The development in wireless technological innovations and various methods for wireless transmission result in the introduction of the wireless sensor networks (WSNs). A WSN typically contains hundreds or thousands of tiny sensor nodes (SNs) and a base station (BS) which is responsible for receiving the collected data from the SNs. These tiny nodes typically are battery-powered, and have limited power, memory and processing capability. Either due to random and wide distribution of SNs, or deploying them in inaccessible location recharging or changing the batteries of SNs become impractical. WSNs can be used in number of demanding applications such as environmental, biomedical, constructive, military applications etc. WSNs are expected to be joined into the ‘Internet of Things’ where the WSNs join the Internet, and utilize it to work together and achieve their tasks. Due to the fact that all SNs are power constrained devices, energy consumption of nodes throughout transmission or reception of packets plays a vital role in the lifetime of the network. Therefore it is very important to design energy-efficient routing protocols in order to prolong the network’s lifetime by organizing and selecting the most suitable paths in between sensor nodes. The properties of the environment where SNs are deployed and intense resources such as power limitation, make the design of routing protocols very interesting and challenging task. Several routing algorithms have been enhanced and developed in this regard. One of the most significant routing strategy for WSNs is the hierarchal or cluster-based routing. In hierarchical routing protocol nodes with high energy are used to send data to the BS while the nodes with low energy are used to sense and collect data of a specific area.

Using intelligent approaches such as fuzzy logic within the routing protocols could greatly improves the efficiency of WSNs. The main concern of this thesis is to take advantage of fuzzy logic approach in design of cluster-based routing protocols in
order to improve the energy consumption, throughput and overall performance of WSNs.

The overall network performance can also be influenced by the place of the SN as the sensed data, collected by the ordinary nodes have to be transferred to the SN. Hence in part of this thesis the impact of SN on the network performance has been investigated and additionally a novel algorithm to find the optimal place for the sink node has been proposed.
# TABLE OF CONTENTS

## CHAPTER ONE: INTRODUCTION

1.1 INTRODUCTION ........................................................................................................ 14
1.2 RESEARCH OBJECTIVES ...................................................................................... 15
1.3 THESIS CONTRIBUTIONS ..................................................................................... 15
1.4 STRUCTURE OF THE THESIS .............................................................................. 16

## CHAPTER TWO: WIRELESS SENSOR NETWORKS: AN OVERVIEW

2.1 INTRODUCTION ........................................................................................................ 18
2.2 COMPARISON WITH Ad Hoc NETWORKS .......................................................... 19
2.3 TECHNICAL CHALLENGES OF WSNS .............................................................. 20
2.4 DESIGN METRICS ............................................................................................... 20
2.5 TYPES OF WIRELESS SENSORS ........................................................................ 21
2.6 APPLICATIONS ..................................................................................................... 22
   2.6.1 Environmental Monitoring ............................................................................ 22
   2.6.2 Health Care ................................................................................................... 24
   2.6.3 Military Application ...................................................................................... 25
   2.6.4 Pipeline Monitoring ...................................................................................... 25
   2.6.5 Structural Health Monitoring (SHM) ............................................................. 25
   2.6.6 Industrial application .................................................................................... 26
   2.6.7 Smart Buildings ........................................................................................... 26
2.7 HARDWARE PLATFORMS ....................................................................................... 27
   2.7.1 mica2 ............................................................................................................. 27
   2.7.2 MicaZ ............................................................................................................ 28
   2.7.3 TelosB ............................................................................................................ 30
2.8 OPERATING SYSTEMS ......................................................................................... 31
   2.8.1 TinyOS .......................................................................................................... 31
   2.8.2 Contiki ........................................................................................................... 32
   2.8.3 MANTIS ........................................................................................................ 33

## CHAPTER THREE: BACKGROUND AND RELATED WORK

3.1 INTRODUCTION ..................................................................................................... 37
3.2 ROUTING CHALLENGES AND DESIGN ISSUES IN WSN ............................... 37
3.3 ROUTING CLASSES IN WSNs ............................................................................ 39
3.3.1 Data Centric Routing Protocols ................................................................. 39
3.3.2 Location Based Routing Protocols ............................................................. 45
3.3.3 Hierarchical Routing Protocols ................................................................. 48
3.4 CLUSTERING TECHNIQUES .......................................................................... 54
3.5 FUZZY LOGIC ................................................................................................. 56
3.5.1 Advantages of Fuzzy Logic controller ......................................................... 57
3.5.2 Crisp and Fuzzy Sets .................................................................................... 58
3.5.3 Properties of Fuzzy Sets ............................................................................... 60
3.5.4 Operations on Fuzzy Sets ............................................................................ 60
3.5.5 Linguistic Variables ..................................................................................... 63
3.5.6 Membership Functions ............................................................................... 63
3.5.7 Fuzzy IF-THEN Rules ................................................................................ 65
3.5.8 Fuzzy Inference System .............................................................................. 66
3.6 RELATED WORK ............................................................................................. 70

CHAPTER FOUR: PERFORMANCE ANALYSIS OF CLUSTERING ROUTING ALGORITHMS BASED ON FUZZY LOGIC

4.1 INTRODUCTION ................................................................................................. 75
4.2 GUPTA FUZZY PROTOCOL ............................................................................. 75
4.3 CHEF FUZZY PROTOCOL ............................................................................... 76
4.4 FL-BASED CH SELECTION SYSTEM FOR WSNs ........................................ 76
4.5 SIMULATION AND PERFORMANCE ANALYSIS ........................................... 77
4.5.1 Lifetime of Network .................................................................................... 78
4.5.2 Energy Dissipation Of The Network ............................................................. 78
4.5.3 Throughput of Network ............................................................................. 79
4.5.4 Overall Network performance ................................................................. 79
4.6 Conclusion ........................................................................................................ 79

CHAPTER FIVE: TWO-STEP FUZZY LOGIC SYSTEM TO ELECT THE OPTIMAL CHs

5.1 INTRODUCTION ................................................................................................. 82
5.2 RELATED WORK .............................................................................................. 83
5.3 SYSTEM MODEL .............................................................................................. 85
5.4 PROPOSED ALGORITHM ............................................................................... 86
5.5 FIRST-STEP OF FL SYSTEM ........................................................................... 86
5.5.1 FIS Design ................................................................................................... 86
5.5.2 FIS Parameters and Rules (First-Step) ......................................................... 86
5.5.3 Chance determination of nodes in the First-Step FL system: ....................... 90
LIST OF PUBLICATIONS

This thesis represents my research work carried out during my Ph.D. degree at Department of Computer Science, Aligarh Muslim University. Some parts of this thesis have appeared in the following papers which I got published during my Ph.D. degree.

**International Journals:**


3. Bokhari, Mohammad Ubaidullah, Yahya Kord Tamandani, and Qahtan Makki. “**Bokhari-GM Algorithm** to Locate the Sink Node with the Aim of extending the lifetime And throughput Of WSNs” (Accepted in (Elsevier journal) Egyptian informatics)


**International Conferences Proceedings:**


ACKNOWLEDGMENT

“Allamal-insana malam ya’lam”
(‘Teacheth the man, which he knew not’)

In the name of Allah the most beneficent and merciful, who blessed me with strength, whose benign benediction gave me the required zeal for the completion of this work.

It is my pleasant duty to acknowledge my debt of gratitude to my learned supervisor Professor Mohammad Ubaidullah Bokhari, Chairman, Department of Computer Science, AMU, for his enlightening, scholarly guidance and inspiration right from the start till the accomplishment of this work. He has been a source of confidence, encouragement and strength to me.

I would like to thank my fellow research scholars Qahtan, Haidar, Hasan, Oqail, Nazhir, Subhi and the rest of my friends for their advices and support during my research work. I am highly thankful to the Department of Computer Science, Aligarh Muslim University for providing all necessary facilities for the successful completion of my research.

Apart from the above, I would like to express my heartfelt gratitude to my family, parents, brothers and sisters, especially to my wife and beautiful daughter for being with me and supporting me through all the difficulties. A special thank goes to my wonderful father, who trusted me, without his love trust and support nothing would have been accomplished. I would like to dedicate my thesis to my adorable parents. I owe them a lot and without their love, affection, encouragement and support this journey was not possible.

(Yahya Kord Tamandani)
LIST OF FIGURES

Figure 2.1 Architecture of a typical Sensing node .......................................................... 19
Figure 2.2 Classification of environmental monitoring .................................................. 23
Figure 2.3 Mica2 and its components ......................................................................... 27
Figure 2.4 Micaz Sensor Node .................................................................................... 29
Figure 2.5 Components of a sensor node TelosB ....................................................... 30
Figure 2.6 Architecture of TinyOS ............................................................................ 32
Figure 2.7 Architecture of Contaki Operating System ............................................... 33
Figure 2.8 Architecture of MANTIS OS ................................................................. 34
Figure 3.1 Classification of Routing protocols in WSNs.......................................... 40
Figure 3.2. Working procedure of SPIN protocol ...................................................... 41
Figure 3.3. Working procedure of Direct Diffusion protocol ..................................... 42
Figure 3.4. Working procedure of the Rumor routing algorithm ................................. 43
Figure 3.5. Virtual grid in GAF ................................................................................ 47
Figure 3.6. Modules of SPEED routing protocol ..................................................... 47
Figure 3.7. GEAR network ...................................................................................... 48
Figure 3.8. Cluster formation in LEACH routing protocol ......................................... 50
Figure 3.9. LEACH setup phase .............................................................................. 52
Figure 3.10. Steady phase in LEACH ...................................................................... 52
Figure 3.11. Typical architecture of PEGASIS routing protocol ................................. 53
Figure 3.12. Self-Organizing protocol ..................................................................... 55
Figure 3.13 Clustering architecture ......................................................................... 57
Figure 3.14 classification of clustering algorithms .................................................. 58
Figure 3.15 example of Crisp set .............................................................................. 61
Figure 3.16. A typical fuzzy set ............................................................................. 62
Figure 3.17. Example of fuzzy set ........................................................................... 62
Figure 3.18. Intersection (AND logic operator) ......................................................... 63
Figure 3.19. Union (OR logic operator) ................................................................... 64
Figure 3.20. Complement (NOT logic operator) ....................................................... 64
Figure 3.21. Hierarchical structure the linguistic variable ‘AGE’ ............................... 65
Figure 3.22 Membership Functions for temperature ............................................... 66
Figure 3.23. Commonly used membership functions .............................................. 67
Figure 3.24. Interpretation of an if-then rule ............................................................ 68
Figure 3.25. Fuzzy Inference System ..................................................................... 69
Figure 3.26. Fuzzy inference approaches .................................................................. 70
Figure 3.27 Defuzzification process ......................................................................... 70
Figure 4.1. Number of dead nodes vs. Rounds ............................................................. 77
Figure 4.2. Energy Dissipation of Network vs. Rounds ............................................. 78
Figure 4.3. Packet sent to the BS vs. Rounds ............................................................. 80
Figure 5.1. Radio energy dissipation model (TWO-STEP FUZZY) ............................. 85
Figure 5.2. Fuzzy Inference System of First-Step (TWO-STEP FUZZY) .................. 86
Figure 5.3. Membership functions of Residual Energy (TWO-STEP FUZZY)........87
Figure 5.4. Membership functions of Node Density (TWO-STEP FUZZY)........87
Figure 5.5. Membership functions of Distance to BS (TWO-STEP FUZZY)........88
Figure 5.6. Membership functions of Qualification in First-Step of FL system........88
Figure 5.7. Relationship among fuzzy inputs in First-Step of FL system........89
Figure 5.8. Fuzzification of crisp Energy level (=0.03 J).................................91
Figure 5.9. Fuzzification of crisp Density (=30).............................................92
Figure 5.10. Fuzzification of crisp distance to the BS (=100 m).......................93
Figure 5.11. Output evaluation of fuzzy if-then rules of First-Step of FL system....95
Figure 5.12. The Centroid point for first-step of FL system............................97
Figure 5.13. Fuzzy Inference System of Second-Step.......................................98
Figure 5.14. Membership functions of Vulnerability Index..............................99
Figure 5.15. Membership functions of Centrality............................................99
Figure 5.16. Membership functions of Distance B/W CHs.............................99
Figure 5.17. Membership functions of Qualification in Second-Step of FL system100
Figure 5.18. Relationship among fuzzy inputs in Second-Step of FL system.........100
Figure 5.19. Fuzzification of crisp Vulnerability index (=0.07).......................103
Figure 5.20. Fuzzification of crisp Centrality (=0.75)....................................104
Figure 5.21. Fuzzification of crisp distance between CHs (=10 m).................105
Figure 5.22. Output evaluation of Second-Step of FL system..........................107
Figure 5.23. The Centroid point for Second-Step of FL system.......................108
Figure 5.24. Number of alive nodes (network size 60x60)..............................110
Figure 5.25. Number of alive nodes (network size 120x120)............................110
Figure 5.26. Residual energy of the WSN (network size 60x60).......................111
Figure 5.27. Residual energy of the WSN (network size 120x120).....................111
Figure 5.28. Energy variance of the WSN (network size 60x60).......................112
Figure 5.29. Energy variance of the WSN (network size 120x120)...................112
Figure 6.1. Fuzzy Inference System SEPFL................................................120
Figure 6.2. Membership functions of Battery Level (BOKHARI -SEPFL)...........120
Figure 6.3. Membership functions of Distance (BOKHARI -SEPFL)................120
Figure 6.4. Membership functions of Node Density (BOKHARI -SEPFL)..........121
Figure 6.5. Membership functions of Qualification in FLS (BOKHARI -SEPFL)121
Figure 6.6. The detailed flow chart of the proposed protocol (BOKHARI -SEPFL124
Figure 6.7. No. of Alive Nodes vs. Rounds (BOKHARI -SEPFL).......................126
Figure 6.8. Energy Dissipation of Network vs. Rounds (BOKHARI -SEPFL)........127
Figure 6.9. Packet sent to the BS vs. Rounds (BOKHARI -SEPFL)....................127
Figure 7.1. A typical Wireless Networks..................................................131
Figure 7.2. Arrangement of sensor nodes within filed (BOKHARI -GM)..............135
Figure 7.3. Number of Nodes Dead vs. Number of rounds (BOKHARI -GM)....137
Figure 7.4. Nodes death Vs. Number of Rounds (BOKHAI-GM).....................137
Figure 7.5. Energy Dissipation of WSN (BOKHARI-GM)...............................138
Figure 7.6. Closer view of Energy Dissipation of WSN (BOKHARI -GM)...........138
Figure 7.7. Packet sent to the BS (BOKHARI -GM)....................................139
LIST OF TABLES

Table 2.1 Summary of the mica2 platform features ................................................... 28
Table 2.2 Summary of the micaz platform features .................................................. 29
Table 2.3 Summary of the TelosB platform features ................................................ 30
Table 3.1 Comparison of the main data centric routing protocols ............................. 45
Table 3.2 Comparison of some location-based routing protocols ......................... 49
Table 3.3 Comparison of the main hierarchical routing protocols .................... 55
Table 3.4 Rules that a driver needs to follow (fuzzy logic concept) ...................... 59
Table 3.5 Properties of fuzzy logic ........................................................................ 62
Table 4.1 Various parameters used in the simulation ........................................ 77
Table 5.1 Input Functions (First-Step of FL system) ........................................... 87
Table 5.2 Output Functions (First-Step of FL system) ......................................... 88
Table 5.3 Fuzzy If-then rules in First-Step of FL system ...................................... 89
Table 5.4 Fuzzy Inference System IF-THEN rules ............................................. 94
Table 5.5 Priority value for different SNs in First-step FL system ........................ 97
Table 5.6 Input Functions (Second-Step of FL system) ....................................... 98
Table 5.7 Output Functions (Second-Step of FL system) .................................... 100
Table 5.8 Fuzzy If-then rules in Second-Step of FL system ................................. 100
Table 5.9 IF-THEN rules of Second-Step of FL system .................................... 105
Table 5.10 Priority Value For Different SNs To Be CH .................................... 109
Table 5.11 Parameters used for simulation ......................................................... 109
Table 6.1 Input Functions (BOKHARI-SEPFL) ................................................. 120
Table 6.2 Output Functions (BOKHARI-SEPFL) ............................................... 121
Table 6.3 Fuzzy If-then Rules (BOKHARI-SEPFL) ............................................ 122
Table 6.4 Simulation Parameters (BOKHARI-SEPFL) ....................................... 125
Table 6.5 Comparison of algorithms results (BOKHARI-SEPFL) .................... 126
Table 7.1 Various parameters used in the simulation (BOKHARI-GM) .......... 136
Table 7.2 Various locations for the base station (BOKHARI-GM) ..................... 136
# LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th></th>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>WSNs</td>
<td>Wireless Sensor networks</td>
</tr>
<tr>
<td>2</td>
<td>SNs</td>
<td>Sensor nodes</td>
</tr>
<tr>
<td>3</td>
<td>CHs</td>
<td>Cluster heads</td>
</tr>
<tr>
<td>4</td>
<td>BS</td>
<td>Base Station</td>
</tr>
<tr>
<td>5</td>
<td>SN</td>
<td>Sink node</td>
</tr>
<tr>
<td>6</td>
<td>GA</td>
<td>Genetic algorithm</td>
</tr>
<tr>
<td>8</td>
<td>TEEN</td>
<td>Threshold sensitive Energy Efficient Sensor Network Protocol</td>
</tr>
<tr>
<td>9</td>
<td>PEGASIS</td>
<td>Power-Efficient Gathering in Sensor Information Systems</td>
</tr>
<tr>
<td>10</td>
<td>LEACH</td>
<td>Low Energy Adaptive Clustering Hierarchy</td>
</tr>
<tr>
<td>11</td>
<td>REAR</td>
<td>Reliable Energy Aware Routing</td>
</tr>
<tr>
<td>12</td>
<td>ACQUIRE</td>
<td>Active Query forwarding in the WSNs</td>
</tr>
<tr>
<td>13</td>
<td>SPIN</td>
<td>Security Protocols for Sensor Networks</td>
</tr>
<tr>
<td>14</td>
<td>OS</td>
<td>Operating system</td>
</tr>
<tr>
<td>15</td>
<td>IWSNs</td>
<td>Industrial Wireless Sensor Networks</td>
</tr>
<tr>
<td>16</td>
<td>SHM</td>
<td>Structural Health Monitoring</td>
</tr>
<tr>
<td>17</td>
<td>GI</td>
<td>Global identification</td>
</tr>
<tr>
<td>18</td>
<td>FLC</td>
<td>Fuzzy logic control</td>
</tr>
<tr>
<td>19</td>
<td>FL</td>
<td>Fuzzy Logic</td>
</tr>
<tr>
<td>20</td>
<td>FIS</td>
<td>Fuzzy inference system</td>
</tr>
<tr>
<td>21</td>
<td>CHEF</td>
<td>Cluster Head Election mechanism using Fuzzy Logic in Wireless Sensor Networks</td>
</tr>
<tr>
<td>22</td>
<td>SEP</td>
<td>Stable Election Protocol</td>
</tr>
<tr>
<td>23</td>
<td>COD</td>
<td>Centre of Gravity</td>
</tr>
<tr>
<td>24</td>
<td>GM</td>
<td>Geometric median</td>
</tr>
<tr>
<td>25</td>
<td>GAF</td>
<td>Geographic Adaptive Fidelity</td>
</tr>
<tr>
<td>26</td>
<td>BGR</td>
<td>Blind Geographic Routing</td>
</tr>
<tr>
<td>27</td>
<td>EAGPR</td>
<td>Energy Aware Geographic Routing Protocol</td>
</tr>
<tr>
<td>28</td>
<td>GEAR</td>
<td>Geographic and Energy Aware Routing</td>
</tr>
<tr>
<td>29</td>
<td>TTDD</td>
<td>Two-Tier Data Dissemination</td>
</tr>
<tr>
<td>30</td>
<td>EEFS</td>
<td>Energy Efficient Forwarding Strategies for Geographic Routing</td>
</tr>
<tr>
<td>31</td>
<td>MDSAP</td>
<td>Modified Directional Source Aware Routing Protocol</td>
</tr>
<tr>
<td>32</td>
<td>DSAP</td>
<td>Directional Source Aware Routing Protocol</td>
</tr>
<tr>
<td>33</td>
<td>IGF</td>
<td>Implicit Geographic Forwarding</td>
</tr>
<tr>
<td>34</td>
<td>LND</td>
<td>Last node dies</td>
</tr>
</tbody>
</table>
CHAPTER ONE

INTRODUCTION
1.1 INTRODUCTION

The primary concept of anytime and anywhere processing results in the new subject known as mobile computing. The improvement in wireless technological innovations are additionally among the main causes for the development and popularity of mobile computing. The progress in these tiny computing model and various approaches for wireless transmission result in the introduction of the wireless sensor networks (WSNs). Sensor networks are required in the applications such as industrial control units, environmental monitoring, military applications and structural health monitoring applications. Because of the fact that all sensor nodes are energy constrained devices, power consumption of nodes throughout transmission or reception of packets plays a vital role in the life-time of the network. In order to make routing, energy efficient, wide range of protocols have been designed and introduced. Sensor nodes are usually constructed from four key units: power unit, processing unit, communicational unit, as well as a sensing unit. The sensing unit of a node controls particular physical attribute for instance temperature or the level of moisture of a place for which it is used. The processing element is in charge of processing the obtained data. The communication element of a node is used for transmitting/gathering of the obtained data to/from other sensors. The processing and communication consumes most of the energy of a sensor to function, and the power section, is in charge for supplying of energy into other components. Regardless of the diversity in the goals and objectives of sensor applications, the most important job of wireless sensor nodes would be to sense and accumulate data from a potential domain, processing the data, and send them back to specific sites. Accomplishment of this mission efficiently needs the building of energy-
efficient routing protocols to arrange paths in between sensor nodes and the data sink. The path choice should be in such a way to prolong network’s life as much as possible. The properties of the environment inside which sensor nodes generally function, in addition to intense resource and power limitation, make the design of routing protocols very challenging.

