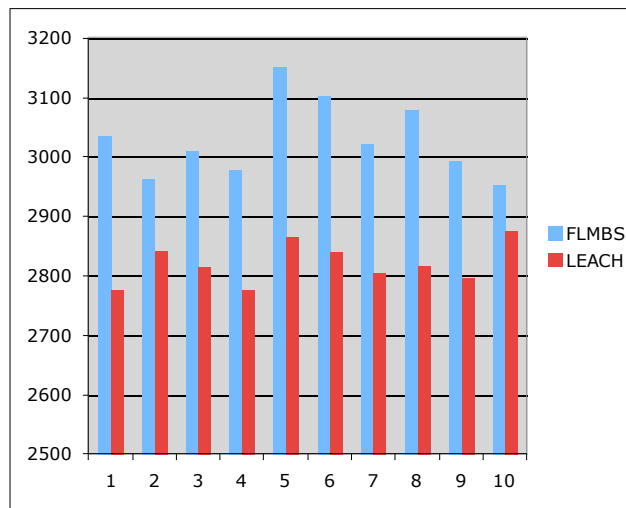


Figure 4.8: Experiment 4 - FND value for FLMBS and LEACH

T-test:

T-test was performed as before and the null hypothesis was rejected. The calculated p-value was 3.8×10^{-6} . Hence FLMBBS produced significant improvement over LEACH in terms of network lifetime. FLMBBS scaled reasonably well with respect to the number of nodes.

Data for the analysis of energy dissipation per round are not shown, as they were similar to the results in experiment 3.

4.3 Second Group of Experiments (FLMBBS vs LEACH vs PSO)

In this group of experiments we included the PSO algorithm and compared it with FLMBBS and LEACH. The PSO algorithm was explained in detail earlier in chapter 2. To recap, it involves a group of randomly generated particles moving through the network field (i.e. search space), and eventually settling on a near optimum BS location that would minimize the energy dissipation due to CH-to-BS transmission. In this section we performed different experiments, with different scenarios, in order to compare the performance of our proposed FLMBBS with PSO. FLMBBS and LEACH ran as before. The PSO algorithm (which obtains the optimum BS location in a given round) is inserted into the LEACH algorithm after CH selection and before CH transmission, i.e. the same place where the FIS evaluation and movement strategy in FLMBBS occurs.

In this series of experiments, we chose the initial energy of all nodes to be 1J or 500J depending on the size of the field, and the initial velocity of each particle to be zero for simplicity. We had chosen 10 starting particles and 100 iterations for the PSO algorithm, which was enough for the optimization result to plateau. Also, w , c_1 and c_2 were chosen to be 0.6, 1, and 1 respectively, values that were used in the author paper [37]. All other parameters were same as for experiments in the previous section.

Table 4.8: Input Parameters (Second group of experiments)

Parameters	Experiment 1	Experiment 2	Experiment 3
Size of the area	100m*100m	1000m*1000m	1000m*1000m
Init. BS location	(50,50)	(500,500)	(500,500)
Number of nodes	100	100	200
Initial energy	1J	500J	500J
Data rate	1-5	1-5	1-5
Number of Particles	10	10	10
Initial Velocity	0	0	0
Number of iteration	100	100	100

4.3.1 Experiment 1

In the first experiment our field dimension was $100 \times 100 \text{ m}^2$ and we randomly distributed 100 sensor nodes. The default BS location is at (50,50). This was the static BS location for LEACH, and the initial BS location for FLMBs and PSO in each round. Figure 4.9 shows the number of nodes that were alive for all three algorithms. In a typical run, the number of rounds

before the first node died in the FLMBS, PSO, and LEACH were 2691, 2718, and 2677 respectively. Figure 4.9 also shows the number of rounds before the last node. In the same run, the number of rounds completed before the last node dies in FLMBS, PSO, and LEACH were 3583, 3589, and 3598 respectively.

Since we are more concerned about the number of rounds before the first node dies we show the FND results for all three algorithms for the ten runs in table 4.9. The average value for FND was 2715 for FLMBS, 2706.80 for PSO, and 2696 for LEACH. The same data is plotted in Figure 4.10 for clarity.