1.2 RESEARCH OBJECTIVES

The main aim of this thesis is to improve the routing algorithm for WSNs in terms of energy efficiency and balancing the power consumption in the network. The specific research objectives associated with achieving the research aim are as follows:

1. Design a protocol that can equally and efficiently distribute the energy consumption across all nodes and achieve an extended network lifetime.
2. Improving the cluster heads election in Hierarchical routing protocols using the concept of fuzzy logic.
3. Finding the optimal location for the sink node/ bases station in WSNs in order to improve the WSNs lifetime and throughput.
4. Finally, we want to be able to assess the performance of our design strategies with a focus on energy and throughput of the WSNs.

1.3 THESIS CONTRIBUTIONS

The contributions of this research are listed as follows:

1. Various routing protocols in different categories have been compared against one another based on various factors such as energy efficiency, their primary objectives, etc.
2. The performance analysis of well-known clustering routing algorithms based on fuzzy logic has been done based on energy variance and lifetime of the WSN.
3. A novel approach called Two-Step Fuzzy Logic System to achieve energy efficiency and prolonging the lifetime of WSNs have been proposed. This approach aims at finding the optimal cluster heads based on different factors.
such as remaining energy of sensor nodes, their distance from base station, density of sensor nodes etc.

4. A new routing Algorithm BOKHARI-SEPFL, has been proposed. The routing protocol is based on SEP protocol and uses the concept of fuzzy logic in order to improve the lifetime and throughput of the WSNs.

5. In order to improve the WSNs lifetime and throughput a novel algorithm (Bokhari-GM Algorithm) has been proposed in an effort to find the optimal location of the sink node in WSNs. To spot the optimal place our algorithm finds the geometric median of all the locations associated with the sensor nodes.

1.4 STRUCTURE OF THE THESIS

The rest of the thesis is organized as follows:

Chapter 2 gives the overview of WSNs, their differences from Ad-Hoc networks, technical challenges, and design metrics, type of wireless sensors, applications of WSNs, sensor node hardware platforms and operating systems.

Chapter 3 includes the background and related work. It contains the overview of routing challenges and design issues, comparison of various routing protocols in different categories, clustering techniques, concept fuzzy logic and related work.

Chapter 4 contains the performance analysis of some well-known clustering routing algorithms based on fuzzy logic.

In Chapter 5, a novel approach called Two-Step Fuzzy Logic System to achieve energy efficiency and prolonging the lifetime of WSNs have been given in details and discussed.

Chapter 6 presents the proposed routing protocol BOKHARI-SEPFL (Stable election protocol based on fuzzy logic). In this chapter the routing protocol has been discussed in details and compared based on energy efficiency and overall throughput against some other existing routing protocols.

In Chapter 7 contains a novel algorithm Called BOKHARI-GM to find the optimal location of the sink node in order to extend the lifetime of the WSNs.

Finally, Chapter 8 includes the conclusion and summarizes the whole thesis, and proposes some areas for further study.
CHAPTER TWO
WIRELESS SENSOR NETWORKS: AN OVERVIEW
2.1 INTRODUCTION

Technological developments in the field of microelectronic devices, memory, processors and radio technologies, have been leading into rapid production and development of tiny, low price, battery operated and multi-functional sensor nodes [1], [2], [3]. Wireless sensor networks are made of hundreds or thousands of these tiny sensor nodes (SNs). Wireless Sensor network (WSN) is considered as the most potential approach for keeping track of a physical location wherein monitoring and checking is critical e.g. security surveillance, military applications, health applications, traffic monitoring, home applications and environmental applications. A sensor node is usually constructed from four major units: power unit, processing unit, communicational unit, and a sensing unit [4]. The figure 2.1 [5] demonstrates the element in a typical sensing node. The sensing unit of a node controls particular physical attribute for instance temperature or senses soil moisture of a place for which it is used. The processing element is in charge of processing the obtained data. The wireless communication element in a sensor node is used for transmitting/gathering of the obtained data to/from other sensors. The processing and communication consumes most of the energy of a sensor to function, and the power component, which is of restricted amount, is responsible for supplying of energy into the three other components [4].
Regardless of the diversity in the goals and objectives of sensor applications, the most important job of wireless sensor nodes would be to sense and accumulate data from a potential domain, processing the data, and send them back to specific sites.

### 2.2 COMPARISON WITH Ad Hoc NETWORKS

Although ad hoc and wireless sensor networks are among the same class of wireless networks they differ from one another. Main differences are given as follows:

- **WSNs** are made of enormous number of tiny sensor nodes having limitations, such as restricted energy, short range of communication, low bandwidth, and limited storage and processing power.
- **Sensor Nodes** do not have global identification (GI) due to massive amount of overhead and huge number of sensors.
- **Nodes in WSNs** have limited power, and due to wide distribution of sensor nodes recharging of their batteries is not feasible.
- **SNs** are tightly positioned, therefore, the data obtained from neighbouring sensor nodes are very likely to be redundant.
- Comparing with ad-hoc networks **WSNs** are more vulnerable and more likely to fail due to insufficient power, radio interference or physical damages.

**Figure. 2.1.** Architecture of a typical Sensing node
SNs mostly utilize a broadcast communication paradigm or asymmetric many-to-one data flows, but ad-hoc networks are mostly based on point-to-point communications.

WSNs are meant to be deployed in any sort of environments, even extreme ones, such as battlefields, volcanoes, etc.

Because of failure of sensor nodes and changes in environment, WSNs are designed to adapt to dynamic topology.

2.3 TECHNICAL CHALLENGES OF WSNS

Some of the main technical challenges to be faced in order to design Wireless Sensor Networks are as follows:

- **Hardware:** An effective Hardware platforms for WSNs has to provide balance between the required functionality and satisfying system limitations such as restricted energy, short range of communication, limited storage and processing power etc. Sensor node’s design necessitates a grouping of micro-sensor equipment, low power computation and signal processing, and economically efficient wireless networking capability.

- **Wireless networking:** One of the Main issues and challenges to design routing protocols for wireless sensor networks is to deliver an energy efficient and robust communication mechanism that meets the application requirements. Wireless networking involves the issues related to the design of physical layer routing and mobility management.

- **Applications:** Applications of wireless sensor networks need different requirements such as effectual extraction, transport, manipulation and representation of information which are derived from SNs. Detection and data collection, data fusion and signal processing are among various components of different system.

2.4 DESIGN METRICS

There are several design metrics that in fact affect the overall performance of a sensor network. Some of the main metrics include energy efficiency, scalability, accuracy, fault tolerance and latency which are explained briefly as follows:
Efficient usage of energy: One of the main issues in WSNs is the Efficient usage of energy due to restricted power of SNs. Typically SNs are battery operated and in many scenarios it is not feasible or impracticable to replace or recharge the batteries due to wide distribution and huge number of SNs. Therefore energy management in order to extend the lifetime of the network has to be carefully considered [6].

Latency: Delay guaranteed service is required by several sensor applications such as multimedia networks etc. these type of applications need to deliver data to users in a certain delay time [7].

Robustness and fault-tolerance: WSNs have high rate of failure as they are deployed mainly in harsh areas. The designed protocols has to provide wide variety of robustness different type of failures such as sensor nodes failure, links failure etc. in order to allow uninterrupted operation of the network. The possible approaches could be via collaborative processing redundancy as well as communication [8].

Scalability: There could be WSNs built of millions SNs in a large areas. Scalable routing algorithms are needed in order to guarantee the efficient operation of the WSNs in an extensive range of sensor network. Network performance should not be affected considerably by a growth in the number of SNs or as the node density increases. Hence, WSNs has to use distributed protocols and hierarchical architecture along with localized communication in order to be scalable.

2.5 TYPES OF WIRELESS SENSORS

There exist a large choice of wireless sensors based on the type of sensing. For example there are sensors to sense [9][10]:

- Humidity
- Temperature
- Pressure
- Vehicular movement
- Soil makeup
- The absence or presence of objects
- Stress levels on the machinery of a factory
And in addition, there are plenty of functions regarding the wireless sensor nodes, for example the following [10]:

- Continues sensing
- Local control
- Object tracking
- Event detection
- Location sensing

### 2.6 APPLICATIONS

WSNs have major advantages compared to wired sensor networks [11]. Reducing deployment delay, low cost and more importantly the capability of being applied in any sort of environment such as deep oceans, active volcanos and battlegrounds are some of the main advantages. Although the military applications were the initial motivation for developing the WSNs but various other type of applications have also been inspired by them such as health care, environment monitoring, underground mining, precision agriculture, pipeline monitoring etc. Each of these applications have its own effect on the specifications and requirements of the hardware and software needed for the WSNs. Hence the understanding of various application possibilities is vital to the design of wireless sensor networks. An overview of the main applications are as follows:

#### 2.6.1 Environmental Monitoring

This is one of the earliest applications that is used to monitor various environmental attributes or conditions. Nowadays many important applications of WSNs are used for environmental monitoring as concerns about global warming, and other climate changes are arising. These applications such as pollution detection, habitant monitoring, planetary exploration, flood and forest fire detection, monitoring of earthquakes and active volcanos, are of significant importance for protecting our living environment. Variety of systems to detect natural disasters, monitoring energy resources and hazard response could be developed by employing WSNs. Environmental Monitoring can be classified into five groups as sown in figure 2.2 [2].
Figure 2.2 Classification of environmental monitoring

- **Habitat monitoring**
  It can be described as surveys intended for discovering and describing changes within the environment, and also to figure out the impacts of any act regarding wildlife conservation. Solutions, provided by WSNs for habitat monitoring offer considerable benefits over conventional approaches such as longstanding data collection at scales which would be extremely hard or even impossible to acquire otherwise [12]. In [13] authors designed and developed a WSN to monitor the seabirds on Skomer Island, a UK National Nature Reserve.

- **Energy Monitoring**
  Consumption of energy resources is critical to the global economy, and usually it is feasible to reduce expenditures by utilizing modern technologies as well as innovative management approaches. Being easy and efficient is one of the biggest advantages of employing WSNs to reduce and manage the energy waste. For instance sensor nodes can be used to measure the temperature or presence of humans in room and engaging the needed actions such as reducing the heat or turning off lights [14].

- **Pollution Monitoring**
  Undoubtedly a significant concern of the current century is the growth in pollution and its destructive consequences. Variety of pollutions such as Air pollution, radiological contamination, noise pollution and water pollution, just to mention a few of them, has to be monitored and dealt with in an effort to reduce its negative effects. Using WSNs Authors in [15] proposed an innovative
air pollution system named WAPMS which uses an Air Quality Index in order to categorize the level of air pollution.

- **Geological Monitoring**
  It involves repeated measurements to discover short and long term changes of different physical geological magnitudes. A common feature among geological catastrophes such as volcanic eruptions, earthquakes and tsunamis is the point that they are associated with an underground event [16]. A geological disaster could potentially cause loss of life and often causes several economic damage. WSN are easier to deploy for monitoring and keeping track of underground changes compared with the other methods where sensors need to be buried and linked via wire. In [17] an innovative quality-driven approach using WSN has been proposed in order to achieve real-time detection of volcanic earthquake.

- **Meteorological Monitoring**
  It is about controlling, supervising, and studying atmospheric and physical magnitudes. Information such as speed of wind and its direction, air humidity and temperature, rainfall and barometric pressure provided by conventional weather station could be very helpful for forecasting the weather and to predict natural phenomena [18] such as floods, droughts and hurricanes which could endanger human lives and significant economy losses. Deployment of WSNs for meteorological monitoring is very advantageous as large amounts of data could be easily acquired and stored in an effort for improving the reliability of predictions. In [19] a novel method based on WSN has been proposed in order to monitor the air temperature in a low cost. The method decreases the error of sensed air temperature by considering the solar radiation into account.

### 2.6.2 Health Care
Technology of WSNs have the ability to impact various health-care applications. These potential applications could include patient monitoring [20], people rescue [21], disability assistance [22], and bio-surveillance [23]. It can be deployed to monitor elderly people at their homes which would result in reducing the expenditure of health care as well as relieving the shortage of personnel. Researches of Harvard University designed a wearable system [24] in order to monitor the patients having Parkinson’s disease. The system was made of 8 low power nodes attached to different part of patient’s body and keep track of motion performance. A 2 GB memory was used to
store the obtained data. The experiment reveals that the system achieve long life, about 18 hours a day and high quality of data.

2.6.3 Military Application
WSNs have grown to be a major component of military command, control, communication and intelligence systems. Deployment of WSNs could easily monitor and track the presence and movement of army vehicles and forces in battlefields. It could be used for remote sensing of chemical, biological and nuclear weapons and reconnaissance and detection of potential terrorist attacks [25]. Employing sensor near sensitive areas such as nuclear plants, military communication centers and strategic construction could be done for protection purposes. Authors in [26] proposed a novel architecture of a tiered sensor network which uses advanced technologies of WSN for military operations. As the result declares the architecture results in an agile surveillance system.

2.6.4 Pipeline Monitoring
Pipeline monitoring is another interesting application for WSNs. sensors can be deployed to monitor water, gas and oil pipelines for leakages caused by landslides, earthquakes or even deliberate damages. Due to long length, value and high risks of pipelines, their management poses a tough challenge. To spot leakages, it is essential to know about properties of the substance which is transported by the pipelines. For instance, fluid results in generating a hot-spot at the place of the leak on the pipeline, but gas produce a cold-spot due to its pressure. Similarly, fluid moves at a higher speed in pipelines made of metal compared with polyvinyl chloride. In [27], a pipeline monitoring solution for oil and gas has been investigated and proposed based on WSNs with main attention on leakage sensing. The system is capable of recording and reporting health related statistics of the pipeline over a huge geographical region.

2.6.5 Structural Health Monitoring (SHM)
Another significant field for sensor network application is Structural Health Monitoring (SHM). The procedure of carrying out a damage discovery approach for engineering constructions is known as SHM. Some of the main objectives of SHM system are discovering damages, spotting the location of damages, measuring the scope of damages as well as computing the lasting lifetime of the structure. In this context damage is considered as changes to geometric features or material of a structural system
in such a way that disturbs its performance. It is critical to monitor the health of civil infrastructure such as buildings and bridge constructions in an attempt to reduce threat to human lives by warning about the possible collapses and dangerous structures. WSNs can be used to perform real-time monitoring to minimize the long-term operational cost and assure the safety and security of the old structures. In [28] authors have deployed a WSN for Structural Health Monitoring. The approach is called FTSHM which also guarantees a specified degree of fault tolerance for the WSN.

2.6.6 Industrial application

WSNs offer great value to organization to gain real-time access to information regarding their plants’ environment, tools and processes to optimize and improve the products and services. Some of the main application scenarios in this filed are Factory automation, Real-time monitoring of equipment’s health and Process Control. Industrial Wireless Sensor Networks (IWSNs) integrates WSNs into industrial systems offering several benefits such as intelligent controlling, cost efficient, quick development and wireless communication into existing industrial application. In [29] the design, implementation and testing of a remote energy monitoring system using WSN has been presented. The system supports energy efficient sustainable manufacturing in an industrial workshop. The results declares that the proposed system discovers energy relationships among different manufacturing processes which could be utilized in order to select efficient machining scheme and for energy saving discovery.

2.6.7 Smart Buildings

A smart building basically is referred to the buildings that achieves energy conservation, monitoring and management by taking advantage of developed technologies and materials. A vast amount of energy is wasted by inefficient usage of HVACR (Heating Ventilating, Air Conditioning and Refrigerating). WSN technology could be integrated into the buildings to achieve real-time monitoring of humidity, temperature and airflow and other physical parameters such as presence of inhabitants in an effort to reach energy efficiency and enhancing the convenience of occupants [30].
2.7 HARDWARE PLATFORMS

An effective Hardware platforms for WSNs need to provide an acceptable balance between the required functionality and satisfying system limitations such as restricted energy, short range of communication, limited storage and processing power etc. Hardware platforms for WSNs are mainly based on well-known microprocessor groups, like 8051 MCU industrialised by Intel, the MSP430 from Texas Instruments and the ATmega designed and developed by Atmel company. Most of the current researchers on WSNs are based on sensor nodes called MICA mote [31]. These sensor nodes which are developed by UC Berkeley are very energy efficient, as a two AA batteries make a MICA mote to run for about two years. Mica2 [32] is one of the popular and widely used motes that includes an ATMega128L [33] microcontroller and a Chipcon CC1000 [34] radio interface. Mica2 and couple of other well-known and mature commercially available platforms for WSNs are discussed as follows:

2.7.1 mica2

The mica2 [32] from Berkeley motes was one of the earliest platforms that became popular after it was commercialized by Crossbow Technology. The platform uses a processing unit, an 8-bit RISC architecture from ATmega family having a 128 KB flash memory and 4 KB RAM. The low amount of RAM could be a preventive factor in solutions demanding huge amounts of local data processing or message buffering. A Chipcon CC1000, a FSK transceiver chip, supporting low data rate is used as radio interface in the mica2. The image and features of mic2 are given in Figure 2.3 and Table 2.1 [35] respectively.

![Mica2 and its components](image)

**Figure. 2.3.** Mica2 and its components
Table 2.1. Summary of the mica2 platform features.

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Processor</strong></td>
<td>Atmel ATmega128L</td>
</tr>
<tr>
<td></td>
<td>Clock frequency 7.4 MHz</td>
</tr>
<tr>
<td></td>
<td>Flash 128KB</td>
</tr>
<tr>
<td></td>
<td>Ram 4KB</td>
</tr>
<tr>
<td></td>
<td>Current consumption 8mA</td>
</tr>
<tr>
<td>Minimum operating</td>
<td>Voltage 2.7 V</td>
</tr>
<tr>
<td><strong>Transceiver</strong></td>
<td>Chipcon CC1000</td>
</tr>
<tr>
<td></td>
<td>Modulation FSK</td>
</tr>
<tr>
<td></td>
<td>Frequency band 315/433 MHz; 868/916 MHz ISM</td>
</tr>
<tr>
<td>Data rate</td>
<td>38.4 kbps</td>
</tr>
<tr>
<td>Current consumption</td>
<td>27 mA (TX) 10 mA (RX)</td>
</tr>
<tr>
<td>Minimum operating</td>
<td>Voltage 2.1 V</td>
</tr>
<tr>
<td><strong>External storage</strong></td>
<td>Atmel AT45DB041B</td>
</tr>
<tr>
<td></td>
<td>Capacity 512 KB</td>
</tr>
<tr>
<td>Energy consumption</td>
<td>10 mA (R); 35 mA (W/E)</td>
</tr>
<tr>
<td>Minimum operating</td>
<td>Voltage 2.7 V</td>
</tr>
<tr>
<td><strong>Energy source</strong></td>
<td>2x AA batteries</td>
</tr>
</tbody>
</table>

2.7.2 MicaZ

MicaZ [36] includes the same processing unit as mic2, an ATmega128L. Their design is almost similar, both having the expansion connector, external power connector and a couple of AA batteries as the energy source. The only main difference is that MicaZ uses a Chipcon CC2420 [37] radio interface which improves its performance and functionality over mic2. The radio interface ChipCon CC2420 enables the MicaZ to send up to 250 kbps data at a frequency of 2.4 GHz. Additionally, the IDLE a new state
for radio transceiver is presented. It is faster to transit from IDLE to other states such as transmit or receive comparing with traditional SLEEP. The image and features of MicaZ are given in Figure 2.4 and Table 2.2 [35] respectively.

Figure 2.4. Micaz Sensor Node

Table 2.2 Summary of the micaz platform features.

<table>
<thead>
<tr>
<th>Micaz</th>
<th>UC Berkeley and Crossbow Technologies</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Processor</strong></td>
<td>Atmel ATmega128L</td>
</tr>
<tr>
<td>Clock frequency</td>
<td>7.4 MHz</td>
</tr>
<tr>
<td>Flash</td>
<td>128KB</td>
</tr>
<tr>
<td>Ram</td>
<td>4KB</td>
</tr>
<tr>
<td>Current consumption</td>
<td>8mA</td>
</tr>
<tr>
<td>Minimum operating voltage</td>
<td>2.7 V</td>
</tr>
<tr>
<td><strong>Transceiver</strong></td>
<td>Chipcon CC2420</td>
</tr>
<tr>
<td>Modulation</td>
<td>O-QPSK</td>
</tr>
<tr>
<td>Frequency band</td>
<td>2.4 GHz ISM</td>
</tr>
<tr>
<td>Data rate</td>
<td>250 kbps</td>
</tr>
<tr>
<td>Energy consumption</td>
<td>17.4 mA (TX); 19.7 mA (RX)</td>
</tr>
<tr>
<td>Minimum operating voltage</td>
<td>2.1 V</td>
</tr>
<tr>
<td><strong>External storage</strong></td>
<td>Atmel AT45DB041B</td>
</tr>
<tr>
<td>Capacity</td>
<td>512 KB</td>
</tr>
<tr>
<td>Current consumption</td>
<td>10 mA (R); 35 mA (W/E)</td>
</tr>
<tr>
<td>Minimum operating voltage</td>
<td>2.7 V</td>
</tr>
<tr>
<td><strong>Energy source</strong></td>
<td>2x AA batteries</td>
</tr>
</tbody>
</table>
2.7.3 TelosB

Just like the MicaZ, the TelosB [38] utilizes the same radio interface, a Chipcon CC2420. It has an integrated on board antenna and can send up to 250 kbps data at a frequency of 2.4 GHz. TelosB makes use of a MSP430F1611 [39] as the processing unit with 10 KB of RAM from Texas Instruments. Programming, data collection enabling power supply are all done through a USB (Universal Serial Bus) connector which makes the interfacing with other external devices easier. It includes on board sensors of Light, Temperature and Humidity. The energy source of the device could be either the USB connector or 2x AA batteries. The following figure 2.5 [40] shows the components of a sensor node TelosB and table 2.3 presents the image and features of the device.

![Figure 2.5. Components of a sensor node TelosB](image)

Table 2.3. Summary of the TelosB platform features.

<table>
<thead>
<tr>
<th>TelosB</th>
</tr>
</thead>
<tbody>
<tr>
<td>UC Berkeley and Crossbow Technologies, Moteiv Corporation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Processor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Texas Instruments MSP430F1611</td>
</tr>
<tr>
<td>Clock frequency</td>
</tr>
<tr>
<td>Flash</td>
</tr>
<tr>
<td>Ram</td>
</tr>
<tr>
<td>Current consumption</td>
</tr>
<tr>
<td>Minimum operating voltage</td>
</tr>
</tbody>
</table>
2.8 OPERATING SYSTEMS

Operating systems (OS) designed specifically for WSNs are normally less complex comparing with general-purpose ones. As WSNs are having some sever resource limitations such as energy, storage and processing power etc, the main concern of the OS is to manage the resources in an efficient manner while providing system developers with a programming interface. Storage management, energy management, processor time management, task scheduling, multitasking and multi-threading are some of the services that needs to be offered by the OS. Number of OS have been designed and developed specially for WSNs. Some of the well-known ones are discussed as follows:

2.8.1 TinyOS

TinyOS [41,42] basically is an open source, flexible, even-driven, component-based operating system (OS), designed especially for low-power sensor networks at UC Berkeley. The core OS is 400 bytes and many application fit within only 16KB of memory. All the components of TinyOS are written in the programming language nesC [43] a dialect of C. Each components is an independent entity having 3 computational abstractions: tasks, events and commands. A command (function invocation) is a request for running a service, and its completion is signalled by an event. The latest version is TinyOS 2.1.2 and was released on 2012 and multithreading, called TOS threads are supported in this release. The concept of Active Message [44] is used as the
communication architecture of TinyOS. Several multi hop routing protocols such as IPv6 and 6lowpan [45] are included. An implementation of some MAC protocols like a single hop TDMA protocol, B-MAC, TDMA/CSMA hybrid protocol, and IEEE 802.15.4 complaint MAC are also provided by the OS. Moreover variety of hardware platforms including mica2, MicaZ, TelosB etc. are also supported by the TinyOS. The figure 2.6 [42] depicts the architecture of TinyOS.

![Figure 2.6. Architecture of TinyOS](image)

### 2.8.2 Contiki

Contiki [46] is another lightweight open source operating system implemented in C for resource-constrained devices. Communication with hardware is done directly through a set of C functions. Initially was designed and developed by Dunke Lsetal at Swedish institute of computer science. Its main features include protothreads programming (using the advantages of event-driven and multi-threaded programming) and dynamic loading and unloading of code at run time. The latest version is 3.0 and was released on August 2015. All of the facilities delivered by the Contiki such as communication, data handling of sensors etc. are offered in the form of services. Each service has its interface and implementation and applications need to know only the service interface in order to use that particular service. Contiki delivers Internet Protocol (IP) communication, for both IPv4 as well as IPv6. Contiki covers two main communication stacks: uIP and
Rime. Communication over the internet is done through uIP TCP/IP stack and many MAC layers such as SMAC, CXMAC, X-MAC, ContikiMAC, Null MAC are supported via Rime stack. Figure 2.7 [42] illustrates the architecture of the Contiki.

![Contiki Operating System Architecture](image)

**Figure. 2.7.** Architecture of Contaki Operating System

### 2.8.3 MANTIS

MANTIS (MultimodAI system for NeTworks of In-situ wireless Sensors) [47] provides a lightweight, multithreaded and energy efficient operating system. One of the main feature of this operating system is its portability across multiple platforms, which allows one to test the MANTIS applications on a personal digital assistant (PDA) or a PC. It supports various platform including, Micaz, Mica2 and Telos motes. The MantisOS driver architecture is similar to POSIX (Portable Operating System Interface) model [48], and allows a short range of system calls with many different parameters. For instance, it uses only four system calls to interact completely between the driver layer and application code. These system calls are as follows: dev_read () (for reading the data) and dev_write () (for writing the data) dev_ioctl () (to pass device configuration) and dev_mode () (for controlling the device power state). Energy-efficiency is achieved by taking advantage of power-efficient scheduler, that effectively puts the processor...
into a sleep mode after all active threads in the system called the sleep () function of MantiOS. This function is similar to UNIX sleep () function. The code of the kernel, network stack and scheduler, occupies about 14 KB memory and less than 500 bytes of RAM which provides enough space for executing multiple application threads. The figure 2.8 shows the architecture of the operating system [47].

**Figure 2.8.** Architecture of MANTIS OS
CHAPTER THREE

BACKGROUND AND RELATED WORK
BACKGROUND AND RELATED WORK

3.1 INTRODUCTION
Due to number of unique characteristics of Wireless Sensor Networks (WSNs) the design of routing protocols for these type of networks is a very challenging task. Firstly it’s not practicable to design an overall addressing system for WSNs as in typical communication networks. Secondly because of the significant redundancy of the generated data in WSNs, to preserve the energy and improve the bandwidth, such redundancy has to be dealt with. Thirdly sensor nodes are extremely constrained, regarding their energy, storage, and processing capabilities. During the last decade many routing algorithms have been proposed and designed specifically for WSNs and most of them aim to prolong the life of the network [49-51].

3.2 ROUTING CHALLENGES AND DESIGN ISSUES IN WSN
A routing algorithm might be thought as a technique by which a node comes to a decision about its neighbouring route when considering to send a packet to a targeted destination. The distinguished characteristics of WSN makes the routing a very challenging task. As there is a huge number of sensor nodes it won’t be feasible to use a global addressing scheme [52]. In WSN data flows from multiple sources to a base station. There are several other issues which have to be considered while designing of routing protocols for WSNs. Some of the main challenging factors [53] are discussed below:

- **Node Deployment**
  In wireless sensor networks node deployment is application-specific and could be manual or randomized. Where the deployment is manual, the SNs are manually positioned and data will be routed through fixed paths. Though in case of random
deployment SNs will be scattered in a random manner forming an ad hoc routing structure.

- **Energy Consumption**
  Energy consumption could be addressed as one of the main challenges. SNs are typically battery powered and won’t be feasible to recharge them once employed. Failure of SNs could significantly affect the performance and topology of the WSNs which might lead to reconfiguration of network and re-routing of packets.

- **Fault Tolerance**
  Due to variety of reason such as lack of energy, technical issues, environmental interference or physical damages SNs may fail or stop functioning. The overall functionality of the WSNs should not be affected by this failure which means there should be alternative routes to send data to the destination.

- **Scalability**
  A WSN could be constructed of hundreds or thousands of SNs. In many cases there might be situations where expanding of the network is required. These expansion is usually done by adding some more SNs. Every routing scheme must be capable of handling these huge number of SNs and managing the new SNs added to the network. Normally nodes within the sensor network are in sleep state and each time an event is sensed, they are switched to active mode.

- **Coverage**
  Each and every SNs in the network is responsible of covering a specific physical area of the environment. The coverage ability of the SNs is limited in accuracy as well as range. Therefore the coverage of the network should be taken care of as it is one of the important design issues in WSNs.

- **Data Aggregation**
  As SNs are most likely to generate a huge amount of redundant data, aggregation of similar packets from several SNs could be done in order to reduce the number of transmissions and improving energy consumption of the network. Data aggregation is the mixture of data from several sources based on a specific aggregation function.

- **Quality of Service**
Many applications require data to be delivered in a specific period of time else data will be of no use. So restricted latency for delivery of is one more condition for time-constrained applications. Though in many other applications energy conservation and prolonging the network lifetime is considered more important than the time of delivery. In some cases WSN may need to reduce the quality of result in an effort to prolong the network lifetime. As a result, energy-aware routing protocols will need to obtain this requirement.

3.3 ROUTING CLASSES IN WSNs

The routing approaches especially for WSNs can be categorized as Data-centric, Location-based and hierarchical routing protocols as shown in Figure 3.1. Data centric routing protocols are typically based on query, hence capable of reducing the redundancy of the data significantly. In location-based approach a message is forwarded from a source to the destination via the most efficient path discovered by the location of the neighboring sensor nodes of the sender. Hierarchical routing protocols divides the networks into number of clusters and each cluster possesses a cluster head which is responsible for gathering data from other nodes (within the cluster), performing data aggregation and fusion then sending them to the sink. Next some of main routing protocols in different classes, will be discussed and compared against each other based on their designs and primary goals.

![Figure 3.1. Classification of Routing protocols in WSNs](image)

3.3.1 Data Centric Routing Protocols
In WSNs, each single node is required to send out data to the sink which leads to a considerable redundancy, resulting in large wastage of energy waste. Thus, routing approaches have been introduced that are capable of selecting a range of nodes on query based, known as Data Centric Routing. Queries are sent by the base station (BS) to a particular region for the desired information. Considering that data are demanded via queries only a certain data from an interested region needs to be transmitted to and as a result this will reduce the redundancy of the data as well as the number of transmission which will improve the energy consumption and lifetime of the network significantly. Few of the main routing protocols in this class are briefly given below.

### 3.3.1.1 SPIN (Security Protocols for Sensor Networks)

One of the earliest routing protocol based on data centric approach is SPIN [54]. In SPIN protocol data are named using meta-data or other high-level descriptors. With the help of an advertising mechanism the data descriptors are exchanged before the transmission process. As soon as a node possesses a new data to be shared, it will generate an ADV message and send it to its neighboring nodes, in return the neighboring nodes use a request message in order to obtain the desired data (Provided they haven’t possessed it already) through a REQ message. Finally, the source from where the ADV message has been generated and sent transmits the real data to finish up the process. The process is shown in Figure 3.2 which is redrawn from [54]. The table.3.1 shows the most important routing protocols in this category along with their key characteristics and objectives.

![Figure 3.2. Working procedure of SPIN protocol.](image)

### 3.3.1.2 Direct Diffusion
Direct diffusion [55] is basically a data centric protocol where the aggregation of data is performed at every node within the network. The sensed data by nodes will be advertised once a request is made by the BS. In Direct Diffusion a naming scheme is used in order to reduce the number of transmission. All the data produced by SNs are named by attribute-value pairs. To the rate of transmission for the shortest path a reinforcement technique is used as depicted in figure 3.3 [56]. The process of data collection is initialized by the BS. This process could be described in three steps as given below [57, 58].

![Figure 3.3. Working procedure of Direct Diffusion protocol.](image)

**Step 1:**
An interest packet is broadcasted by BS to all its neighbor nodes, then these neighbors broadcast the packet to their neighbors till the packet reaches the source node containing the data. The interest packet contains gradient value which includes attributes value as well as direction.

**Step 2:**
The data packet is sent to the BS by the source node which has the demanded data. The data packet is sent through multi-paths according to gradient.

**Step 3:**
BS reinforces the best paths as depicted in figure 3. Choosing the best path is depended on the gradient value as well as the application, for instance, in some applications the path consuming the lowest energy is desired and some others application require the shortest path.

In Directed Diffusion request packet is always generated by the BS and broadcast to SNs, while in SPIN protocol SNs advertise that they have data and let the interested SNs to demand it. In a Directed Diffusion based network all the SNs are application-aware, that allows diffusion accomplish energy saving by choosing empirically decent routes, and through caching and processing of data within WSN.

### 3.3.1.3 Rumor Routing

In Rumor routing [59] the BS sends a query straight to the SN that detects an event, not flooding the query to the whole network as in Directed Diffusion. An event is a concept taken from a group of sensor readings that is considered to be a local phenomenon happening within the network filed. A query is a call for information, sent by the BS, and as soon as the query gets to its destination the data can start to move back to the source of query. Once an event is detected by a SN, it is added to an event table then an agent is generated. The agent moves to each and every nodes and gives them the information regarding the event. SNs that receive the agent and also know the routes response to the query of BS. Figure 3.4 [60] below illustrates the working of the rumor routing algorithm.

![Figure 3.4](image)

**Figure 3.4.** Working procedure of the Rumor routing algorithm.

In figure 4 (a) a path of distance 2 has been initially recorded by the agent. The table of node A indicates a distance of 2 for E2 and a distance of 3 for E1. When the agents...
visits the node A first updates its own information and add the path to E2. As the node A is visited by the agent its table also is optimized and the path distance to E1 becomes 2 through node B. the updated status of the node table and agent are given in figure 5 (b). Once a query is generated at sink, it will be sent on random walk with the hope of finding a path leading to the event. After a period of time if the query does not find a path to the event the sink uses flooding to broadcast the query. For example, as in Figure 5 (c), a query is generated for event E1 by node P. the query by a random walk, reaches node A, and then query is directed to E1 through node B by using the table information on node A. In comparison with Directed Diffusion and Flooding, Rumor is able to manage node failure and consume less energy. When there is only few event applications Rumor operates effectively, however if exist a lot of events, the building of an event tables as well as agents would lead to extra overhead on the resources of the network which includes computation operations and memory [58].

3.3.1.4 ACQUIRE (Active Query forwarding in the WSNs)
ACQUIRE [61] considers query as an active query that will be routed within the WSN, searching for a solution. At every SN, the query is forwarded utilizing the information from other SNs within $d$ hops, which actually resolves the query partly. Within the SN where the query is totally resolved, a response is send back to the source of query. It is just like Directed Diffusion. Though, ACQUIRE can also work well with complex queries. Directed Diffusion makes use of flooding to broadcast the query, whereas ACQUIRE transmits an active query to every SNs. In the event this query is resolved, the response is sent straight to the querying node. In case the hops among the source node and the destination node is equivalent to the WSN diameter, the performance of ACQUIRE would be same as the Flooding performance [58]. Furthermore, ACQUIRE is more energy efficient comparing with Directed Diffusion. Generally, it reduces the energy consumption around 60 % as compared with Directed Diffusion.

3.3.1.5 Reliable Energy Aware Routing (REAR)
REAR [62] is basically a routing protocol that tries to achieve reliable packet delivery as well as energy efficiency. There are three type of node used by this protocol: Base Station, Intermediate Nodes (IN) and last one is Target Source (TS). The algorithm uses
a layer to provide the energy aware path by utilizing energy-reservation mechanism and uses another layer (transport layer) in network to deliver reliability. Different parts of REAR are explained as below:

**SPD (Service Path Discovery):** In SPD, BS sends the request for path discovery within the WSN. It makes use of flooding for the request. While travelling, the speed of broadcasting is combined with available energy in order to decide on the energy efficient path. When source gets this candidate path, a request for path reservation is generated.

**BPD (Backup Path Discovery):** BPD is also started by BS and has almost a similar process as SPD. The only difference is the nodes that have already been selected for service path won’t be included in backup path. The backup path will be used if the service path fails.

**Reliable Transmission:** Every transfer SN stores the data while waiting for acknowledgment from the receiver node.

**Reserved Energy Release:** In case the link fails, all the intermediate SN will receive an error message and releasing the energy that was reserved for that path.

The Table 3.1 compares the some of the well-known data centric routing protocols and also signifies their main objectives.

### Table 3.1. Comparison of the main data centric routing protocols and their main objectives

<table>
<thead>
<tr>
<th>Routing Protocol</th>
<th>NO. of possible BS</th>
<th>Data Aggregation</th>
<th>Adaptive To mobility</th>
<th>considering The Battery Lifetime</th>
<th>Key Objectives Of The Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPIN[54]</td>
<td>SINGLE</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>• Preserving energy to Extend network Lifetime</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Reducing number of messages</td>
</tr>
<tr>
<td>Direct Diffusion[55]</td>
<td>MULTIPLE</td>
<td>YES</td>
<td>LIMITED</td>
<td>NO</td>
<td>• Fault tolerance</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Improving on Data diffusion</td>
</tr>
</tbody>
</table>
### 3.3.2 Location Based Routing Protocols

Location based routing protocols utilizes the geographical location of sensor nodes in order to form the optimal route and send packets from a source to the desired destination. A packet is sent from source to destination by considering the geographical position of the neighbouring nodes of the forwarder. The information about sensor node’s location are obtained via Global Positioning System (GPS) which are of tiny in size and low consuming power devices embedded in the body of the sensor nodes. Some of the main routing protocols in this class are briefly given below.

#### 3.3.2.1 GAF (Geographic Adaptive Fidelity)
GAF [67] is basically a location based as well as an energy-aware routing protocol. GAP preserves the energy of the network by disabling the needless nodes of the network without having an effect on the routing fidelity. A virtual grid id made and every node utilizes its location to link itself with a place in it as it is shown in figure 4. Nodes connected with the exact same place on the grid are considered to have same cost for routing of packets. Hence in order to prolong the lifetime of the network some nodes linked to the same point on the virtual grid can be put into the sleeping state. In figure 3.5, node 1 is able to reach node 2, 3 or 4 and moreover nodes 2, 3, and 4 are capable of reaching the node 5. Thus nodes 2, 3 and 4 are considered to be equal and any two of them can go into the sleeping mode in order to save energy [68].