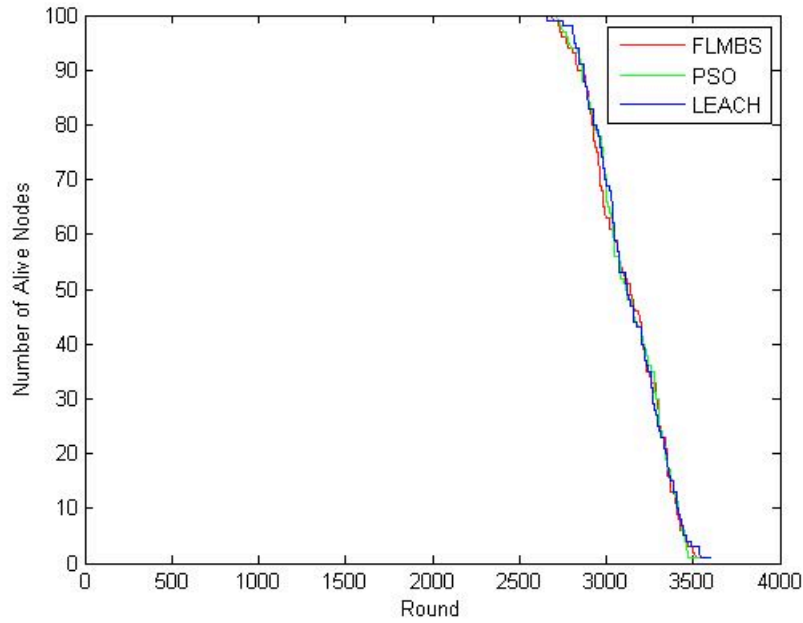


Figure 4.9: Experiment 1: The number of alive nodes for an area 100m x100m with 100SNs and BS at (50,50)

Table 4.9. Experiment 1 - FND metrics for all three protocols

Run	FLMBS FND	PSO FND	LEACH FND
1	2766	2706	2716
2	2731	2705	2690
3	2691	2718	2677
4	2726	2695	2704
5	2688	2693	2675
6	2714	2721	2729
7	2733	2717	2713
8	2722	2703	2674
9	2678	2699	2682
10	2701	2711	2700
Average	2715	2706.80	2696

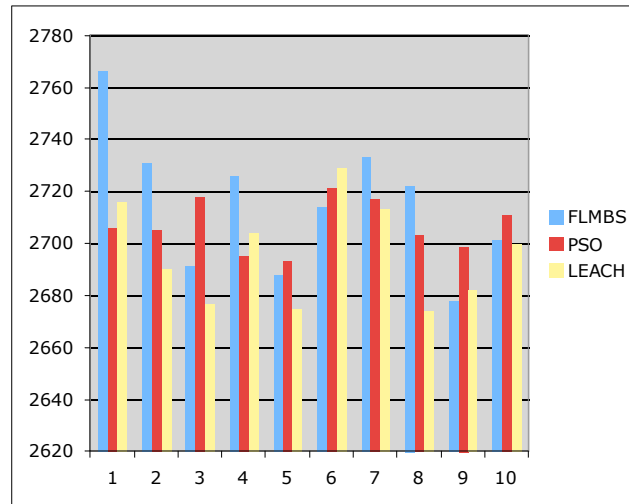


Figure 4.10: Experiment 1 - FND value for FLMBS and LEACH

T-test:

The paired t-test ($p=0.05$) was performed for FLMBS vs. LEACH, and PSO vs. LEACH, and the calculated p-values were 0.22 and 0.08 respectively. The null hypothesis was not rejected in either case and hence there was no significant difference between the 3 algorithms in terms of network lifetime on the $100 \times 100 \text{ m}^2$ grid. This was as expected, following our earlier observations about how reduction in transmission distance had relatively little impact on energy dissipation on the $100 \times 100 \text{ m}^2$ grid.

In the table below, we show the CH energy dissipation in one round, given the same network and the same CH's chosen, for FLMBS, PSO and LEACH. As can be seen, there was very little difference just like earlier, for the $100 \times 100 \text{ m}^2$ grid.

Table 4.10: Experiment 1 – Energy Dissipation after each round

RUN	FLMBS	PSO	LEACH
1	0.0044	0.0044	0.0046
2	0.0041	0.0041	0.0041
3	0.0042	0.0042	0.0042
4	0.0063	0.0064	0.0064
5	0.0048	0.0048	0.0048
6	0.0033	0.0033	0.0035
7	0.0036	0.0035	0.0036
8	0.0041	0.0041	0.0041
9	0.0037	0.0038	0.0039
10	0.0203	0.0202	0.0205
Avg	0.0059	0.0059	0.0060

4.3.2 Experiment 2

In the second experiment, we increased the size of the field to $1000 \times 1000 \text{ m}^2$ and deployed 100 sensor nodes randomly into the field. For this experiment the default BS location was at (500,500) and initial energy of all sensor nodes was set to 500J. Step size was scaled from 20 to 200.

Figure 4.11 shows the results from a typical run. The number of rounds before the first node dies in FLMBs, PSO, and LEACH were 1531, 1546, and 1426 respectively, while the number of rounds completed before the last node dies in FLMBs, PSO, and LEACH were 5331, 5444, and 5358 respectively.

FND results for all three algorithms for the ten runs are tabulated in Table 4.11. The average value for FND was 1522.20 for FLMBs, 1541.70 for PSO, and 1467.90 for LEACH. The same data is plotted in Figure 4.12 for clarity.

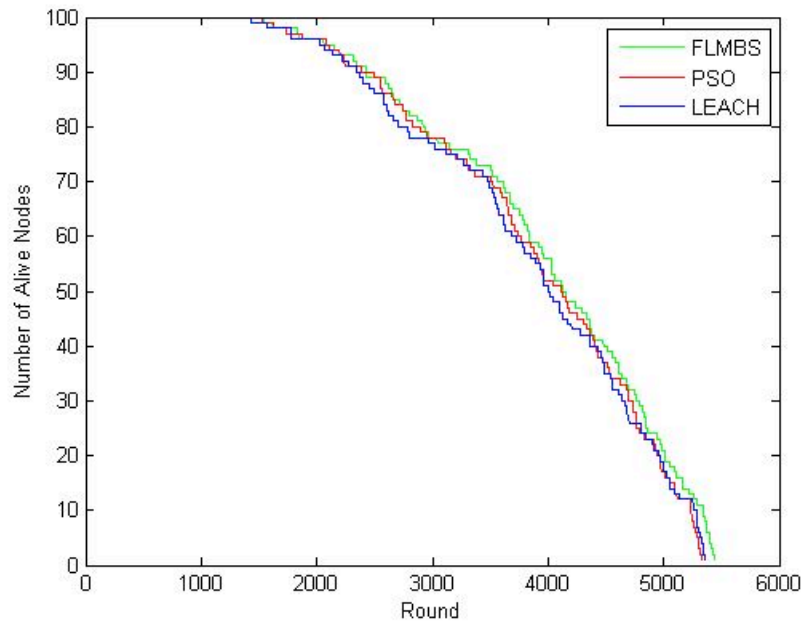


Figure 4.11: Experiment 2 - The number of alive nodes for an area 1000m x1000m with 100SNs and BS at (500,500)

Table 4.11. Experiment 2 - FND metrics for all three protocols

Run	FLMBS FND	PSO FND	LEACH FND
1	1531	1546	1426
2	1520	1513	1440
3	1544	1550	1432
4	1490	1563	1498
5	1524	1532	1480
6	1532	1570	1475
7	1519	1593	1567
8	1527	1515	1494
9	1496	1499	1463
10	1539	1541	1504
Average	1522.20	1541.70	1467.90

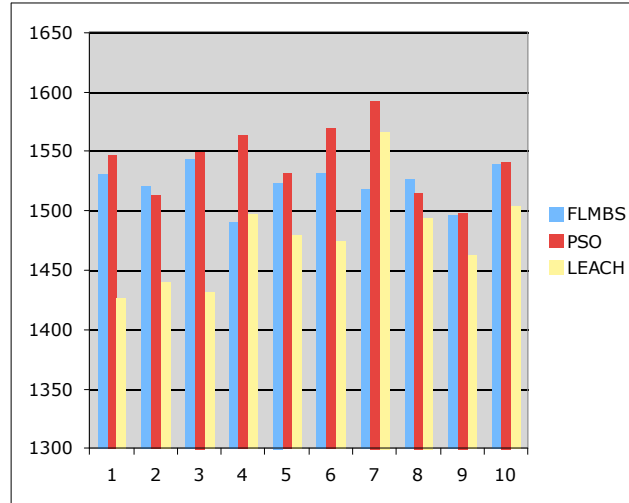


Figure 4.12: Experiment 2 - FND value for FLMBs and LEACH

Table 4.12: Experiment 2 – Energy Dissipation after each round

RUN	FLMBs	PSO	LEACH
1	11.687	11.484	11.789
2	13.179	13.227	13.356
3	12.382	11.412	14.484
4	12.801	12.678	12.822
5	16.819	15.264	18.047
6	13.566	12.481	13.432
7	11.293	11.003	11.719
8	11.623	11.024	14.304
9	11.152	10.985	11.611
10	13.666	13.869	14.32
Avg	12.816	12.342	13.588

T-test:

Using the T-test, we were able to show significant improvement of PSO over LEACH (p-value was 3.5×10^{-4}), and also a significant difference between FLMBBS and PSO (p-value was 0.0486), although for the latter the difference was barely significant because the calculated p-value was so close to the threshold of 0.05.

The network lifetime results were supported by analysis of the CH energy dissipation in one round. This is shown in table 4.12.

4.3.3 Experiment 3

In this final experiment, we evaluated the scalability of our model compared to both LEACH and PSO. We retained the $1000 \times 1000 \text{ m}^2$ grid but increased the number of sensor nodes to 200. The rest remained the same. Figure 4.13 shows the number of nodes that were alive for all three algorithms in a typical run of the experiment. The number of rounds before the first node dies in the FLMBBS, PSO, and LEACH were 2896, 3053, and 2792 respectively, while the number of rounds completed before the last node dies in FLMBBS, PSO and LEACH were 8185, 8025, and 8193 respectively.