![Virtual grid in GAF](image)

**Figure 3.5. Virtual grid in GAF**

### 3.3.2.2 SPEED

SPEED [69] demands each individual node to preserve information regarding its neighboring nodes to locate the paths. Moreover, SPEED attempt to guarantee a specific speed to each and every packet within the network so applications could predict the delay of the messages through the act of dividing the BS distance by the packets’ speed. And in addition, SPEED can offer congestion escaping in the event that network is overcrowded. In SPEED protocol the routing module is known as SNFG which functions with other four units, as depicted in figure 3.6.
GEAR (Geographic and Energy Aware Routing)

GEAR [70], makes use of energy aware neighbour selection for routing a message towards the desired area. In GEAR, every individual node maintains two different costs, an expected cost along with a knowledge cost of getting to the target as shown in Figure 3.7. The expected cost would be calculated by the distance (source to destination) and the remaining energy. In another hand the knowledgeable cost would be a modification of the expected cost that represents routing about holes. A hole in the network happens whenever a node has no other closer neighbor than itself to the target region.

![Figure 3.6. Modules of SPEED routing protocol](image)

![Figure 3.7. GEAR network](image)

EAGR (Energy Aware Greedy Routing)

EAGR [71] is a location-based routing algorithm and its main objectives are prolonging the network lifetime, balancing the energy consumption of the SNs and to achieve a higher data delivery rate. The algorithm works based on location and energy level of SNs. In EAGR all the SNs are assumed to know their location and energy level as well as location and energy level of their neighbours. EAGR assumes that all the SNs in the network have the same energy and a threshold of energy level is defined. SNs having less energy than that threshold are considered to be dead. All the SNs need to have energy level more than their locations energy cost. The forwarding node will write the address of the destination in the header file and send it to the sub-destination. The sub-
destinations should be near to the destination and having enough energy (energy level should be greater than then threshold), then the sub-destinations will route the packet by taking the reliable and shortest path.

The Table 3.2 compares the some of the well-known location-based protocols based on few parameters such as number of possible BS that protocols can have, ability to do data aggregation, mobility and whether the protocol consider the battery lifetime in order to make decisions. Moreover the main objectives of the protocols have been given.

**Table 3.2.** Comparison of some location-based routing protocols and their main objectives

<table>
<thead>
<tr>
<th>Routing Protocol</th>
<th>NO. of possible BS</th>
<th>Data Aggregation</th>
<th>mobility</th>
<th>Considering Battery Lifetime</th>
<th>Key Objectives Of The Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAF [67]</td>
<td>MULTIPLE</td>
<td>NO</td>
<td>LIMITED</td>
<td>YES</td>
<td>• Extending network Lifetime</td>
</tr>
<tr>
<td>SPEED [69]</td>
<td>MULTIPLE</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
<td>• Extending network Lifetime</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Achieving Real Time</td>
</tr>
<tr>
<td>GEAR[70]</td>
<td>SINGLE</td>
<td>NO</td>
<td>LIMITED</td>
<td>NO</td>
<td>• Extending network Lifetime</td>
</tr>
<tr>
<td>EAGR [71]</td>
<td>MULTIPLE</td>
<td>NO</td>
<td>LIMITED</td>
<td>YES</td>
<td>• Enhancement of GAF protocol</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Extending network Lifetime</td>
</tr>
<tr>
<td>MMSPEED [72]</td>
<td>MULTIPLE</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
<td>• Extending network Lifetime</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Achieving Real Time</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Enhancing SPEED protocol</td>
</tr>
</tbody>
</table>

### 3.3.3 Hierarchical Routing Protocols

In this type of routing the whole network is divided into number of clusters and in each cluster one node will act as the cluster head (CH). The CH is in charge of receiving the
sensed data from other nodes within the cluster (cluster members) and performing data aggregation and/or data fusion, then sending the data to the base station. Different routing protocols have different techniques for selection of cluster heads. However the residual energy of a node and its distance from the base stations are the main factors that are considered by the recent and modern routing protocols while election of CHs. The main objective of this category of routing protocol is to balance the energy among the sensor nodes in order to extend the lifetime of the network.

3.3.3.1 LEACH (Low Energy Adaptive Clustering Hierarchy)

The earliest and most well-known hierarchal routing protocol is LEACH [73]. It reduces the energy consumption of the network by breaking down the network into number of clusters. One node from each cluster will be elected as the cluster head (CH). In LEACH all the SNs send out the collected data to cluster heads (CHs), and the CHs Combine and then pass it to the sink (Base Station). Figure 3.8 depicts the cluster formation in LEACH routing protocol. The CHs are changed and selected in a random fashion to distribute energy load evenly to each and every node as the normal nodes have minor ranges to pass data and CHs are in charge of large transmission of data towards the BS.

![Figure 3.8. Cluster formation in LEACH routing protocol.](image)

The procedure of LEACH is split up into rounds and each single round is divided into a pair of phases known as: setup phase and steady state phase. Steady-state phase tends
to be longer compared with the setup phase to decrease the overhead. A single round starts off with the set-up phase where clusters are created, leading to the steady-state phase in which a number of frames of data are forwarded from the nodes towards the CH and then to the base station as shown in Figure 3.9. In the set-up phase, each individual sensor node attempts in selecting itself as a CH according to probability model. For becoming a CH, all sensor nodes produce a number randomly between 0 and 1. In case the number is lower than the threshold $T(n)$, the sensor node becomes a CH for the current round, the threshold is given as follows:

$$T(n) = \begin{cases} \frac{p}{1-P*(r \mod \frac{p}{n})} & \text{if } n \in M \\ 0 & \text{otherwise} \end{cases} \quad (3.1)$$

Where, $n$ indicates set of given nodes, $r$ stands for the current round, $p$ represents the desired percentage of CHs and $M$ indicates the set of sensor nodes that haven’t been selected as CH in the last $1/p$ rounds.

A message will be broadcasted by nodes which are selected as cluster heads, thereafter other nodes which have not been chosen as cluster heads collect the information of all CHs. Based on the intensity of signal of each CH, nodes which have the lowest cost of communication can join the cluster. Figure 3.10 depicts procedure of CH selection in LEACH routing protocol.

![Figure 3.9. LEACH setup phase](image)

![Figure 3.10. Steady phase in LEACH](image)
3.3.3.2 PEGASIS (Power-Efficient Gathering in Sensor Information Systems)

PEGASIS [74] is basically a near optimal chain-based routing protocol which was introduced as an enhancement to LEACH. In PEGASIS SNs are required to only talk to their neighbours. Hence it reduces the bandwidth consumption and improves the network lifetime. In order to build the chain, PEGASIS utilizes a greedy approach, which starts with the furthest node from the BS, and attempts to discover a neighbouring node that has a smaller distance to the BS. By utilizing the greedy algorithm, every node gets connected to its closest neighbour and nodes which are already a part of the chain cannot be reconsidered. Every round a node obtains observations from its neighbouring node, combines them with its own observations and sends them to the other neighbours on the chain, until eventually all of the data is combined and aggregated at the node, known as chain leader. The node ‘n’ at a random place ‘p’ on the chain is elected as the chain leader which will be responsible for receiving data from other members on the chain and transmitting them to the BS. The Figure 3.11 shows a typical architecture of PEGASIS routing protocol.

![Figure 3.11. Typical architecture of PEGASIS routing protocol](image)

In PEGASIS routing protocol, it is assumed that nodes have global knowledge about the location of all the other sensor nodes in the network. There exist an extreme delay for distant sensor nodes on the chain.
3.3.3.3 TEEN (Threshold sensitive Energy Efficient Sensor Network Protocol):  
TEEN [75] was introduced as an enhancement to LEACH. It is also a cluster based routing protocol. TEEN is suitable for time critical sensing applications (e.g. fire alarms) and also applications that need to sense physical phenomenal (e.g. Pressure, temperature). The main aim of the protocol is to prolong the lifetime of the network by controlling the forwarded number of readings from the SNs to the CHs. TEEN achieves this by introducing two thresholds, Hard Threshold (HT) and Soft Threshold (ST). Within this protocol, every individual node continually senses the environment, however the transmitter of a node having a new reading will be turned on to transmit the sensed data only if sensed value (SV) fulfils a pair of conditions. First, if the current SV is greater than HT; and the second condition is if the SV is different from the previous SV.

3.3.3.4 APTEEN (Adaptive Periodic Threshold-sensitive Energy Efficient Sensor Network Protocol)  
The main disadvantage of the TEEN could be in circumstances where the thresholds are not reached and no data might be obtained from the network. To overcome this drawback, the same authors introduced an improved routing protocol called APTEEN [76]. APTEEN is a hybrid protocol that takes the advantages of two different communication policies; reactive as in LEACH and proactive as in TEEN. In APTEEN a node is forced to send its data if the value of Count Time (CT) is exceeded. The CT defines the duration between any two successive reports by a node. In APTEEN because of additional complexity and further data transmissions, lifetime of the network is less comparing with TEEN protocol.

3.3.3.5 Self-Organizing Protocol  
The Self-Organizing [77] is also a hierarchal routing protocol that is capable of self-configuring the network and arranging the nodes in the network according to the applications. A group of SNs sense and collect data of the surrounding environment, and another group of sensor which are fixed in their position act as routers. Routers are responsible for receiving the data from the sensing nodes and forward then to the sink nodes. Sink nodes have the most energy in the network. In order to achieve fault tolerance among the SNs the local Markov Loops (LML) algorithm is used [77]. The figure 3.12 shows the process of this protocol.
The protocol could be described in four phases [77] as given below.

I. **Discovery phase:** In this phase the related neighbours of nodes are discovered.

II. **Association phase:** Where a hierarchy is made based on the group and position of the sensors.

III. **Maintenance phase:** In this phase each nodes informs the neighbour nodes its level of energy and routing tables. Any update on routing table or level of energy happens throughout this phase.

IV. **Self-reorganization phase:** This is the last stage in this algorithm. In self-reorganization phase the failure of any node will be overcome.

The Table 3.2 compares some of the well-known hierarchical routing protocols and also gives their main objectives.

**Table 3.3. Comparison of the main hierarchical routing protocols and their main objectives**

<table>
<thead>
<tr>
<th>Routing Protocol</th>
<th>NO. of possible BS</th>
<th>Data Aggregation</th>
<th>Adaptive To mobility</th>
<th>Taking Into Account The Battery Lifetime</th>
<th>Key Objectives Of The Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEACH [73]</td>
<td>SINGLE</td>
<td>YES</td>
<td>FIXED BS</td>
<td>YES</td>
<td>• Extending network Life time</td>
</tr>
<tr>
<td>PEGASIS [74]</td>
<td>SINGLE</td>
<td>NO</td>
<td>FIXED BS</td>
<td>YES</td>
<td>• Extending network Life time</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Improving bandwidth of network</td>
</tr>
</tbody>
</table>
### 3.4 CLUSTERING TECHNIQUES

Clustering is known as a grouping of the same data items together. In other words, clustering is basically a grouping or organizing of things that share one or multiple characteristics. In case of WSNs due to limited energy supply, SNs might fail and the failure of even one node could possibly disturb the application or entire system. An effective way to reduce the energy consumption is using appropriate clustering algorithm, due to the fact that clustering algorithms tend to be more power efficient compared to direct routing algorithm. In this, sensor nodes are put together to create small clusters and a CH for any cluster is selected. Basically CH acts as a local coordinator for its cluster and performs the following:

- Arranging the schedule for intra-cluster transmission
- Collecting data obtained by cluster members
- Performing data aggregation after receiving data from cluster members
- Forwarding data to BS

Within this architecture, sensor node sends data to their particular CH and after aggregation the CH forward them to BS as shown in Figure 3.13.
Clustering offers the following advantages:

- It minimizes collision at the time of intra-cluster communication.
- It provides load and energy balancing by rotation of CHs.
- Decreases the required number of information to be updated (the node information such as joins and deaths will be updated only by the cluster members of that node).
- (Spatial reuse and scalability are offered reuse (for transmission, non-neighbour clusters could utilize the same frequency or code).

A variety of clustering approaches have already been proposed for WSNs in just the past few years. Each of these clustering strategies add flexibility in achieving many goals such as: Extending the lifetime of WSNs, efficient operation, and reducing the number of required nodes for communicating with the BS. Clustering algorithms in WSN are organized based on their approaches, applications and motivations. The authors of [80] arrange clustering into four groups: Heuristic Schemes, Weighted Schemes, Hierarchical Schemes, and Grid Schemes, as shown in Figure 3.14.

**Figure 3.13 Clustering architecture**
3.5 FUZZY LOGIC

The concept of Fuzzy Logic (FL) was introduced in mid-1960s by L.Zadeh [81, 82, 83]. It was specifically designed and developed for dealing with reasoning that is imprecise rather than fixed and exact. L.Zadeh noted: "The closer one looks at a real-world problem, the fuzzier becomes its solution" [84]. In FL, uncertainty and vagueness are represented mathematically to offer formalized tools for handling fuzziness to numerous problems. In other words FL allows approximate human reasoning abilities to be used in knowledge-based systems. The traditional solutions for representation of knowledge, lack the means to represent the purport of the fuzzy concepts hence do not provide a proper theoretical framework to handle common-sense knowledge. The design and development of FL was motivated and started due to an essential need for a conceptual framework that could address and deal with the issues of imprecision and vagueness. Fuzzy concepts are demonstrated by using degree of membership. Basically the classic Boolean logic deals with only two values True (YES) presented numerically as 1 and False (NO) which is presented numerically as 0, in contrast FL uses the degree of membership by taking the values between 0 and 1. One main benefit to fuzzy logic is, natural languages are used in order to formalize human reasoning. For instance,
Table 3.4 shows few rules that a driver needs to follow, in order not to lose his driving license [85]:

**Table 3.4. Rules that a driver needs to follow (fuzzy logic concept)**

<table>
<thead>
<tr>
<th>Condition 1</th>
<th>Condition 2</th>
<th>Condition 3</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>If the light is red</td>
<td>If my speed is high</td>
<td>And if the light is close</td>
<td>Then I brake hard.</td>
</tr>
<tr>
<td>If the light is red</td>
<td>If my speed is low</td>
<td>And if the light is far</td>
<td>Then I maintain my speed.</td>
</tr>
<tr>
<td>If the light is orange</td>
<td>If my speed is average</td>
<td>And if the light is far</td>
<td>Then I brake gently.</td>
</tr>
<tr>
<td>If the light is green</td>
<td>If my speed is low</td>
<td>And if the light is close</td>
<td>Then I accelerate.</td>
</tr>
</tbody>
</table>

Intuitively, it appears that the input variables as given in the above example are approximately appreciated by the brain, such as the degree of verification of a condition in fuzzy logic [85].

Most important characteristics of FL are as follows [86, 87, 88]:

- In FL exact reasoning is seen as a limiting case of approximate reasoning.
- In FL Everything is a matter of degree.
- In FL Knowledge is interpreted a collection of elastic or, equivalently, fuzzy constraint on a collection of variables.
- In FL Inference is seen as a process of propagation of elastic constraints.
- Any logical system can be fuzzified.
- Fuzzy systems are suitable for uncertain or approximate reasoning.
- FL lets decision making with approximate values under incomplete or uncertain information

3.5.1 **Advantages of Fuzzy Logic controller**
• No mathematical model is needed in the control algorithm. Fuzzy Logic has been developed as such a mathematical model.

• While starting to design the system there is no need for more knowledge about system which lead into Rapid prototyping for designing a system

Rapid prototyping for designing a system. Here, does not need more knowledge about the system while starting to design a system.

• In such cases where there is no explicit process model available, or in which the analytical model is too difficult to evaluate e.g. multiple input multiple output systems.

• It is not necessary to model the whole controller with Fuzzy Control, sometimes Fuzzy Control just interpolates between a series of locally linear models, or dynamically adapts the parameters of a 'linear controller', thereby rendering it non-linear, or alternatively just 'zoom in' onto a certain feature of an existing controller that needs to be improved [89].

• More cheaper than conventional controller systems because it is easy and less time to build.

• Increase robustness of the system.

• It is can derive a solution for a given case out of rules that have been defined for similar cases and fuzzy logic will effectively put this knowledge into solution, e.g. very hot, hot, warm, cold, very cold.

3.5.2 Crisp and Fuzzy Sets

Crisp and fuzzy sets are different from one another by the concept of membership function. Let’s consider A to be a crisp set defined within the universe of discourse X. Thereafter for just about any element x in X, whether x is a member of A or not. Crisp sets are often referred to as classical sets. Universe of discourse or just simply universe is the set of all possible elements that could be considered. In a fuzzy set, all the possible value to be assigned to a variable are represented by x-axis and the membership value of fuzzy set is represented by y-axis [89].

The \( f_A \) (membership function) of an element x for a crisp set A is defined as follows:
Thus, if \( x \) belongs to the set \( A \), for every single element \( x \) of universe of discourse \( X \), membership function \( f_A(x) \) is 1, and if \( x \) does not belong, then it is equal to 0. As an example let \( X \) be set of real numbers between 1 to 20 and let \( B = [11, 15] \) be the subset of \( X \) of real numbers between 11 and 18. This will result in the figure 3.15.

![Figure 3.15. example of Crisp set](image)

In the other hand we the elements of the fuzzy sets belong to a subordinate fuzzy set, having a particular degree of membership \([90]\). For every single element \( x \) of universe of discourse \( X \), if \( X \) belongs to set \( A \), membership function \( \mu_A(X) \)) is equal to the degree in which \( X \) belongs to set \( A \) as illustrated in Figure 3.16. However if \( x \) is not a member of set \( A \), then the membership function \( \mu_A(X) \) would be equal to zero. The most important difference between crisp sets and fuzzy set is the fact that the elements in fuzzy sets are capable of having partial membership with regards to a set. The membership function \( \mu_A(X) \) of an element \( X \) for a fuzzy set \( A \) is defined as follows:

\[
\mu_A(X) : X \rightarrow [0,1], \text{ where } \begin{cases} 
\mu_A(X) = 1 & \text{if } X \text{ is a full member of set } A \\
\mu_A(X) = 0 & \text{if } X \text{ is not a member of set } A \\
0 < \mu_A(X) < 1 & \text{if } X \text{ is partial member of set } A
\end{cases}
\]
Figure 3.16. A typical fuzzy set

As same as the previous example of crisp set, let $X$ be set of real numbers between 1 and 20. A fuzzy set of real numbers close to 13 could be provided as the result in the figure 3.17.

Figure 3.17. Example of fuzzy set

3.5.3 Properties of Fuzzy Sets

Classical set and fuzzy set share some alike properties. Crisp set is a unique case of fuzzy set wherein values of membership are a subset of the interval $[0,1]$. The rules given below in table 2 [91], which are common in classical set theory (crisp set theory), get applied to fuzzy set theory as well.

Table 3.5. Properties of Fuzzy Sets

<table>
<thead>
<tr>
<th></th>
<th>Commutativity</th>
<th>$\tilde{A} \cup \tilde{B} = \tilde{B} \cup \tilde{A}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\tilde{A} \cap \tilde{B} = \tilde{B} \cap \tilde{A}$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Associativity</th>
<th>$\tilde{A} \cup (\tilde{B} \cup \tilde{C}) = (\tilde{A} \cup \tilde{B}) \cup \tilde{C}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\tilde{A} \cap (\tilde{B} \cap \tilde{C}) = (\tilde{A} \cap \tilde{B}) \cap \tilde{C}$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Distributivity</th>
<th>$\tilde{A} \cup (\tilde{B} \cap \tilde{C}) = (\tilde{A} \cup \tilde{B}) \cap (\tilde{A} \cup \tilde{C})$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>
| 4. | Idempotency | $\bar{A} \cap \bar{A} = \bar{A}$  
$\bar{A} \cap \bar{A} = \bar{A}$ |
| 5. | Identity | $A \cup \emptyset = A$ and $A \cap X = A$  
$A \cap \emptyset = \emptyset$ and $A \cup X = X$ |
| 6. | Transitivity | if $A \subseteq \bar{B}$ and $\bar{B} \subseteq \bar{C}$ then $\bar{A} \subseteq \bar{C}$ |

### 3.5.4 Operations on Fuzzy Sets

As a way to effortlessly manipulate the fuzzy sets, the operators of the classical set theory have been redefined to fit the particular membership function of fuzzy logic for values between 0 and 1. Union (OR), intersection (AND) and complement (NOT) are the main operations on fuzzy sets.

- Intersection: Intersection (AND logic operator) contains the shared elements of the sets. The minimum value of membership for both the sets is taken to find the fuzzy intersection. Figure 3.18 depicts the intersection of two sets. The membership function of the intersection of two given fuzzy sets ($\bar{A}$ and $\bar{B}$) is defined as [92]:

\[
\mu_{\bar{A} \cap \bar{B}}(X) = \min(\mu_{\bar{A}}(X), \mu_{\bar{B}}(X)) \quad \forall x \in X
\]

(3.5)

![Figure 3.18. Intersection (AND logic operator)](image)
Union: Union (OR logic operator) is opposite of the intersection operator. Basically collection of elements of two or more sets into a single set is called union. The maximum membership value of both the sets is taken to find the fuzzy union. Figure 3.19 illustrates the union of two sets. The membership function of the union is defined as [92]:

\[
\mu_{\tilde{A} \cap \tilde{B}}(x) = \max(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)) \forall x \in X
\]  

(3.6)

![Figure 3.19. Union (OR logic operator)](image)

Complement: Complement (NOT logic operator) is basically the opposite of a given set. Its operation is just the same as the NOT operators in the classical sets. Figure 3.20 shows the complement of a set. The membership function of the union is defined as [92]:

\[
\mu_{\tilde{A}}(x) = 1 - \mu_{\tilde{A}}(x) \forall x \in X
\]  

(3.7)

![Figure 3.20. Complement (NOT logic operator)](image)
3.5.5 Linguistic Variables

A linguistic variable which is a fuzzy variable [93] are input or output variables of a system whose values are non-numeric. The values of linguistic variables are words or sentences from a natural language. A linguistic variable is normally divided into levels, each one called a Linguistic values or terms. For instance, “age is old” implies that the linguistic variable ‘age’ takes the linguistic term or value ‘old’. Figure 3.21 represents the hierarchal structure the linguistic variable ‘AGE’ [92]. As we can see it has four Linguistic values: Very young, young, old, very old.

![Hierarchal structure the linguistic variable ‘AGE’](image)

Figure 3.21. Hierarchal structure the linguistic variable ‘AGE’ [92]

Basically linguistic values are utilized to provide flexibility to linguistic variables, in order to check or manage the system. In the following chapters three input linguistic variables (Distance, Density and battery level of nodes), each having three linguistic values, and one output linguistic variable (Chance), having nine linguistic values are used in our fuzzy system.

3.5.6 Membership Functions

The classic set of a specific universe involves distinct elements, which have one or more feature in common. Each of these elements are known as characteristic functions, which could be determined by 0 to form a non-membership or 1 to form a full membership. The following equation shows how the characteristic function can be determined.
\[
\mu_A(x) = \begin{cases} 
1 & \text{if and only if } x \in A \\
0 & \text{if and only if } x \notin A
\end{cases}
\] (3.8)

Therefore, \( \mu_A(x) \) is a characteristic function for set \( A \), and \( A \) is a classical set of the universe.

As the fuzzy set is an extension of the classical set, the notion of binary characteristic functions are extended by the elements in a fuzzy set to multiple values on the continuous interval \([0, 1]\) as shown in the following equation.

\[
\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in U\}
\] (3.9)

Where, \( \tilde{A} \) represents a fuzzy set of the universe \( U \), and \( \mu_{\tilde{A}}(x) \) is the membership function of \( x \) in \( \tilde{A} \).

Membership function are utilized in the fuzzification and defuzzification steps of a fuzzy logic system to map the non-fuzzy input values to fuzzy linguistic values and the opposite way round. A membership function is utilized in order to quantify a linguistic value. For example, as shown in Figure 7 [94], membership functions for the linguistic values of temperature variable are given. A numerical value does not need to get fuzzified only using one particular membership function. To put it differently, a value might fit to several sets at the same instant. Considering the figure 3.22 a temperature value can be measured at the same time as “cold” and “too-cold”, by having different degree of memberships.

![Temperature](image)

**Figure 3.22** Membership Functions for temperature [94].
There exist several kinds of membership functions, some commonly used membership functions includes trapezoidal, triangular, bell-shaped etc. as shown in Figure 3.23 [95]. Most typical forms of membership functions are in fact trapezoidal, triangular and Gaussian shapes. The form of the membership function could in fact be context dependent which normally is selected arbitrarily as per the user experience [96].

![Commonly used membership functions](image)

**Figure 3.23.** Commonly used membership functions

### 3.5.7 Fuzzy IF-THEN Rules

After defining the linguistic terms and values, the rules of the fuzzy logic system are expressed. The fuzzy rules or IF-THEN rules, map the inputs of fuzzy system to its outputs. Fuzzy rules are statements comprised of three parts: fuzzy proposition, antecedent and consequence. One antecedent might have multiple of the AND= min or OR= max operators. The fuzzy IF-THEN rule can be expressed as:

\[
\text{if } x_1 \text{ is } D \text{ and/or } x_2 \text{ is } E \text{ then } x_3 \text{ is } F
\]  

(3.10)

Where \( D \), \( E \), and \( F \) are the linguistic values, while \( x_1 \), \( x_2 \), and \( x_3 \) are the linguistic variables.

Interpretation of an if-then rule contains different parts: first evaluation of the antecedent by fuzzifying the input and then applying every essential fuzzy operators, and finally applying that outcome to the consequent that is known as implication. These steps are shown in the following figure 3.24 [97].
3.5.8 Fuzzy Inference System

A fuzzy inference system (FIS) could be described as a system that maps inputs to outputs by using the fuzzy set theory. The procedure of FIS includes all the parts we have discussed previously such as fuzzy IF-THEN rules, fuzzy logical operation and membership functions. Figure 3.25 depicts the typical structure of a FIS which consist of four main modules: Fuzzification, Knowledge base, Inference engine and Defuzzification module.
Figure 3.25. Fuzzy Inference System

**Fuzzification:**

Transforming the system inputs (crisp inputs) into the fuzzy inputs by applying a fuzzification function.

**Knowledge-base:**

Storing the IF-THEN rules

**Inference engine:**

Simulates the human reasoning process by performing fuzzy inference operation on the inputs based on IF-THEN rules.

**Defuzzification module:**

Converting the fuzzy set obtained from the inference engine into a crisp value.

Fuzzy inference approaches could be classified into direct and indirect methods Figure 3.26. Direct methods are used more commonly due to being less complex, comparing with indirect methods.
Mamdani [93] and sugeno[98] are among the commonly used types of FIS. They have been used in several of application and differ from one another only in the way they obtain the outputs and defuzzification. Application of the FIS include expert systems, decision analysis, data classification etc. Since the Well-known Mamdani method proposed by Professor Ebrahim Mamdani [93] is less complex and let the expert knowledge to be described in a more instinctive style, it was used in this thesis. The process for applying Mamdani-style fuzzy inference could be broken down into four distinct steps: Fuzzification, Rule evaluation, Aggregate output(s) and Defuzzification explained next by the help of the example [90] given in Figure 3.27.

**Figure 3.26.** Fuzzy inference approaches

**Figure 3.27.** Defuzzification process [90]

**Step 1: Fuzzification**
Fuzzification includes the procedure of converting crisp input values into grades of membership function for linguistic values of fuzzy sets. A grade is associated to each linguistic value by the membership function. Considering the crisp inputs, we find out our input values from the fuzzy sets. As depicted in figure 3.27 the crisp input energy having value of 60 intersects the input fuzzy sets having a membership value of 0.3 and 0.5.

**Step 2: Rule evaluation**

Once the fuzzification values are obtained, (e.g 0.3 and 0.5) the rule-base of the FIS which is composed of IF-THEN rules has to be designed. Fuzzy operators such as AND or OR are used for fuzzy rule having multiple antecedents. Fuzzy operators are utilized to acquire only one number that could be described as the result of the antecedent evaluation. This result which is known as truth value will be used with the consequent membership function.

**Step 3: Aggregate output(s)**

Aggregation is basically the Procedure of bundling of the outputs of all rules. As all the decision within a fuzzy inference system is based on the checking and examining all the rules, the rules has to be bundled in a specific manner so the decision could be made. As shown in the given example, figure 3.27 above a new fuzzy set output is produced based on two fuzzy sets. The fuzzy logical operator OR is applied in mamdani FIS in order to achieve Aggregation.

**Step 4: Defuzzification**

Basically this is considered to be the last step where the final fuzzy system crisp output is determined. This step includes production of a crisp number out of a single fuzzy set output obtained from the previous step. Centroid [89] is considered as one of the most popular defuzzification methods in which a point representing the centre of gravity of the fuzzy set is found. The centre of gravity is a technique of calculation that takes all the subdivided area’s centre into consideration and allocates a specific weight according to the contribution to the whole to produce a weighted average point. It is calculated as follows:

\[
\text{COG} = \frac{\int \mu_A(x) \cdot x \, dx}{\int \mu_A(x) \, dx}
\]

(3.11)
Where, ∫ signifies an algebraic integration, and μ a(x) stands for degree of membership function of set A

Hence in the given example Figure 3.27, once defuzzification is done, it provides a value of 55 which is election chance of the cluster-head.

### 3.6 RELATED WORK

Hierarchical clustering routing protocols have proved to have a better performance regarding the energy conservation compared to flat routing protocols. The LEACH [73] which is the earliest hierarchical protocol, decreases the energy usage by selecting the CHs in a random style to distribute energy load evenly to each node. However the LEACH protocol is dependent upon the period of time to re-establish new clusters among the entire network without ever considering the leftover energy of the various clusters or their distance from the BS.

In some of the clustering algorithms fuzzy logic has been employed in order to deal with uncertainties in WSNs. Undoubtedly fuzzy logic is one of the best problem-solving control system strategies that delivers the best and easy way to get to a particular conclusion with vague, non-numerical, noisy, or absent input information. It makes use of heuristic knowledge as well as human reasoning to handle contradictory situations and inaccurate data. Fuzzy logic is basically used by fuzzy clustering algorithms in order to elect the most suitable sensor nodes cluster-heads.

In [90] fuzzy logic was used in order to find the CHs. The algorithm uses three fuzzy variables to select CHs. These variables are: Energy, concentration and centrality of sensor nodes. All the information regarding the nodes are collected by the BS. In this approach in each round there would be a single CH selected, while improving the network lifetime and balancing the energy consumption within the network requires more CHs to be elected.

The CHEF protocol [99], uses a Fuzzy Logic approach same as [90], with the aim of improving the network lifetime and balancing the energy consumption. The approach is almost similar to the algorithm introduced in [90] but in contrast the BS does not collect the information from all nodes but uses a localized CH selection mechanism based on Fuzzy Logic. A random number between 0 and 1 is chosen by each individual node. In case if this random number is less than $P_{opt}$, then the node computes the
chance making use of an FIS and announces a candidate message having the chance value.

Authors In [100] present a fuzzy decision-making approach in order to find and select the most appropriate nodes as CHs by using three criteria: density, distance from the BS and energy level of the nodes.

In [101] a FL based energy-aware dynamic clustering technique is introduced which uses just only two inputs (Node’s centrality and residual energy) and not considering many of other possible parameters (Such as distance between cluster heads, vulnerability index etc.), in order to improve lifetime of the WSN in terms of Last Node Dies (LND).

In [102], an energy efficient clustering algorithm is proposed for WSNs based on FL concept. The selection of CHs is done according to a threshold value. Then among the selected CHs based on some fuzzy input parameters a CH will be selected as a Super Cluster Head (SCH) which will be responsible for sending data to the BS. The main issue with this algorithm is the absent of checking the vulnerability of the SCH as it acts as the main node to send data to BS.
CHAPTER FOUR

PERFORMANCE ANALYSIS OF CLUSTERING ROUTING ALGORITHMS BASED ON FUZZY LOGIC
4.1 INTRODUCTION

As we noted earlier Clustering is among approaches to effectively increase the lifetime of WSNs. In a clustering process the close and neighbouring sensor nodes are put together to create virtual groups known as clusters. Among all the nodes in a cluster one node will be elected as a CH and the remaining nodes will act as the members of that cluster. A variety of computational approaches including fuzzy logic, genetic algorithms, artificial bee colony, evolutionary algorithms, neural networks, swarm intelligence etc. have been proposed in order to select the ideal sensor nodes as CHs in clustering routing algorithms designed especially for WSNs [90][103][104][105].

Among the aforementioned techniques, fuzzy logic is known to be one of the best problem-solving methodologies that gives an easy and effective way to reach to a certain conclusion even tough if there are imprecise, missing or noisy input information.
Fuzzy logic deals with imprecise, noisy and missing data inputs by using heuristic knowledge and human reasoning. Protocols designed specifically for WSN by employing fuzzy logic acquires the following advantages:

- Fuzzy logic is reasonably very effective at making real-time decisions, in spite of not complete information. Normal control systems depends on a reliable an accurate representation of a given environment, which actually in most of the cases is not the reality [90].
- Fuzzy logic delivers a great selection of operators to put together uncertain information in just a significantly better way compared to any other systems [90].
- Fuzzy logic is scalable, can be easily adapted, and moreover it needs fewer resources, less development cost, and less design time.

Several of the clustering approaches employ fuzzy logic in order to manage doubts and uncertainties in the WSNs. Within this chapter some clustering algorithms [90][99][106] which use the concept of fuzzy logic for CHs selection will be analyzed and compared against one another.

### 4.2 GUPTA FUZZY PROTOCOL

The Gupta protocol [90] employed a Fuzzy Logic approach in order to select CHs. The algorithm uses three fuzzy variables to elect the most appropriate CHs. These variables are: Energy, concentration and centrality of sensor nodes and each of them is divided into three levels. There is also one output called chance, which has been divided into seven levels. The system utilizes a 27 IF-THEN rules. All the information regarding the nodes are collected by the BS. There are two phases (set-up and steady-state) within this protocol just Like LEACH routing protocol. The only distinction among the two protocols is in the set-up phase where the BS requires to gather information about energy level and location for each nodes, and evaluate all of them in the designed FIS to determine the chance of every individual node for becoming a CH. The BS then elects the node which has the highest chance of becoming a CH. Following the CH selection, just about everything are in fact similar to LEACH.
4.3 CHEF FUZZY PROTOCOL

The CHEF protocol [99] (Cluster Head Election mechanism using Fuzzy Logic in Wireless Sensor Networks), also utilizes a Fuzzy Logic technique to increase the WSN lifetime. In fact it’s similar to the Gupta protocol, however in contrast to Gupta protocol CHEF does not require the BS to collect the information from all the SNs. Alternatively the CHEF protocol benefits from a localized CH selection system. A random number between 0 and 1 is chosen by each individual node. In case if this random number is less than $P_{opt}$, then the node computes the chance making use of an FIS and announces a candidate message having the chance value. The message points that the node is an applicant for becoming CH having the chance value. $P_{opt}$ is calculated as:

$$P_{opt} = p \times \alpha$$  \hspace{1cm} (4.1)

Where $p = k/N$; $k$ would be the anticipated number of CHs within the round and $N$ represents the number of nodes inside of the network and $\alpha$ is a constant value that is a definition of the ratio of the applicant for CH.

Following the CH selection, all the things are similar to LEACH. The FIS makes use of two variables: energy and location, one output and 9 IF-THEN rules.

4.4 FL-BASED CH SELECTION SYSTEM FOR WSNs

In [106] authors have proposed a new method for selection of CHs in WSNs based on fuzzy logic. The power reduction algorithm main part is a Fuzzy Logic Controller (FLC) which consists of four main units: Fuzzifier, inference engine, fuzzy rule base and a defuzzifier. The algorithm uses three fuzzy variables in order to elect the CHs. These parameters are: Distance, remaining energy, and traffic of sensor nodes and each of them is divided into three levels. There is also one output called probability (possibility), which has been divided into seven levels. The system uses both trapezoidal and triangular membership function in the design of fuzzy logic system. It also uses a 27 fuzzy IF-THEN rules.
4.5 SIMULATION AND PERFORMANCE ANALYSIS

Simulations are performed in order to analysis the energy efficiency of the clustering algorithms GUPTA [90], CHEF [99] and the system for CH selection proposed in [106]. The mentioned clustering algorithms based on fuzzy logic have been compared against one another. The simulation environment comprises of 1000 homogenous SNs deployed randomly over 1000 x 1000 m² networks. Each SN has an initial energy of 0.5J. In our experiment the communication energy parameters are set as $E_{elec} = 50$ nJ/bit, $E_{fs} = 10$ pJ/bit/m² and energy for data aggregation is set as $E_{DA} = 5$ nJ/bit/signal.

Various parameters used in the simulation are given in the Table 4.1.

<table>
<thead>
<tr>
<th>Sr.</th>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Environment Size</td>
<td>1000 x1000 m</td>
</tr>
<tr>
<td>2</td>
<td>Number of nodes</td>
<td>1000</td>
</tr>
<tr>
<td>3</td>
<td>Message Size</td>
<td>6400 bytes</td>
</tr>
<tr>
<td>4</td>
<td>Control Packet</td>
<td>100 bits</td>
</tr>
<tr>
<td>5</td>
<td>Initial energy per node ($E_0$)</td>
<td>0.5 J</td>
</tr>
<tr>
<td>6</td>
<td>$E_{elec}$</td>
<td>50 nJ / bit</td>
</tr>
<tr>
<td>7</td>
<td>$E_{fs}$</td>
<td>10 pJ/bit/ m²</td>
</tr>
<tr>
<td>8</td>
<td>$E_{mp}$</td>
<td>0.0013 pJ/bit/ m³</td>
</tr>
<tr>
<td>9</td>
<td>$E_{DA}$</td>
<td>5 nJ / bit</td>
</tr>
</tbody>
</table>

4.5.1 Lifetime of Network

Figure 4.1 depicts the total of dead nodes with respects to the functioning of the network in different rounds. Considering Figure 4.1, it is clearly observed that the algorithms using FL to select the CHs considerably effect and prolongs the lifetime of WSN. FL-base CH selection system [106] performs significantly better than others in the first 100 rounds, however in second half of rounds the difference in the performance of the routing protocols became less.
Figure 4.1. Number of dead nodes vs. Rounds

4.5.2 Energy Dissipation Of The Network of Network

Analysing the energy dissipation of the network can be very helpful to determine the energy efficiency of an algorithm. The Figure 4.2 shows the comparison of energy consumption among the aforementioned fuzzy clustering algorithms. It is noticeably observed that the algorithms using FL to select the CHs effect the balance of energy and offer a fair distribution in the WSN.

Figure 4.2. Energy Dissipation of Network vs. Rounds
4.5.3 Throughput of Network

By the term throughput we mean the total number of data packets which are received at the base station/sink. Considering Figure 4.3 it is noticed that the clustering algorithms using FL impacts and improve the throughput of the WSN significantly.

![Figure 4.3. Packet sent to the BS vs. Rounds](image)

4.5.4 Overall Network performance

it is observed that the aforementioned clustering algorithms take the advantage of FL to elect the most appropriate CHs by considering some vital information such as the energy level of nodes, their locations etc. Taking into account the number of dead nodes (Figure 4.1) and Packet sent to the BS (Figure 4.3) in each round, along with the overall energy dissipation of network (Figure 4.2) it is found that the energy consumption and overall performance of the network has been significantly improved.

4.6 Conclusion

Within this chapter the performance analysis of some clustering routing algorithms based on fuzzy logic has been examined and evaluated. By considering the number of dead nodes in each round and energy dissipation of the network, it was revealed that algorithms using FL for selecting the CHs significantly affect the energy consumption of the network which proves a better management and balance of the energy in the network. FL-base CH selection system [106] performs significantly better than others in the first 100 rounds, however in second half of rounds the difference in the performance of the routing protocols became less.
CHAPTER FIVE

TWO-STEP FUZZY LOGIC SYSTEM TO ELECT THE OPTIMAL CHs
TWO-STEP FUZZY LOGIC SYSTEM TO ELECT THE OPTIMAL CHs

5.1 INTRODUCTION

Popular methods such as Fuzzy Logic (FL) and Genetic Algorithm (GA) have been employed in various novel and existing algorithms to enhance the process of CHs election. There exist several algorithms based of FL to select the most proper CHs for the network. But these algorithms do not consider all the important parameters and information of the sensor nodes in order to guarantee the optimal selection of the CHs. In this chapter a new routing protocol based on FL is proposed, where a two-step FL system is used to select the appropriate CHs in order to reduce the energy consumption and extending the lifetime of the network. In first-step of the proposed algorithm, the main factors of each SN such as residual energy, density and distance to BS are checked and according to the output, most suitable candidates are selected. Furthermore in the second-step of FL system the vulnerability index, centrality and distance between CHs are checked and according to the output the CHs will be elected. In the proposed fuzzy system the usage of vulnerability index as a parameter for each sensor node has been introduced which assures the availability of the network as the nodes with high vulnerability index has a fewer chance to be chosen as the CHs. Centrality and distance between CHs are other factors which are checked to promise proper distribution of CHs within the network. The result of the simulation indicates that the proposed algorithm
performs better in case of fair distribution and balancing of the overall energy consumption comparing with other famous existing protocols.