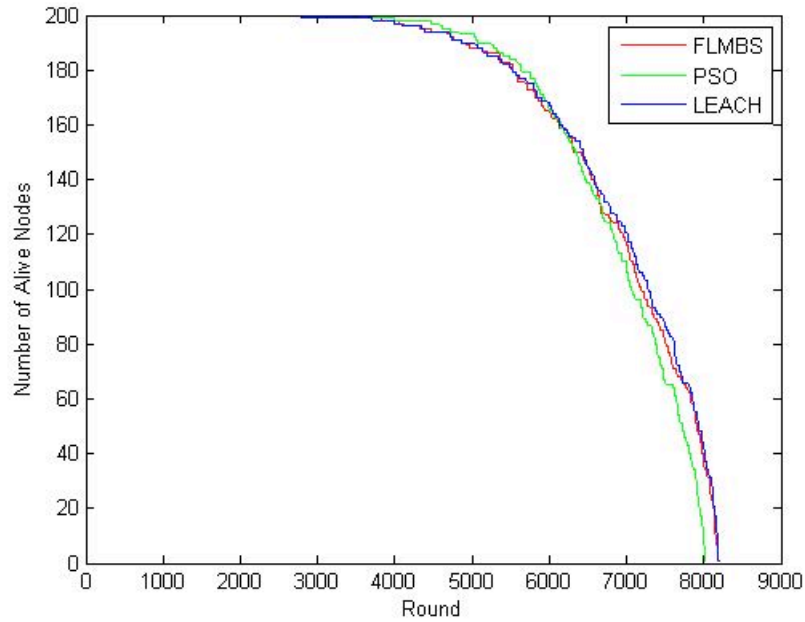


Figure 4.13: Experiment 3 - The number of alive nodes for an area 1000m x1000m with 200SNs and BS at (500,500)

Moreover we show the FND results for all three algorithms for the ten runs of the experiment in Table 4.13. The average value for FND is 3011 for FLMBs, 3170 for PSO, and 2816 for LEACH. The same data is plotted in Figure 4.14 for clarity. We can see that both FLMBs and PSO scaled well with respect to the number of nodes: improvement over LEACH increased with the number of nodes. This was as expected as more nodes meant greater lifetime and hence more time for the two algorithms to distance themselves from LEACH. T-test once more confirmed significant differences between the 3 algorithms.

Table 4.13. Experiment 3 - FND metrics for all three protocols

Run	FLMBS FND	PSO FND	LEACH FND
1	2950	3219	2797
2	2932	3181	2810
3	3004	3224	2825
4	2896	3053	2792
5	2966	3155	2829
6	3014	3207	2854
7	3104	3231	2832
8	3056	3046	2821
9	3079	3183	2787
10	3108	3196	2816
Average	3010.90	3169.50	2816.30

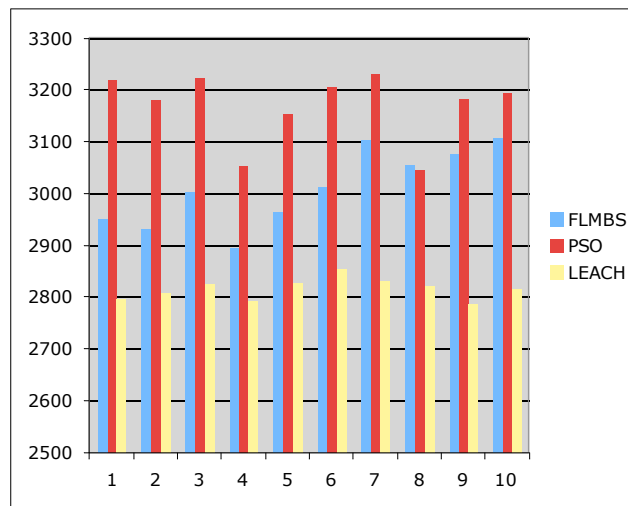


Figure 4.14: Experiment 3 - FND value for FLMBS, PSO and LEACH

4.4 Summary

In the first group of experiments we compared our proposed model with the original LEACH protocol. We first showed that for a fair and meaningful comparison, we must place the default BS location within the network field (we put it right in the middle). We then showed that FLMBBS could not produce significant improvement over LEACH on a $100 \times 100 \text{ m}^2$ network grid, but could do so on a larger grid (at least 5% improvement on a grid that's $1000 \times 1000 \text{ m}^2$ or larger). This was due to the way CH energy dissipation was calculated, and was supported by our analysis of CH energy dissipation in a single round. In the second group of experiments we compared our proposed model with another mobile BS approach, which is a population based stochastic optimization algorithm that is called particle swarm optimization (PSO). Here we showed once more that reduction in transmission distance only has a significant effect on a larger grid. PSO outperformed FLMBBS but not by a large margin, especially considering the relative simplicity of FLMBBS and the fact that PSO does a global search to obtain new BS location within each round. Both PSO and FLMBBS scaled well with respect to the number of nodes in the network.