### 5.2 RELATED WORK

In an effort to enhance the energy consumption of WSNs various techniques, such as data communication protocols [107-112], data collection [113-116], data aggregation [117-119] and coverage of the network [120-122] have been employed. LEACH protocol [73] which is the most primitive clustering-based algorithm was designed to reduce the energy consumption and prolonging the WSNs lifetime. In LEACH protocol, SNs are randomly arranged into pre-determined number of clusters and each cluster is allocated a CH at each round. To become a CH, a random number (between 0 and 1) is generated by each SN. SNs having generated numbers less than the threshold $T(n)$ (eq.1), will become CHs for that specific round. Randomized rotation of CHs are used to avoid draining the energy of a single SN. Re-election of a SN as a CH will be possible only if all the other candidates have been elected.

\[
T(n) = \begin{cases} 
\frac{p}{1 - p \times (r \mod \frac{1}{p})} & \text{if } n \in M \\
0 & \text{otherwise} 
\end{cases} 
\quad (5.1)
\]

where, $p$ stands for ratio of CHs in the WSNs (the value of $p$ is obtained through dividing the expected number of CHs in the current round by the number of nodes in WSN), $r$ indicates the round number and $M$ represents the set of SNs which have not been chosen as CH in the last $l/p$ rounds (if $n \notin M$, it won’t become a CH for that specific round as the $T(n)$ value for that node would be set to 0).

As it was mentioned earlier, because the CHs are randomly chosen in LEACH without considering important characteristics of SNs, it could result in inappropriate distribution or improper selection of CHs. FL is a decision system technique which works similar to human logic [123] offers a simple method to reach at a confident decision based upon unclear, vague, imprecise, or absent input information.

In [90], FL is used to select CHs based on three fuzzy variables, node’s energy, concentration and centrality. In each round there is only one chosen CH whereas
balancing the energy consumption and extending the network lifetime required more CHs to be selected.

Based on FL control in [50] a new routing protocol to extend the lifetime and throughput of WSNs has been proposed. The distance between cluster heads is not considered which might lead into improper distribution of CHs within the network. Nodes’ density, nodes’ battery level and distance of nodes form the base station are the only parameters used to select the CHs. Similarly Authors In [100] present a fuzzy decision-making approach to select the appropriate CHs by using three criteria including density, distance from the BS and energy level of the nodes. In [101] authors propose a FL based energy-aware dynamic clustering technique which uses just only two inputs (Node’s centrality and residual energy) and not considering many of other possible parameters (Such as distance between cluster heads, vulnerability index etc.), in order to improve lifetime of the WSN in terms of Last Node Dies (LND).

In [102], an energy efficient clustering algorithm has been proposed for WSNs based on FL concept. The selection of CHs is done according to a threshold value. Then among the selected CHs based on some fuzzy input parameters (remaining battery power, mobility and centrality) a CH will be selected as a Super Cluster Head (SCH) which will be responsible for sending data to the BS. The main issue with this algorithm is the absent of checking the vulnerability of the SCH as it acts as the main node to send data to BS.

In [99], to compute the chance for a node to become the CH, two fuzzy parameters, node’s energy and local distance are computed. In every round, SNs generate a random number and if the generated number is smaller than predefined threshold, the chance is calculated for the SNs. Therefore, some suitable nodes may not get selected because of this random number.

In this chapter a new routing protocol based on LEACH is proposed, where a two-step FL based algorithm is used to select the appropriate CHs. The main difference of this algorithm with the existent ones, is the number of parameters involved to select the ideal CHs. the proposed algorithms uses six parameters (residual energy, density and distance to BS, vulnerability index, centrality and distance between CHs) which guarantees the ideal selection of CHs. Consideration of vulnerability index for each SN...
enhances the availability of the WSN as SNs having high vulnerability index have less chance to be chosen as the CHs.

5.3 SYSTEM MODEL

Few basic assumptions are made about the sensor nodes, which are deployed to monitor an environment. These assumptions are as follows:

- WSN contains homogeneous SNs, having equivalent initial energy.
- SNs are deployed randomly.
- SNs and BS are kept stationary once deployed.
- All the SNs have battery-limited power and battery recharge is impractical.
- The distance of SNs from BS and other nodes is computed based on the strength of signals.

A simplified model [124] for radio energy dissipation is used which has been shown in Figure 5.1 and explained next.

\[
E_{\text{TX}}(l, d) = E_{\text{TX-\text{elec}}} + E_{\text{TX-\text{amp}}(l, d)} = \begin{cases} 
 lE_{\text{elec}} + l\varepsilon_{\text{fs}}d^2, & d < d_o \\
 lE_{\text{elec}} + l\varepsilon_{\text{mp}}d^4, & d \geq d_o 
\end{cases} (5.2)
\]

The consumed energy for transmitting an \( l \)-bit message over the distance \( d \), can be calculated with eq.2. \( E_{\text{RX}}(l) = lE_{\text{elec}} \)

\( (5.3) \)
5.4 PROPOSED ALGORITHM

Proposed algorithm uses FL system that is composed of two main steps in order to select the CHs. In the first step of FL system nodes are selected if they are in the proper locations with respects to the energy concerns. Furthermore in the second step, the level of vulnerability, centrality and desired distance between CHs of all the nodes (selected in the first step) are checked and according to the output, CHs will be elected. The second-step of the FL system enhances the availability of the network as the node with high vulnerability have less chance to be elected as CHs. The proposed FL system is explained in details as follows:

5.5 FIRST-STEP OF FL SYSTEM

5.5.1 FIS Design

As we stated earlier the system of FL includes four steps: fuzzification, rule evaluation, aggregation, and defuzzification. In the proposed model the most popular inference system, has been deployed, which is known as Mamdani fuzzy inference system. It lets us to define how the system is working in a straightforward and simplified mathematical way. The Fuzzy Logic system for the proposed protocol is illustrated in Figure 5.2.

![Figure 5.2. Fuzzy Inference System of First-Step](image)

5.5.2 FIS Parameters and Rules (First-Step)

The first step of the FL system selects the nodes based on their locations and energy levels. This step attempts, that nodes with adequate residual energy and proper location
will be selected for further consideration. The input functions named, residual energy, node density and distance to BS are used to convert the crisp inputs into fuzzy sets. Different membership function is used to illustrate the different level of the input functions as shown in Table 5.1 as well as Figure 5.3, Figure 5.4 and Figure 5.5.

**Table 5.1. Input Functions (First-Step of FL system)**

<table>
<thead>
<tr>
<th>INPUT</th>
<th>Membership Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Residual Energy</td>
<td>Low</td>
</tr>
<tr>
<td>Node Density</td>
<td>Low</td>
</tr>
<tr>
<td>Distance To BS</td>
<td>Low</td>
</tr>
</tbody>
</table>

**Figure 5.3. Membership functions of Residual Energy**

**Figure 5.4. Membership functions of Node Density**
The output (fuzzy cost) of the First-Step FL system, shows the probability level for each node to be chosen for further consideration in next step. The output function along with its membership functions are shown in Table 5.2 and Figure 5.6.

Table 5.2. Output Functions (First-Step of FL system)

<table>
<thead>
<tr>
<th>OUTPUT</th>
<th>Membership Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chance</td>
<td>Very small, Small, Rather</td>
</tr>
<tr>
<td></td>
<td>Small, Medium Small, Medium, Medium High,</td>
</tr>
<tr>
<td></td>
<td>Rather High, High, Very High</td>
</tr>
</tbody>
</table>

Figure 5.5. Membership functions of Distance to BS

Figure 5.6. Membership functions of Qualification in First-Step of FL system
The fuzzy if-then rules which are mostly based on experience and knowledge of experts in that domain, and relationship among the fuzzy input variables are given in Table 5.3 and Figure 5.7 respectively.

**Figure 5.7.** Relationship among fuzzy inputs in First-Step of FL system

**Table 5.3.** Fuzzy If-then rules in First-Step of FL system

<table>
<thead>
<tr>
<th>Rule</th>
<th>Residual Energy</th>
<th>Node Density</th>
<th>Distance to BS</th>
<th>fuzzy cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>High</td>
<td>High</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>2</td>
<td>High</td>
<td>High</td>
<td>Medium</td>
<td>Very High</td>
</tr>
<tr>
<td>3</td>
<td>High</td>
<td>High</td>
<td>Low</td>
<td>Very High</td>
</tr>
<tr>
<td>4</td>
<td>High</td>
<td>Medium</td>
<td>High</td>
<td>Rather High</td>
</tr>
<tr>
<td>5</td>
<td>High</td>
<td>Medium</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>6</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>7</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>Medium High</td>
</tr>
<tr>
<td>8</td>
<td>High</td>
<td>Low</td>
<td>Medium</td>
<td>Rather High</td>
</tr>
<tr>
<td>9</td>
<td>High</td>
<td>Low</td>
<td>Low</td>
<td>Rather High</td>
</tr>
<tr>
<td>10</td>
<td>Medium</td>
<td>High</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td>11</td>
<td>Medium</td>
<td>High</td>
<td>Medium</td>
<td>Medium High</td>
</tr>
<tr>
<td>12</td>
<td>Medium</td>
<td>High</td>
<td>Low</td>
<td>Medium High</td>
</tr>
<tr>
<td>13</td>
<td>Medium</td>
<td>Medium</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td>14</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>15</td>
<td>Medium</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>16</td>
<td>Medium</td>
<td>Low</td>
<td>High</td>
<td>Rather Small</td>
</tr>
<tr>
<td>17</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
<td>Medium small</td>
</tr>
<tr>
<td>18</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>Medium small</td>
</tr>
</tbody>
</table>
5.5.3 Chance determination of nodes in the First-Step FL system:

In this section with the help of the popular Mamdani fuzzy interface technique we illustrate and clarify how the first-step FIS is used to determine the chance value for the nodes. An example is considered as below:

Assume that, we have a sensor node with residual energy level = 0.03 J having the density value of 40 and it is positioned at a distance of 100 m from the BS. The following steps illustrates how the chance value is calculated in the first-step FIS.

5.5.3.1 Input of Crisp Value and Fuzzification

Based on the above crisp inputs the value of membership function which is basically the intersection of input values with the degree of the membership function is determined. The value of membership function will be used in the next step. The calculation of these degrees of membership functions are explained in Figures 5.8, 5.9 and 5.10.

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

\[
\mu_{\text{low}}(x) = \begin{cases} 
1 & \text{if } x < 0.012 \\
\frac{0.05 - x}{0.038} & \text{if } 0.012 \leq x \leq 0.05 \\
0 & \text{if } x > 0.05
\end{cases}
\]
So while energy is equal to 0.03 J the different membership function values, are calculated as:

\[
\mu_{\text{low}}(0.03) = \frac{0.05 - 0.03}{0.03} = 0.52
\]

\[
\mu_{\text{medium}}(0.03) = \frac{0.03 - 0.02}{0.03} = 0.33
\]

\[
\mu_{\text{high}}(0.03) = 0
\]

2) The membership function for density is calculated as follows:
\[ \mu_{\text{low}}(x) = \begin{cases} 
1 & \text{if } x < 8 \\
\frac{20 - x}{12} & \text{if } 8 \leq x \leq 20 \\
0 & \text{if } x > 20 
\end{cases} \]

\[ \mu_{\text{medium}}(x) = \begin{cases} 
0 & \text{if } x < 2 \\
\frac{x - 2}{18} & \text{if } 2 \leq x \leq 20 \\
\frac{38 - x}{18} & \text{if } 20 \leq x \leq 38 \\
0 & \text{if } x > 38 
\end{cases} \]

\[ \mu_{\text{high}}(x) = \begin{cases} 
0 & \text{if } x < 20 \\
\frac{x - 20}{11} & \text{if } 20 \leq x \leq 31 \\
1 & \text{if } x > 31 
\end{cases} \]

Hence for the density value of 40 we have:

\[ \mu_{\text{low}}(40) = 0 \]

\[ \mu_{\text{medium}}(40) = 0 \]

\[ \mu_{\text{high}}(40) = 1 \]

**Figure 5.9.** Fuzzification of crisp Density (=40)
3) The membership function for Distance to BS is calculated as follows:

\[ \mu_{\text{low}}(x) = \begin{cases} 
1 & \text{if } x < 52 \\
\frac{120 - x}{68} & \text{if } 52 \leq x \leq 120 \\
0 & \text{if } x > 120 
\end{cases} \]

\[ \mu_{\text{medium}}(x) = \begin{cases} 
0 & \text{if } x < 50 \\
\frac{x-50}{100} & \text{if } 50 \leq x \leq 150 \\
\frac{250-x}{100} & \text{if } 150 \leq x \leq 250 \\
0 & \text{if } x > 250 
\end{cases} \]

\[ \mu_{\text{high}}(x) = \begin{cases} 
0 & \text{if } x < 180 \\
\frac{x-180}{75} & \text{if } 180 \leq x \leq 255 \\
0 & \text{if } x > 255 
\end{cases} \]

Therefore, when distance to BS is equal to 100 the membership function values, are:

\[ \mu_{\text{low}}(100) = \frac{120-100}{68} = 0.29 \]

\[ \mu_{\text{medium}}(100) = \frac{100 - 50}{100} = 0.5 \]

\[ \mu_{\text{high}}(100) = 0 \]

Figure 5.10. Fuzzification of crisp distance to the BS (=100 m)
5.5.3.2 Rule Evaluation

After the completion of fuzzification step and obtaining the membership values, these values are supplied to our IF-THEN rules in order to determine our new fuzzy output set. Where, our fuzzy IF-THEN rules include several entrances, which are different variables, identified previously. The fuzzy logic operator (AND), which actually chooses minimum of all membership values is used of in order to obtain a single number as it has been illustrated in the Table 5.4.

Table 5.4. Fuzzy Inference System IF-THEN rules (Example for energy=0.3J, density=40, distance=100)

<table>
<thead>
<tr>
<th>Rule</th>
<th>Residual Energy</th>
<th>Node Density</th>
<th>Distance to BS</th>
<th>fuzzy cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>High(=0)</td>
<td>High(=1)</td>
<td>High(=0)</td>
<td>High(=0)</td>
</tr>
<tr>
<td>2</td>
<td>High(=0)</td>
<td>High(=1)</td>
<td>Medium(=0.5)</td>
<td>Very High(=0)</td>
</tr>
<tr>
<td>3</td>
<td>High(=0)</td>
<td>High(=1)</td>
<td>Low(=0.29)</td>
<td>Very High(=0)</td>
</tr>
<tr>
<td>4</td>
<td>High(=0)</td>
<td>Medium(=0)</td>
<td>High(=0)</td>
<td>Rather High(=0)</td>
</tr>
<tr>
<td>5</td>
<td>High(=0)</td>
<td>Medium(=0)</td>
<td>Medium(=0.5)</td>
<td>High(=0)</td>
</tr>
<tr>
<td>6</td>
<td>High(=0)</td>
<td>Medium(=0)</td>
<td>Low(=0.29)</td>
<td>High(=0)</td>
</tr>
<tr>
<td>7</td>
<td>High(=0)</td>
<td>Low(=0)</td>
<td>High(=0)</td>
<td>Medium High(=0)</td>
</tr>
<tr>
<td>8</td>
<td>High(=0)</td>
<td>Low(=0)</td>
<td>Medium(=0.5)</td>
<td>Rather High(=0)</td>
</tr>
<tr>
<td>9</td>
<td>High(=0)</td>
<td>Low(=0)</td>
<td>Low(=0.29)</td>
<td>Rather High(=0)</td>
</tr>
<tr>
<td>10</td>
<td>Medium(=0.33)</td>
<td>High(=1)</td>
<td>High(=0)</td>
<td>Medium(=0)</td>
</tr>
<tr>
<td>11</td>
<td>Medium(=0.33)</td>
<td>High(=1)</td>
<td>Medium(=0.5)</td>
<td>Medium High(=0.33)</td>
</tr>
<tr>
<td>12</td>
<td>Medium(=0.33)</td>
<td>High(=1)</td>
<td>Low(=0.29)</td>
<td>Medium High(=0.29)</td>
</tr>
<tr>
<td>13</td>
<td>Medium(=0.33)</td>
<td>Medium(=0)</td>
<td>High(=0)</td>
<td>Medium small(=0)</td>
</tr>
<tr>
<td>14</td>
<td>Medium(=0.33)</td>
<td>Medium(=0)</td>
<td>Medium(=0.5)</td>
<td>Medium(=0)</td>
</tr>
<tr>
<td>15</td>
<td>Medium(=0.33)</td>
<td>Medium(=0)</td>
<td>Low(=0.29)</td>
<td>Medium(=0)</td>
</tr>
<tr>
<td>16</td>
<td>Medium(=0.33)</td>
<td>Low(=0)</td>
<td>High(=0)</td>
<td>Rather Small(=0)</td>
</tr>
<tr>
<td>17</td>
<td>Medium(=0.33)</td>
<td>Low(=0)</td>
<td>Medium(=0.5)</td>
<td>Medium small(=0)</td>
</tr>
<tr>
<td>18</td>
<td>Medium(=0.33)</td>
<td>Low(=0)</td>
<td>Low(=0.29)</td>
<td>Medium small(=0)</td>
</tr>
<tr>
<td>19</td>
<td>Low= (0.52)</td>
<td>High(=1)</td>
<td>High(=0)</td>
<td>Small(=0)</td>
</tr>
<tr>
<td>20</td>
<td>Low= (0.52)</td>
<td>High(=1)</td>
<td>Medium(=0.5)</td>
<td>Rather Small(=0.5)</td>
</tr>
<tr>
<td>21</td>
<td>Low= (0.52)</td>
<td>High(=1)</td>
<td>Low(=0.29)</td>
<td>Rather Small(=0.29)</td>
</tr>
</tbody>
</table>
5.5.3.3 Aggregation of the Rule Outputs

Once the fuzzification and rule evaluation is done, the next step will be the aggregation step. It is basically a process of the union of all the outputs acquired by applying all the rules in the FIS model. As we want to aggregate all the rules, then the OR fuzzy logic operator has been used. It chooses the maximum value of our rule evaluation, in order to produce the new aggregate fuzzy set which will be used in next step. The Figure 5.11 shows the aggregation output of the rules.

![Figure 5.11. Output evaluation of fuzzy if-then rules of First-Step of FL system](image)

5.5.3.4 Defuzzification

Defuzzification would be the last step in our First-Step fuzzy Inference system, in which the first priority for the SNs will be calculated and SNs with high priority in this step will be considered in the second-step of FL system where the actual CHs are selected. Considering that centroid, it appears as one of the most widely used
defuzzification method and it actually has been effectively utilized in numerous applications. So it has been decide to be used in this work. Centroid defuzzification technique is designed to obtain the final crisp number. By using the values we received formerly from step 3 within the equation 5.4 and calculating the algebraic integration, we determine the first priority value for the SN is approximately equal to (= 36.71) as depicted in Figure 5.12. The calculation for COG is given as follows:

\[
COG = \frac{\int \mu_A(x) \cdot x \, dx}{\int \mu_A(x) \cdot dx}
\]  

(5.4)

\[
COG = \left[ 0.5 \int_{20}^{25.19} (x - 20) \, dx + \int_{25.19}^{35.22} (0.5) \, dx 
+ 0.5 \int_{35.22}^{40} (40 - x) \, dx
+ 0.33 \int_{56.22}^{57.27} (x - 56.22) \, dx
+ 0.33 \int_{57.27}^{67.59} (0.33) \, dx
\right] \frac{1}{\left[ 0.5 \int_{20}^{25.19} (x - 20) \, dx + \int_{25.19}^{35.22} (0.5) \, dx 
+ 0.5 \int_{35.22}^{40} (40 - x) \, dx
+ 0.33 \int_{56.22}^{57.27} (x - 56.22) \, dx
+ 0.33 \int_{57.27}^{67.59} (0.33) \, dx + 0.33 \int_{67.59}^{70} (70 - x) \, dx \right]}

= \frac{36.71}{36.71} = 36.71
Figure 5.12. The Centroid point for first-step of FL system

COG returns the Centre of Area under the curve. The COG basically would be the point alongside the x-axis about where this shape would be balanced [51]. Some more examples of priorities of SNs in the first-step FL system have been provided by making use of mamdani FIS as well as centroid defuzzification. These examples are given in Table 5.5.

Table 5.5. Priority value for different SNs in First-step FL system

<table>
<thead>
<tr>
<th>Residual Energy</th>
<th>Density</th>
<th>Distance to BS</th>
<th>Chance</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.016</td>
<td>19</td>
<td>152</td>
<td>20</td>
</tr>
<tr>
<td>0.042</td>
<td>28</td>
<td>98</td>
<td>55.3</td>
</tr>
<tr>
<td>0.071</td>
<td>10</td>
<td>155</td>
<td>70</td>
</tr>
<tr>
<td>0.081</td>
<td>36</td>
<td>35</td>
<td>94.5</td>
</tr>
<tr>
<td>0.011</td>
<td>4</td>
<td>173</td>
<td>4</td>
</tr>
<tr>
<td>0.072</td>
<td>20</td>
<td>140</td>
<td>80.4</td>
</tr>
<tr>
<td>0.033</td>
<td>28</td>
<td>173</td>
<td>25</td>
</tr>
<tr>
<td>0.072</td>
<td>33</td>
<td>62</td>
<td>93.5</td>
</tr>
<tr>
<td>0.089</td>
<td>37</td>
<td>29</td>
<td>95</td>
</tr>
</tbody>
</table>
5.6 Second-Step of FL system

5.6.1 FIS Parameters and Rules (second-step)

Same as First-Step, mamdani fuzzy inference system has been used as the fuzzy interface as shown in figure 5.13. In the second-step of FL system the qualified nodes (selected in first-step) are re-evaluated and checked based on three parameters: vulnerability index, centrality and distance between CHs. The main purpose of this step is to enhance the availability of network by considering vulnerability index (Nodes with high vulnerability index have lesser chance to be elected as CHs) and proper distribution of CHs by checking their distances. Input functions along with their different membership functions are given in table 5.6, and Figures 5.14, 5.15 and 5.16.

![Fuzzy Inference System of Second-Step](image)

**Fig 5.13.** Fuzzy Inference System of Second-Step

**Table 5.6.** Input Functions (Second-Step of FL system)

<table>
<thead>
<tr>
<th>INPUT</th>
<th>Membership Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vulnerability Index</td>
<td>Low  Medium  High</td>
</tr>
<tr>
<td>Centrality</td>
<td>Low  Medium  High</td>
</tr>
<tr>
<td>Distance B/W CHs</td>
<td>Low  Medium  High</td>
</tr>
</tbody>
</table>
Figure 5.14. Membership functions of Vulnerability Index

Figure 5.15. Membership functions of Centrality

Figure 5.16. Membership functions of Distance B/W CHs

The output (fuzzy cost) of Second-Step FL system, shows the possibility of SNs to be chosen as CHs. Figure 5.17 and table 5.7 depict the output function along with its membership functions.
Figure 5.17. Membership functions of Qualification in Second-Step of FL system

Table 5.7. Output Functions (Second-Step of FL system)

<table>
<thead>
<tr>
<th>OUTPUT</th>
<th>Membership Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chance</td>
<td>Very small, Small, Rather Small, Medium Small, Medium, Medium, High, Rather High, High, Very High</td>
</tr>
</tbody>
</table>

The relationship among the fuzzy input variables (vulnerability index, centrality and distance between CHs), and fuzzy if-then rules in the Second-Step FL system have been given in Figure 5.18 and Table 5.8 respectively.

Fig. 5.18. Relationship among fuzzy inputs in Second-Step of FL system

Table 5.8. Fuzzy If-then rules in Second-Step of FL system

<table>
<thead>
<tr>
<th>Rule</th>
<th>Vulnerability Index</th>
<th>Centrality</th>
<th>Distance b/w CHs</th>
<th>Fuzzy cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>High</td>
<td>High</td>
<td>High</td>
<td>Small</td>
</tr>
<tr>
<td></td>
<td>HIGH</td>
<td>HIGH</td>
<td>MEDIUM</td>
<td>VERY SMALL</td>
</tr>
<tr>
<td>---</td>
<td>------</td>
<td>------</td>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>2</td>
<td>High</td>
<td>High</td>
<td>Medium</td>
<td>Very small</td>
</tr>
<tr>
<td>3</td>
<td>High</td>
<td>High</td>
<td>Low</td>
<td>Very small</td>
</tr>
<tr>
<td>4</td>
<td>High</td>
<td>Medium</td>
<td>High</td>
<td>Small</td>
</tr>
<tr>
<td>5</td>
<td>High</td>
<td>Medium</td>
<td>Medium</td>
<td>Small</td>
</tr>
<tr>
<td>6</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
<td>Very Small</td>
</tr>
<tr>
<td>7</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>Rather Small</td>
</tr>
<tr>
<td>8</td>
<td>High</td>
<td>Low</td>
<td>Medium</td>
<td>Rather Small</td>
</tr>
<tr>
<td>9</td>
<td>High</td>
<td>Low</td>
<td>Low</td>
<td>Small</td>
</tr>
<tr>
<td>10</td>
<td>Medium</td>
<td>High</td>
<td>High</td>
<td>Medium Small</td>
</tr>
<tr>
<td>11</td>
<td>Medium</td>
<td>High</td>
<td>Medium</td>
<td>Medium Small</td>
</tr>
<tr>
<td>12</td>
<td>Medium</td>
<td>High</td>
<td>Low</td>
<td>Rather Small</td>
</tr>
<tr>
<td>13</td>
<td>Medium</td>
<td>Medium</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td>14</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>15</td>
<td>Medium</td>
<td>Medium</td>
<td>Low</td>
<td>Medium small</td>
</tr>
<tr>
<td>16</td>
<td>Medium</td>
<td>Low</td>
<td>High</td>
<td>Medium high</td>
</tr>
<tr>
<td>17</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
<td>Medium high</td>
</tr>
<tr>
<td>18</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>19</td>
<td>Low</td>
<td>High</td>
<td>High</td>
<td>Rather High</td>
</tr>
<tr>
<td>20</td>
<td>Low</td>
<td>High</td>
<td>Medium</td>
<td>Rather High</td>
</tr>
<tr>
<td>21</td>
<td>Low</td>
<td>High</td>
<td>Low</td>
<td>Medium High</td>
</tr>
<tr>
<td>22</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>23</td>
<td>Low</td>
<td>Medium</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>24</td>
<td>Low</td>
<td>Medium</td>
<td>Low</td>
<td>Rather High</td>
</tr>
<tr>
<td>25</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
<td>Very High</td>
</tr>
<tr>
<td>26</td>
<td>Low</td>
<td>Low</td>
<td>Medium</td>
<td>Very High</td>
</tr>
<tr>
<td>27</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
</tr>
</tbody>
</table>

### 5.6.2 Chance determination of nodes in the Second-Step FL system:

In this section we exemplify how the Second-Step FIS is used to determine the chance value of the nodes (which have been elected in the First-Step of the FL system) to be considered as the CHs. An example is considered as below:
Assuming that, we have a sensor node with vulnerability index = 0.7 having the centrality value of 0.75 and it is positioned at a distance of 10 m from another CH (if we have multiple nodes that have been selected as CHs, then the distance of the closest CH will be considered). The following steps shows how the final chance value of the candidate nodes for becoming CHs, are calculated.

5.6.2.1 Input of Crisp Value and Fuzzification of Second-Step FL system

Similarly, as we did in the First-Step of the FL System, based on the given crisp inputs (vulnerability index = 0.7, centrality value = 0.75 and Distance b/w CHs = 10m) the different value of membership functions are determined. These values will be used in the next step. As it was mentioned previously value of membership function is the intersection of input values with the degree of the membership function. The calculation of these degrees of membership functions are explained in Figures 5.19, 5.20 and 5.21.

1) The membership function of the vulnerability index is calculated as:

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

\[
\mu_{\text{medium}}(x) = \begin{cases} 
0 & \text{if } x < 0.2 \\
\frac{x-0.2}{0.3} & \text{if } 0.2 \leq x \leq 0.8 \\
0 & \text{if } x > 0.8
\end{cases}
\]

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

So if the Vulnerability index is equal to 0.7 J the membership function values, are:

\[\mu_{\text{low}}(0.7) = 0\]

\[\mu_{\text{medium}}(0.7) = \frac{0.8-0.7}{0.3} = 0.33\]
\[ \mu_{\text{high}}(0.7) = \frac{0.7 - 0.5}{0.4} = 0.5 \]

\[ \mu_{\text{low}}(x) = \begin{cases} 
1 & \text{if } x < 0.1 \\
\frac{0.4 - x}{0.3} & \text{if } 0.1 \leq x \leq 0.4 \\
0 & \text{if } x > 0.4 
\end{cases} \]

\[ \mu_{\text{medium}}(x) = \begin{cases} 
0 & \text{if } x < 0.2 \\
\frac{x - 0.2}{0.3} & \text{if } 0.2 \leq x \leq 0.5 \\
\frac{0.9 - x}{0.4} & \text{if } 0.5 \leq x \leq 0.9 \\
0 & \text{if } x > 0.9 
\end{cases} \]

\[ \mu_{\text{high}}(x) = \begin{cases} 
0 & \text{if } x < 0.7 \\
\frac{x - 0.7}{0.2} & \text{if } 0.7 \leq x \leq 0.9 \\
1 & \text{if } x > 0.9 
\end{cases} \]

**Fig 5.19.** Fuzzification of crisp Vulnerability index (=0.07)

2) The membership function of centrality is calculated as follows:

\[ \mu_{\text{low}}(x) = \begin{cases} 
1 & \text{if } x < 0.1 \\
\frac{0.4 - x}{0.3} & \text{if } 0.1 \leq x \leq 0.4 \\
0 & \text{if } x > 0.4 
\end{cases} \]
Hence for the centrality value of 0.75 we have:

\[
\mu_{\text{low}}(0.75) = 0 \\
\mu_{\text{medium}}(0.75) = \frac{0.9 - 0.75}{0.4} = 0.37 \\
\mu_{\text{high}}(0.75) = \frac{0.8 - 0.75}{0.2} = 0.25
\]

![Graph showing fuzzification of crisp Centrality (=0.75)](image)

**Figure 5.20.** Fuzzification of crisp Centrality (=0.75)

3) The membership function for Distance between CHs, is:

\[
\mu_{\text{low}}(x) = \begin{cases} 
1 & \text{if } x < 10 \\
\frac{26 - x}{16} & \text{if } 10 \leq x \leq 26 \\
0 & \text{if } x > 26 
\end{cases}
\]

\[
\mu_{\text{medium}}(x) = \begin{cases} 
0 & \text{if } x < 17 \\
\frac{x - 17}{18} & \text{if } 17 \leq x \leq 35 \\
\frac{53 - x}{18} & \text{if } 35 \leq x \leq 53 \\
0 & \text{if } x > 53 
\end{cases}
\]

\[
\mu_{\text{high}}(x) = \begin{cases} 
0 & \text{if } x < 44 \\
\frac{x - 44}{20} & \text{if } 44 \leq x \leq 64 \\
1 & \text{if } x > 64 
\end{cases}
\]
Therefore, when distance of a SN to the closest CH is equal to 15 the membership function values, are:

$$\mu_{low}(10) = \frac{26 - 10}{16} = 1$$

$$\mu_{medium}(10) = 0$$

$$\mu_{high}(10) = 0$$

Fig 5.21. Fuzzification of crisp distance between CHs (=10 m)

### 5.6.2.2 Rule Evaluation

Similarly as we did in the First-step of the FL system, after the completion of fuzzification step in Second-Step of FL system and obtaining the membership values, values are supplied to IF-THEN rules in order to decide our new fuzzy output set. Where, our fuzzy IF-THEN rules contain several entrances, which are different variables, identified previously. Similarly as we did in the First-step of the FL system, the fuzzy logic operator (AND), is used to obtain a single number as it has been illustrated in the Table 5.9.

Table 5.9. Fuzzy Inference System IF-THEN rules of Second-Step of FL system

<table>
<thead>
<tr>
<th>Rule</th>
<th>Vulnerability Index</th>
<th>Centrality</th>
<th>Distance b/w CHs</th>
<th>Fuzzy cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>High (=0.5)</td>
<td>High (=0.25)</td>
<td>High (=0)</td>
<td>Small (=0)</td>
</tr>
<tr>
<td>2</td>
<td>High (=0.5)</td>
<td>High (=0.25)</td>
<td>Medium (=0)</td>
<td>Very small (=0)</td>
</tr>
<tr>
<td>3</td>
<td>High (=0.5)</td>
<td>High (=0.25)</td>
<td>Low (=1)</td>
<td>Very small (=0.25)</td>
</tr>
<tr>
<td>Rule</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
<td>Small</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>--------</td>
<td>-----</td>
<td>-------</td>
</tr>
<tr>
<td>4</td>
<td>0.5</td>
<td>0.37</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0.5</td>
<td>0.37</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0.5</td>
<td>0.37</td>
<td>1</td>
<td>0.37</td>
</tr>
<tr>
<td>7</td>
<td>0.5</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>0.5</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>0.5</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0.33</td>
<td>0.25</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0.33</td>
<td>0.25</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>0.33</td>
<td>0.25</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>13</td>
<td>0.33</td>
<td>0.37</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>0.33</td>
<td>0.37</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>0.33</td>
<td>0.37</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>0.33</td>
<td>0</td>
<td>0</td>
<td>0.33</td>
</tr>
<tr>
<td>17</td>
<td>0.33</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>0.33</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>0</td>
<td>0.25</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>0.25</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>21</td>
<td>0</td>
<td>0.25</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>0</td>
<td>0.37</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>23</td>
<td>0</td>
<td>0.37</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>24</td>
<td>0</td>
<td>0.37</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>25</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>26</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>27</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**5.6.2.3 Aggregation of the Rule Outputs**

Similarly as we did in the First-Step of FL system, once the fuzzification and rule evaluation is done, aggregation step will start. As we mentioned previously, aggregation is a process of the union of all the outputs. As want to aggregate all our rules, then the OR fuzzy logic operator has been used. The Figure 5.22 shows the aggregation output of the rules of Second-Step of FL system.
5.6.2.4 Defuzzification

Similarly as we did in the First-Step of FL system defuzzification would be done. The difference is here in this step (Second-Step of FL system) the value of defuzzification will be the actual priority value for CHs but in first step (First-Step of FL system) value obtained from defuzzification was a priority for SNs to be considered in this step of FL system. Same as previous the Centroid defuzzification technique is used to obtain the final crisp number which is the actual chance for a SN to be elected as a CH. By using the values we received formerly from step 3 (Second-Step of FL system) within the equation 5.4 and calculating the algebraic integration, we determine the final priority value for the SN (We assumed that this sensor node was chosen in the First-Step of FL system so it was considered in the second step) to be a CH is approximately equal to (= 23.9) as depicted in Figure 5.23. The calculation for COG is shown as follows:
\[
COG = \left[ \int_{0}^{11.41} (0.37) \, dx \right.
\]
\[
+ 0.37 \left[ \int_{11.41}^{15} (15 - x) \, dx \right.
\]
\[
+ 0.25 \left[ \int_{15}^{17.53} (x - 15) \, dx \right.
\]
\[
+ \left. \int_{17.53}^{27.56} (0.25) \, dx \right] + 0.33 \left[ \int_{27.56}^{28.22} (x - 27.56) \, dx \right.
\]
\[
+ \left. \int_{28.22}^{42.94} (0.33) \, dx \right] + 0.33 \left[ \int_{42.94}^{47.02} (47.02 - x) \, dx \right]
\]
\[
\left. \div \left[ \int_{0}^{11.41} (0.37) \, dx \right. \right.
\]
\[
+ 0.37 \left[ \int_{11.41}^{15} (15 - x) \, dx \right.
\]
\[
+ 0.25 \left[ \int_{15}^{17.53} (x - 15) \, dx \right.
\]
\[
+ \left. \int_{17.53}^{27.56} (0.25) \, dx \right] + 0.33 \left[ \int_{27.56}^{28.22} (x - 27.56) \, dx \right.
\]
\[
+ \left. \int_{28.22}^{42.94} (0.33) \, dx \right] + 0.33 \left[ \int_{42.94}^{47.02} (47.02 - x) \, dx \right]
\]
\[
= 23.9
\]

**Figure 5.23.** The Centroid point for Second-Step of FL system

Some more examples of priorities of SNs to be elected as CHs in the Second-Step FL system have been provided by making use of mamdani FIS as well as centroid defuzzification. These examples are given in Table 5.10.
Table 5.10. Priority value for different SNs to be CHs

<table>
<thead>
<tr>
<th>Vulnerability Index</th>
<th>Centrality</th>
<th>Distance B/W CHs</th>
<th>Chance(Output)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.68</td>
<td>0.19</td>
<td>52</td>
<td>45.4</td>
</tr>
<tr>
<td>0.26</td>
<td>0.28</td>
<td>65</td>
<td>70.3</td>
</tr>
<tr>
<td>0.74</td>
<td>0.1</td>
<td>70</td>
<td>37.4</td>
</tr>
<tr>
<td>0.9</td>
<td>0.65</td>
<td>35</td>
<td>15</td>
</tr>
<tr>
<td>0.1</td>
<td>0.43</td>
<td>17</td>
<td>75</td>
</tr>
<tr>
<td>0.39</td>
<td>0.2</td>
<td>14</td>
<td>57.5</td>
</tr>
<tr>
<td>0.2</td>
<td>0.4</td>
<td>15</td>
<td>75</td>
</tr>
<tr>
<td>0.5</td>
<td>0.3</td>
<td>35</td>
<td>53.3</td>
</tr>
<tr>
<td>0.8</td>
<td>0.37</td>
<td>31</td>
<td>16.6</td>
</tr>
</tbody>
</table>

5.7 SIMULATION AND EVALUATION

Simulations are performed in order to verify the efficiency of the proposed algorithm. The simulation environment comprises of two different scenarios, 60 homogenous SNs deployed randomly over 120 x 120 and 60 x 60 m$^2$ networks. Each SN has an initial energy of 0.1J. In our experiment the communication energy parameters are set as $E_{elec} = 50$ nJ/bit, $E_{fs} = 10$ pJ/bit/m$^2$ and $E_{mp} = 0.0013$ pJ/bit/m$^4$ and energy for data aggregation is set as $E_{DA} = 5$ nJ/bit/signal. The Table 5.11 shows the parameters used for simulation.

Table 5.11 parameters used for simulation

<table>
<thead>
<tr>
<th>Sr.</th>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Environment Size</td>
<td>120 x120</td>
</tr>
<tr>
<td>2</td>
<td>Number of nodes</td>
<td>60</td>
</tr>
<tr>
<td>3</td>
<td>Message Size</td>
<td>4000 bits</td>
</tr>
<tr>
<td>4</td>
<td>Election Probability value of Cluster-heads (P)</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>Number of rounds(r$_{max}$)</td>
<td>200 rounds</td>
</tr>
<tr>
<td>6</td>
<td>Initial energy per node (E$_0$)</td>
<td>0.1 J</td>
</tr>
<tr>
<td>7</td>
<td>$E_{elec}$</td>
<td>50 nJ / bit</td>
</tr>
<tr>
<td>8</td>
<td>$E_{fs}$</td>
<td>10 pJ/bit/ m$^2$</td>
</tr>
<tr>
<td>9</td>
<td>$E_{mp}$</td>
<td>0.0013 pJ/bit/ m$^4$</td>
</tr>
<tr>
<td>10</td>
<td>$E_{DA}$</td>
<td>5nJ / bit</td>
</tr>
</tbody>
</table>
The proposed algorithm has been compared against LEACH [73], GUPTA [90] and CHEF [99] in case of network’s Lifetime, energy dissipation and variance of energy.