CHAPTER V

Conclusions

Wireless sensor networks pose interesting challenges for networking research. Foremost among these is the development of long-lived sensor networks in spite of the energy-constraints of individual nodes. In this thesis we introduce the fuzzy logic based routing protocol (FLMBS), which is a modified version of the well-known LEACH protocol. LEACH uses random rotation of the nodes required to be the CHs, in order to evenly distribute energy consumption in the network. LEACH operation is divided into two consecutive phases. The first phase is called setup-phase in which the clusters are formed and a CH is chosen for each cluster. Then in the steady-state phase, data is sensed and then is sent to the central BS. Our proposed protocol outperformed LEACH by assigning a priority to each CH and adjusting the BS position according to the priorities of all CH's, at the end of the setup phase. In our approach, the CH priorities were calculated using fuzzy logic, with three fuzzy input parameters considered: residual energy of each CH, distance of each CH to the BS, and the rate at which each CH sends data to the BS. The popular Mamdani fuzzy inference system is used in our model that includes 27 set of rules and the centroid defuzzification method was performed to calculate the priority values.

Our movement strategy underwent a series of transformations during the course of our work, as we continually sought to improve it. Our initial strategy was different from what we proposed

in this thesis: instead of considering the priorities of all CH's, we only moved the BS towards the CH with the highest priority. We tested this in our very first experiment, where the default/starting BS location was outside the network field, and our initial movement strategy led to good improvement over LEACH in terms of network lifetime. However, as we discussed earlier, making the default BS location outside of the network field was not fair, since any movement strategy that moved the BS towards the inside of the network field would lead to much improvement over LEACH, where BS remains static outside of the field. Therefore, we decided to focus on experiments where the default BS location was inside the network field (in the middle of the field). Here, our initial movement strategy failed miserably. No improvement over LEACH was seen on the 100x100 or 1000x1000 grid, and in many cases the performance was worse (hence data not shown). This was understandable, as the movement strategy was too simple. When we only accounted for one CH, we may have improved the energy dissipation of that particular CH, but by ignoring the other CH's, the overall energy dissipation was often worse. Consequently, we developed a new movement strategy that incorporated the needs of all CH's. Within each round of the protocol, the BS now adjusts its position according to the priorities of all CH's. With our new movement strategy, we again needed to make some modifications, including varying the step size, modifying the FIS, etc. One important modification was resetting the BS location, at the beginning of each round, to the default location in the center of the field. This was performed because different CH's were chosen within each round, and hence it didn't make much sense for the final BS location of the previous round to carry over to the current round. Indeed, our tests showed better results when we always moved the BS from the center of the field (rather than moving it from its final

position in the previous round), for each batch of CH's chosen. These optimizations eventually led to the movement strategy being proposed in this thesis. It was the strategy that we found gave the most consistent results.

With our FIS-driven movement strategy, our proposed FLMBs protocol was capable of outperforming LEACH. There was no significant improvement in terms of CH energy dissipation or network lifetime on the 100x100 network field due to the way energy dissipation was calculated in the radio transmission model that we used (Group 1, experiment 2). However, in a larger grid (e.g. 1000x1000), we saw significant improvement: 5% network lifetime increase with 100 starting nodes, greater increase with more nodes (group 1, experiments 3 and 4). In order to evaluate our purposed movement strategy with another movement strategy, we decided to compare our FLMBs with a similarly modified LEACH protocol that used the PSO algorithm [30] to find the optimal BS location. We saw that this model outperformed the FLMBs but not by a large margin (group 2, experiments 1-3). Our protocol was reasonably successful given its performance and relative simplicity.

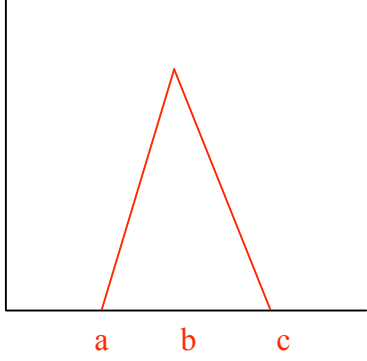
5.1 Future work

In future, some extensions of the FLMBs approach can be applied. By running additional simulations in a network with heterogeneous characteristics, where nodes have different initial energy, the performance of our proposed model can be further evaluated. Also additional improvements in the network life time and energy consumption could be reached by adjusting

the shape of each fuzzy parameter. Different defuzzification methods could be used as well in our fuzzy inference system to further examine the system results. Additional experiments must be performed using more than one BS as well as mobile sensors to assess the performance of our model within different network environments. Finally, the behavior of our proposed FLMBs can be investigated with some other clustering protocol such as TEEN[55], PEGASIS[56] and some other population based, metaheuristic algorithm such as ACO[32,33].

Appendix A: Triangular and Trapezoidal MFs

Triangle membership function is represented as follows:

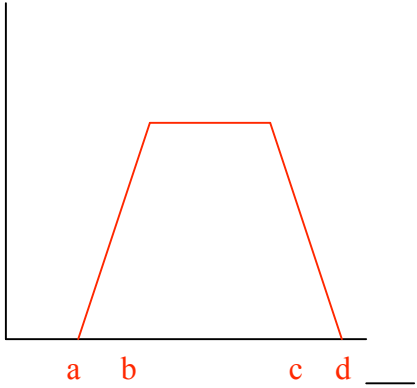
$$\mu(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (A.1)$$


Or, more precisely by using min and max, we have an alternative expression for the proceeding equation:

$$\mu_A(x) = \max(\min(\frac{x-a}{b-a}, \frac{c-x}{c-b}), 0) \quad (A.2)$$

Where, the point (a and c) locates the “feet” of the triangle and the point (b) locates the “top” of the triangle.

Trapezoidal membership function is presented as follows:

$$\mu(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{c-x}{c-b}, & c \leq x \leq d \\ 1, & d \leq x \end{cases} \quad (A.3)$$


Or, an alternative concise expression using min and max is:

$$\mu_A(x) = \max(\min(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}), 0) \quad (\text{A.4})$$

Where, the points (a and d) represent the “feet” of the trapezoidal and the points (b and c) represents the “shoulders” of the trapezoidal.

Appendix B: Distance and Energy MF for a Larger Grid

Below we have shown the MFs for the 1000 x 1000 m² search space, and initial energy of nodes at 500J.

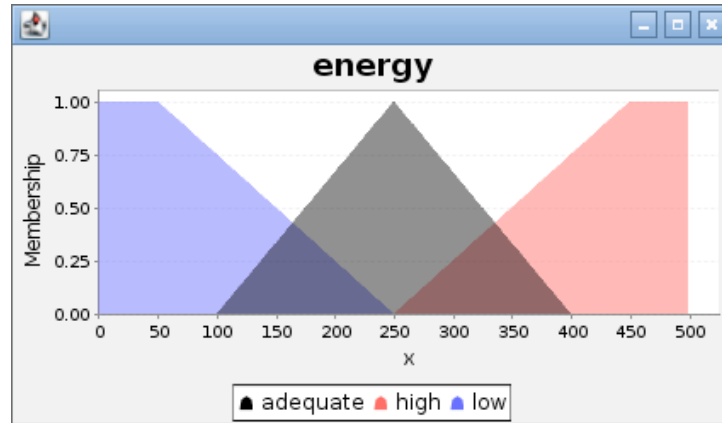


Figure B1: MF for energy in a 1000 x 1000 m²

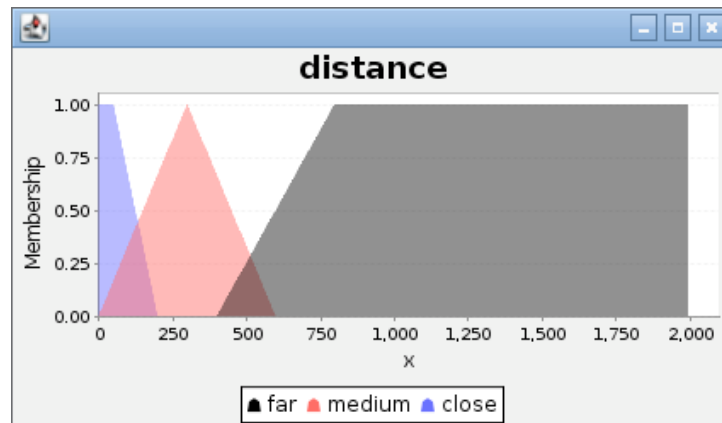


Figure B2: MF for distance in a 1000 x 1000 m²

There were no changes made to the input membership function Rate, the output membership function priority, and IF-THEN rules in a larger grid.

References

- [1] S.K. Singh, M.P. Singh, and D.K. Singh. A Survey of Energy-Efficient Hierarchical Cluster-Based Routing in Wireless Sensor Networks, *International Journal of Advanced Networking and Applications (IJANA)*, vol. 2, issue. 2, pp. 570-580, 2010
- [2] S.K. Singh, M.P. Singh, and D.K. Singh. Energy-efficient Homogeneous Clustering Algorithm for Wireless Sensor Network, *International Journal of Wireless & Mobile Networks (IJWMN)*, vol. 2, issue. 3, pp. 49-61, 2010.
- [3] P. Sethi, and N. Chauhan. Exploring the Feasibility of Mobile Agents in Sensor Networks in Non-Deterministic Environments, *International Journal of Advancements in Technology (IJICT)*, vol. 1, issue. 2, 2010.
- [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] P.P. Stavroulakis, and M. Stamp. Security in Wireless Sensor Networks, *Handbook of Information and Communication Security*, pp. 513-552, 2010.
- [6] C. Cirstea. Energy Efficient Routing Protocols for Wireless Sensor Networks: A Survey, *IEEE 17th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, pp. 277-282, 2011.
- [7] W.R. Heinzelman, A. Chandrakasan and H. Balakrishnan. Energy-efficient Communication Protocol for Wireless Sensor Networks, *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, vol.2, pp. 3005-3014, 2000.

- [8] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey, *Computer Networks*, vol. 38, issue. 4, pp. 393-422, 2002.
- [9] J.P. Lynch, and K. Loh. A Summary review of wireless sensors and sensor networks for structural health monitoring. *Shock and Vibration Digest*. Sage Publication, pp. 91-128, 2005.
- [10] X. Li, L. Xu, H. Wang, J. Song and S.X. Yang. A Differential Evolution-Based Routing Algorithm for Environmental Monitoring Wireless Sensor Networks. *Sensors*, vol. 10, issue. 6, pp. 5425-5442, 2010.
- [11] Y. Wang, G. Attebury, and B. Ramamurthy. A Survey of Security Issues in Wireless Sensor Networks, *IEEE Communications Surveys & Tutorials*, vol. 8, issue. 2, 2006.
- [12] K. Maraiya, K. Kant, and N. Gupta. Application based Study on Wireless Sensor Network, *International Journal on Computer Applications*, vol. 21, issue. 8, pp. 9-15, 2011.
- [13] F.G. Nakamura, F.P. Quintao, G.C. Menezes, and G.R. Mateus. An Optimal Node Scheduling for flat Wireless Sensor Networks. *In Proceedings of the IEEE International Conference on Networking (ICN05)*, vol. 3420, pp. 475-483, 2005.
- [14] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*, Wiley-Interscience, Hoboken, NJ, 2007.
- [15] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A Survey on Sensor Networks. *IEEE Communications Magazine*, vol. 40, pp. 102–114, 2002.
- [16] S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman. A Taxonomy of Wireless Micro-sensor Network Models, *ACM Mobile Computing and Communication Review*, vol. 6, issue. 2, pp. 28–36, 2002.
- [17] S. Ramanathan and M. Steenstrup. A survey of routing techniques for mobile

communications networks, *Mobile Networks and Applications*, vol. 1, issue. 2, pp. 89–104, 1996.

[18] O. Younis, M. Krunz, and S. Ramasubramanian. Node Clustering in Wireless Sensor Networks: Recent Developments and Deployment Challenges. *IEEE Network*, vol. 20, issue. 3, pp.20– 25, 2006.

[19] J.Y. Yu, and P.H.J. Chong. A survey of clustering schemes for mobile ad hoc networks, *IEEE Communications Surveys & Tutorials*, vol. 7, issue. 1, pp. 32-48, 2005.

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

[21] A.B. McDonald, and T. Znati. A Mobility Based Framework for Adaptive Clustering in Wireless Ad-Hoc Networks, *IEEE Journal on Selected Areas in Communications*, vol. 17, issue. 8, pp. 1466-1487, 1999.

[22] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang. A Two-Tier Data Dissemination Model for Large Scale Wireless Sensor Networks, *Proceedings of the 8th annual international conference on Mobile Computing and Networking*, pp. 148-159, 2002.

[23] A.D. Amis, R. Prakash, T.H.P. Vuong, and D.T. Huynh. Max-min d-cluster formation in wireless sensor networks, *Nineteenth Annual Joint Conference of the IEEE Computer and Communication Societies*, vol. 1, pp. 32-41, 2000.

[24] J. Ibriq, and I. Mahgoub. Cluster-based Routing in Wireless Sensor Networks: Issues and Challenges, *Proceedings of the 2004 Symposium on Performance Evaluation of Computer Telecommunication Systems*, pp. 759-766, 2004.

- [25] J. Wan, D. Yuan, and X. Xu. A Review of Cluster Formation Mechanism for Clustering Routing Protocols, 11th *IEEE International Conference on Communication Technology (ICCT)*, pp. 611-616, 2008.
- [26] J. Ibriq, and I. Mahgoub, Cluster-based Routing in Wireless Sensor Networks: Issues and Challenges, *Proceedings of the 2004 Symposium on Performance Evaluation of Computer Telecommunication Systems*, pp. 759-766, 2004.
- [27] C.R. Lin, and M. Gerla. Adaptive clustering for mobile wireless networks, *IEEE Journal of Selected Areas in Communications*, vol. 15, issue. 7, pp. 1265-1275, 1997.
- [28] G. Chen, and I. Stojmenovic. Clustering and routing in mobile wireless networks, *Technical report, TR-99-05, Dept. Computer Science, SITE*, University of Ottawa, 1999.
- [29] O. Younis, and S. Fahmy. HEED: a hybrid, energy-efficient, distributed clustering approach for ad-hoc sensor networks, *IEEE Transaction on Mobile Computing*, vol. 3, issue. 4, pp. 366-379, 2004.
- [30] W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan. An Application-specific Protocol Architecture for Wireless Microsensor Networks, *IEEE Transaction on Wireless Communication*, vol. 1, issue. 4, pp. 660-670, 2002.
- [31] A.E. Smith. Swarm Intelligence: from natural to artificial systems, *IEEE Transaction on Evolutionary Computation*, vol. 4, issue. 2, pp. 192-193, 2000.
- [32] M. Dorigo. "Optimization, Learning and Natural Algorithms", Ph.D Thesis, Department of Electronic and Information, Polytechnic University of Milan, IT, 1992.
- [33] M. Dorigo, V. Maniezzo, and A. Colorni. Ant system: Optimization by a colony of cooperating agents, *IEEE Transaction on Systems, Man, and Cybernetics-Part B*, vol. 26,

issue. 1, pp. 29-41, 1996.

[34] J. Kennedy, R. Eberhart. Particle Swarm Optimization, *Proceedings IEEE International Conference on Neural Networks*, vol. 4, pp. 1942-1948, 1995.

[35] R. Eberhart, J. Kennedy. A new optimizer using particle swarm theory, *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 1995.

[36] G. Liang, and H.L. Bertoni. A new approach to 3-D ray tracing for propagation prediction in cities, *IEEE Transaction on Antennas and Propagation*, vol. 46, issue. 6, pp. 853-863, 1998.

[37] T.P. Hong, G.N. Shiu, and Y.C. Lee. A PSO heuristic algorithm for base-station locations, *In The Joint Conference of the Third International Conference on Soft Computing and Intelligent Symposium on Advanced Intelligent Systems*, Japan, 2006.

[38] L.A. Zadeh, "Fuzzy Sets", *Information and Control*, vol. 8, issue. 3, pp.338-353, 1965.

[39] L.A. Zadeh, "Fuzzy Algorithms", *Information and Control*, vol. 12, issue. 2, pp. 94-102, 1968.

[40] L.A. Zadeh, Outline of a New Approach to the Analysis of Complex systems and Decision Processes, *IEEE Transaction on Systems, Man, and Cybernetics*, vol. 3, issue. 1, 1973.

[41] I. Gupta. Cluster-Head Election Using Fuzzy Logic for Wireless Sensor Networks, Master Thesis, Department of Computer Science, Dalhousie University of Halifax, 2005.

[42] Matlab fuzzy logic toolbox.

[43] T. Mitsuishi, N. Endou, and Y. Shidama. The concept of fuzzy set and membership function and basic properties of fuzzy set operation, *Formalized Mathematics*, vol. 9, issue. 2, pp. 351-356, 2001.

[44] E.H. Mamdani, and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic

controller, *International Journal of Man-Machine Studies*, vol. 7, issue. 1, pp. 1-13, 1975.

[45] L.A. Zadeh. The Concept of a Linguistic variable and its application to approximate reasoning-I, *Information Sciences*, vol. 8, issue. 3, pp.199-249, 1975.

[46] M. Sugeno. Industrial Application of Fuzzy Control. Elsevier Science Publication. Co. Amsterdam, 1985.

[47] M. Negnevitsky, Artificial Intelligence: A Guide to Intelligent Systems, Addison Wesley, Reading, MA, 2001.

[48] G. Ran, H. Zhang, and S. Gong. Improving on LEACH Protocol of Wireless Sensor Networks using Fuzzy Logic, *Journal of Information & Communication Science*, no. 7, issue. 3, pp. 767-775, 2010.

[49] A. Alkesh, A.K. Singh, and N. Purohit. A Moving Base Station Strategy Using Fuzzy Logic for Lifetime Enhancement in Wireless Sensor Network, *International Conference on Communication Systems and Network Technologies (CSNT)*, pp. 198-202, 2011.

[50] N.A Torghabeh, M.R.A. Totonchi, and M.H.Y. Moghaddam. Mobile Base Station Management using Fuzzy Logic in Wireless Sensor Networks, *Second International Conference on Computer Engineering and Technology (ICCET)*, vol. 2, pp. 352-356, 2010.

[51] B.C. Arabacioglu. Using Fuzzy Inference System for Architectural Space Analysis, *Applied Soft Computing Journal*, vol. 10, issue 3, pp. 926-937, 2010.

[52] R.C. Penney. Linear Algebra: Ideas and Applications, 3rd Edition, Wiley, Department of Mathematics, Purdue University, Indiana, 2008.

[53] US Naval Observatory (USNO) GPS Operations, <http://tycho.usno.navy.Mil/gps.html>, 2001.[Accessed 25 Dec 2011]

- [54] Matlab version 12, www.mathworks.com. [Accessed 19 Feb 2012]
- [55] A. Manjeshwar, D.P. Agrawal. TEEN: A Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks, *Proceedings 15th International Parallel and Distributed Processing Symposium*, pp. 2009-2015, 2001.
- [56] S. Lindsey, and C.S. Raghavendra. PEGASIS: Power-Efficient gathering in Sensor Information Systems, *In Proceedings of IEEE Aerospace Conference*, vol. 3, pp. 3-1125 – 3-1130, 2002.