The number of alive nodes with regard to network’s operation for each scenario, have been given in figures 5.24 and 5.25. Regarding the death time of first node, it’s well observed that, the proposed algorithm performs much better in both of the cases. However while the network size is 120 x 120 m$^2$, the CHEF algorithm achieves slightly better performance than the proposed and other algorithms regarding the death of last node as it’s shown in figure 5.25.

![Figure 5.24. Number of alive nodes (network size 60x60)](image1)

![Figure 5.25. Number of alive nodes (network size 120x120)](image2)
To verify the energy effectiveness of the proposed algorithm, the residual energy of the WSN in each round has been analysed. The figures 5.26 and 5.27 compare the energy dissipation of different protocols. Considering these figures it's found that in both of the situations (network size of 60x60 and 120x120 m$^2$) the curve of the proposed algorithm is smoother than the rest of protocols, which verifies its better performance in case of fair distribution and balance of the overall energy.

Figure 5.26. Residual energy of the WSN (network size 60x60)

Figure 5.27. Residual energy of the WSN (network size 120x120)
Another way to validate the reasonable consumption of the energy within the WSNs, is to study the variance of overall energy with regard to network’s operation. Having slighter variance of residual energy confirms a reasonable energy consumption. The figures 5.28 and 5.29 depict the variance of energy in both of the scenarios in different rounds. In both of the cases, the proposed algorithm appears to have smaller variance of energy.

**Figure 5.28.** Energy variance of the WSN (network size 60x60)

**Figure 5.29.** Energy variance of the WSN (network size 120x120)
5.8 CONCLUSION

In this chapter, a new routing protocol based on fuzzy logic is proposed, where a two-step FL system is used to select the most appropriate sensor nodes as CHs. The cluster heads nodes selection is based on six descriptors; residual energy, density and distance to BS, vulnerability index, centrality and distance between CHs. Conceptually, there exist several algorithms based on fuzzy logic. But it is realized that they do not take into account all the significant information of the sensor nodes in order to guarantee the optimal selection of the CHs. Comparison of our proposed routing protocol against some other similar approaches indicates that, the proposed algorithm performs better in case of fair distribution and balancing of the overall energy consumption. Additionally, the result shows a smaller variance of energy for the proposed algorithm which also validates its reasonable consumption of the energy.
CHAPTER SIX

BOKHARI-SEPFL ROUTING ALGORITHM
6.1 INTRODUCTION

A wireless Sensor Network is comprised of many sensor nodes and a base station (BS). These nodes are tiny in size and have restricted and limited, power supply, memory and processing capability. Each particular sensor is made up of four main units, power supply unit, communication unit, processing unit and a sensing unit. WSNs can be employed in number of applications such as environmental, biomedical and military applications. WSNs are expected to be joined into the ‘Internet of Things’ [125] where the WSNs join the Internet, and utilize it to work together and achieve their tasks. Despite the goals and objectives of sensor applications, the main task of wireless sensor nodes would be to sense and gather data of a potential field, and sending them to specific sites. Achievement in this task efficiently needs the design of energy-efficient routing protocol. Many routing algorithms have been optimized and developed in this regard. One of the most significant routing strategy for WSNs is the hierarchal or cluster-based routing due to their high efficiency and scalability. In hierarchical routing protocol nodes with high energy are used to send data to the BS while the nodes with low energy are used to sense and collect data of a specific area. LEACH [73], TEEN [75], and PEGASIS [74] are some of well-known hierarchical routing protocol. The LEACH routing protocol was one of the earliest clustering-based routing protocol. It improves the energy consumption of the network by selecting the CH nodes randomly to allocate energy load evenly to all the nodes. SEP [126] (Stable Election Protocol) is another type of cluster-based routing protocol which includes two tier energy level, in order to prolong the lifetime of the network. Taking advantage of intelligent approaches such as fuzzy logic within the routing protocols could greatly improves the efficiency of WSNs.
For those applications requiring real-time decision making, fuzzy logic is a strong and powerful tool that would make decisions no matter if there exists inadequate data. In this chapter fuzzy logic control is used to enhance the process of cluster head selection in the existing SEP protocol based on three variables, distance of nodes form BS, density of nodes and the battery level of nodes to improve the lifetime of and throughput of the WSN. The result of the simulation which has been done in MATLAB[127] indicates that our proposed routing protocols Stable Election Protocol Using Fuzzy Logic (BOKHARI-SEPFL) improves the lifetime of the network significantly by almost 73.2% and 42.4%, comparing with the SEP and its enhanced version SEP-E [128] protocol respectively.

### 6.2 RELATED WORK

LEACH (Low-Energy Adaptive Clustering Hierarchy) [73] is one of the earliest and most well-known clustering-based routing protocol, designed for WSNs to minimize the power consumption. In LEACH, the sensor nodes arrange themselves into number of clusters, with one sensor node acting as the CH at each round. LEACH uses randomized rotation of the high-energy cluster-head position in an attempt to not drain the battery of a single sensor. Additionally, LEACH does data aggregation and fusion to compress and reduce the size of data being sent to the base station, in order to enhance energy dissipation and improving system lifetime.

SEP (A Stable Election Protocol) [126] is an enhanced version of LEACH protocol. It has been designed for two-level heterogeneous networks. The protocol includes two types of nodes advanced and normal nodes, different from one another in term of their initial energy. The protocol working is almost same as LEACH but with this distinction that in SEP advanced nodes that have a higher initial energy get more frequently elected as CHs. This is to assure a uniform usage of node energy and prolong the lifetime of the network. In case of SEP protocol the \( p \) (desired percentage of CHs) for normal and advance nodes has been given in equation (2) and (3) respectively:

For normal nodes:

\[
 p_{norSEP} = \frac{p_{opt}}{1+\alpha.m} \tag{6.1}
\]
For advance nodes:

\[
p_{\text{advSEPa}} = \frac{p_{\text{opt}}}{1+\alpha.m} \times (1 + \alpha) \tag{6.2}
\]

Where \( m \) is the percentage of advanced nodes, \( \alpha \) stands for additional energy of advanced nodes and \( p_{\text{opt}} \) is the optimal probability of a given node to become the CH.

The new advanced nodes does not have any effect on the density of the network but, the overall energy of the network changes. Energy level of the every advanced node rises by \((1+\alpha)\) times and energy level of the whole network by \((1+\alpha.m)\). Presume that \( E_0 \) is the initial energy level for every normal sensor, so energy of each advanced nodes and the total network can be calculated respectively as:

\[
E_{\text{advanced}} = (1+\alpha).E_0 \tag{6.3}
\]

\[
E_{\text{totalSEP}} = n.E_0 (1+\alpha.m) \tag{6.4}
\]

In SEP protocol there are two different threshold equations for the normal nodes and advanced nodes which are given as follows:

For normal nodes:

\[
T(n_{\text{normSEP}}) = \begin{cases} 
\frac{p_{\text{normSEP}}}{1-p_{\text{normSEP}}(r \mod \frac{1}{p_{\text{normSEP}}})} & \text{if } n_{\text{normSEP}} \in M^I \\
0 & \text{otherwise}
\end{cases} \tag{6.5}
\]

For advanced nodes:

\[
T(n_{\text{advSEP}}) = \begin{cases} 
\frac{p_{\text{advSEP}}}{1-p_{\text{advSEP}}(r \mod \frac{1}{p_{\text{advSEP}}})} & \text{if } n_{\text{advSEP}} \in M^{III} \\
0 & \text{otherwise}
\end{cases} \tag{6.6}
\]

In [90], the fuzzy system has been used to elect cluster heads. The algorithm uses three fuzzy variables (Energy, centrality and concentration of a node) for selection of CHs. In this approach, the essential information are collected by the BS and then according to the rules of fuzzy logic system a node is elected as a CH. In [99] also the CHs are
elected based on fuzzy logic rules. The fuzzy set contains the energy levels of nodes as well as their distances. In [99] a random number for each node is generated, and if the generated number is less than a specific threshold, the node might be discarded from becoming CH although it could be a qualified one. Authors in [129] use three linguistic variables to elect nodes as CH with attention on nodes having high probability, as well as high initial and final energy.

6.3 SYSTEM MODEL

Few basic assumptions are made about the sensor nodes, which are deployed to monitor an environment. These assumptions are as follows:

- All SNs have limited energy resources and not rechargeable.
- All nodes are homogeneous.
- The base station is static and has no mobility.
- Position of all the SNs are fixed.
- The BS has unlimited power.
- The distance of SNs from BS is computed based on the strength of signals.

6.4 PROPOSED SYSTEM MODEL

The fuzzy logic has been employed to enhance the cluster head selection and prolong the lifetime of the SEP protocol. In the proposed model the most popular inference system, has been deployed, which is known as Mamdani fuzzy inference system. It lets us to define how the system is working in a straightforward and simplified mathematical way. The Fuzzy Logic system for the proposed protocol is illustrated in Figure 1. Three input functions namely battery level, distance and node density were used to convert the crisp inputs into fuzzy sets. Each one of the input functions possesses three membership functions. Different membership function is used to illustrate the different level of the input functions as shown in Table 6.1 as well as Figure 6.1, Figure 6.2 and Figure 6.3.
Figure 6.1. Fuzzy Inference System BOKHARI-SEPFL

Table 6.1: Input Functions

<table>
<thead>
<tr>
<th>INPUT</th>
<th>Membership</th>
</tr>
</thead>
<tbody>
<tr>
<td>Battery Level</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td>High</td>
</tr>
<tr>
<td>Distance</td>
<td>Near</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td>Far</td>
</tr>
<tr>
<td>Node Density</td>
<td>Sparsely</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td>Densely</td>
</tr>
</tbody>
</table>

Figure 6.2. Membership functions of Battery Level

Figure 6.3. Membership functions of Distance
The fuzzy system output shows one of the probability level (Another probability will be computed using traditional threshold) for each node to be chosen as a cluster head. The output function of our proposed approach which contains 9 membership functions is shown in table 6.2 and figure 6.4.

**Table 6.2: Output Functions**

<table>
<thead>
<tr>
<th>OUTPUT</th>
<th>Membership Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probability</td>
<td>Very small , Small , Rather small ,</td>
</tr>
<tr>
<td></td>
<td>Medium small , Medium , Medium</td>
</tr>
<tr>
<td></td>
<td>large ,</td>
</tr>
<tr>
<td></td>
<td>Little strong , Strong , Very Strong</td>
</tr>
</tbody>
</table>

**Figure 6.4. Membership functions of Node Density**

**Figure 6.5. Membership functions of Qualification in FLS**
To elect nodes as CHs most efficiently in order to prolong the lifetime of the network we first compute the battery level, node density and distance to the base station, for each and every SN and then calculating the probability by considering fuzzy rule base given in Table 6.3. Then the computed probabilities for each node is stored in some variables.

<table>
<thead>
<tr>
<th>Battery</th>
<th>Density</th>
<th>Distance to BS</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>Sparsely</td>
<td>Far</td>
<td>Very small</td>
</tr>
<tr>
<td>Low</td>
<td>Sparsely</td>
<td>Medium</td>
<td>Small</td>
</tr>
<tr>
<td>Low</td>
<td>Sparsely</td>
<td>Near</td>
<td>Rather small</td>
</tr>
<tr>
<td>Low</td>
<td>Medium</td>
<td>Far</td>
<td>Small</td>
</tr>
<tr>
<td>Low</td>
<td>Medium</td>
<td>Near</td>
<td>Rather small</td>
</tr>
<tr>
<td>Low</td>
<td>Densely</td>
<td>Far</td>
<td>Rather Small</td>
</tr>
<tr>
<td>Low</td>
<td>Densely</td>
<td>Medium</td>
<td>Small Medium</td>
</tr>
<tr>
<td>Low</td>
<td>Densely</td>
<td>Near</td>
<td>Medium</td>
</tr>
<tr>
<td>Medium</td>
<td>Sparsely</td>
<td>Far</td>
<td>Rather Small</td>
</tr>
<tr>
<td>Medium</td>
<td>Sparsely</td>
<td>Medium</td>
<td>Small Medium</td>
</tr>
<tr>
<td>Medium</td>
<td>Sparsely</td>
<td>Near</td>
<td>Medium</td>
</tr>
<tr>
<td>Medium</td>
<td>Medium</td>
<td>Far</td>
<td>Small Medium</td>
</tr>
<tr>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>Medium</td>
<td>Medium</td>
<td>Near</td>
<td>Medium Large</td>
</tr>
<tr>
<td>Medium</td>
<td>Densely</td>
<td>Far</td>
<td>Medium</td>
</tr>
<tr>
<td>Medium</td>
<td>Densely</td>
<td>Medium</td>
<td>Medium Large</td>
</tr>
<tr>
<td>Medium</td>
<td>Densely</td>
<td>Near</td>
<td>Little Strong</td>
</tr>
<tr>
<td>High</td>
<td>Sparsely</td>
<td>Far</td>
<td>Medium</td>
</tr>
<tr>
<td>High</td>
<td>Sparsely</td>
<td>Medium</td>
<td>Medium Large</td>
</tr>
<tr>
<td>High</td>
<td>Sparsely</td>
<td>Near</td>
<td>Little Strong</td>
</tr>
<tr>
<td>High</td>
<td>Medium</td>
<td>Far</td>
<td>Medium Large</td>
</tr>
<tr>
<td>High</td>
<td>Medium</td>
<td>Medium</td>
<td>Little Strong</td>
</tr>
<tr>
<td>High</td>
<td>Medium</td>
<td>Near</td>
<td>Strong</td>
</tr>
<tr>
<td>High</td>
<td>Densely</td>
<td>Far</td>
<td>Little Strong</td>
</tr>
</tbody>
</table>
The next step would be to compute the threshold value for each node (Equation 6, 7) and then storing them in separate variables. The two variant probabilities obtained from our fuzzy logic system and the threshold equations will be used to find the most suitable nodes as CHs. We finally calculate the weighted probability by considering the mean of the previous probabilities for each and every node and then sorting them according to their value, nodes having height weighted probabilities would be elected as the CHs. As we can see here we used the old threshold equations as well as considering other properties of each node (battery level, node density and distance to the base station) to elect the most suitable nodes as CHs in order to prolong the network lifetime and increase its efficiency. The working procedure of the protocol is given below:

- **Step 1.** Calculating the battery level, distance and node density for each and every node
- **Step 2.** Computing the first probability, by entering the computed values (step 1) into the fuzzy inference system
- **Step 3.** Computing the second probability for each node using old threshold values. (Using equation 6 and 7)
- **Step 4.** Using the mean value of the obtained probabilities of each node and compute the weighted probability for each node
- **Step 5.** Sorting the weighted probabilities
- **Step 6.** Election of nodes having highest weighted probability as the CHs
- **Step 7.** Repeat steps 1 to 6, for each cluster till the maximum number of cycle is reached.

The detailed flow chart of the proposed protocol is given below in figure 6.5:
6.5 SIMULATION AND RESULTS

In order to evaluate the performance of the BOKHARI-SEPFL and compare it with SEP and its existing enhanced version SEP-E, simulation tests have been carried out in
MATLAB (2014a) [127]. 100 nodes have been deployed and distributed randomly within an area of 100*100 meter. Energy model used here is the same as in SEP protocol. A list of simulation parameters are given in table 6.4. In case of SEP protocol 20% of the total nodes are advanced nodes having 100% more energy than the normal ones and for SEP-E protocol 20% of the nodes deployed are advanced with 100% more energy than normal nodes, 10% nodes are intermediate nodes with 50% more energy than normal nodes.

**Table 6.4: Simulation Parameters**

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Environment Size</td>
<td>100 x100</td>
</tr>
<tr>
<td>2</td>
<td>Number of nodes</td>
<td>100</td>
</tr>
<tr>
<td>3</td>
<td>Packet Size</td>
<td>4000 bits</td>
</tr>
<tr>
<td>4</td>
<td>Election Probability value of Cluster-heads (P)</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>Initial energy per node (E₀)</td>
<td>0.5 J</td>
</tr>
<tr>
<td>6</td>
<td>E_{elec} = E_{bit}</td>
<td>50 nJ/bit</td>
</tr>
<tr>
<td>7</td>
<td>E_{fs}</td>
<td>10 pJ/bit/ m²</td>
</tr>
<tr>
<td>8</td>
<td>E_{mp}</td>
<td>0.0013 pJ/bit/ m⁴</td>
</tr>
<tr>
<td>9</td>
<td>E_{DA}</td>
<td>5nJ/bit</td>
</tr>
</tbody>
</table>

**6.5.1 Lifetime of Network**

Figure 6.6 depicts the total of alive nodes with respects to the functioning of the network in different rounds. Considering Table 6.5, it is clearly observed that in the proposed algorithm the death of the first node as well as the last node have been prolonged by 18.57% and 73.2% respectively comparing with SEP protocol. Bearing in mind the result obtained from SEP-E routing protocol we notice that improvement for the first node death and the last death is 6.45% and 42.4% respectively. Therefore the proposed algorithm improved the overall lifetime of the network significantly.
Figure 6.7. No. of Alive Nodes vs. Rounds

Table 6.5. Comparison of algorithms results

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>First Death</th>
<th>Half Death</th>
<th>Last Death</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOKHARI-SEPFL</td>
<td>1194</td>
<td>2092</td>
<td>4027</td>
</tr>
<tr>
<td>SEP-E</td>
<td>1117</td>
<td>1703</td>
<td>2828</td>
</tr>
<tr>
<td>SEP</td>
<td>1007</td>
<td>1500</td>
<td>2325</td>
</tr>
</tbody>
</table>

6.5.2 Residual Energy of Network

Analyzing the Residual energy of network can be very handy to determine the energy efficiency of an algorithm. The Figure 6.7 shows the comparison of energy consumption of SEP, SEP-E and BOKHARI-SEPFL protocols. As we observe the curve of BOKHARI-SEPFL is smoother than the other compared protocols thus it has a better performance in terms of balancing and distributing of the overall energy.
6.5.3 Throughput of Network

By term throughput we mean the number of packets which have been sent to the base station/sink. Considering Figure 6.8 we notice that our proposed algorithm has a higher throughput, about 68.54% comparing with SEP and 33.5% while taking into account the result obtained from SEP-E routing protocol.

6.5.4 Overall Network performance

It is observed that the aforementioned clustering algorithms take the advantage of FL to elect the most appropriate CHs by considering some vital information such as the
energy level of nodes, their locations etc. Taking into account the number of dead nodes (Figure 4.1) and Packet sent to the BS (Figure 4.3) in each round, along with the overall energy dissipation of network (Figure 4.2) it is found that the energy consumption and overall performance of the network has been significantly improved.

6.6 CONCLUSION

Within this chapter the SEP protocol has been optimized, using the concept of fuzzy logic system. This is done by balancing the energy load of the network. Nodes with high residual energy, high probability, and requiring less energy for communicating with other nodes as well as the BS are elected as the CHs. The new proposed algorithm BOKHARI-SEPFL shows more stability and runs for a longer time. The lifetime of the network has been extended by 73.2% and 42.4%, in comparison with the SEP and SEP-E protocols respectively. The result also declares the throughput of the network to be improved by 68.54% and 33.4% compared with the mentioned protocols.
CHAPTER SEVEN
BOKHARI-GM
ALGORITHM TO LOCATE
THE SINK NODE IN WSNs
7.1 INTRODUCTION

As we previously mentioned, WSNs due to variety of applications and future potential [125], [130-132] has gained a tremendous attention among the researchers. One on the main focusing area in this filed is to prolong the network by reducing the energy and balancing the overall power in the network. Two main components of a WSN are the sensor nodes and the sink node. Despite the fact that wireless sensor networks (WSNs) are capable of having a variety of topologies, for instance star, mesh or ring, the signals generated by the sensor nodes are provided to the end users through the sink nodes. A sink node or a base station is basically a designated device similar to the normal sensor nodes but more powerful. One of the primary tasks of the sink node is to bridge a WSN with the remote users (figure 7.1). Actually being not the same as ad hoc networks, sensor nodes in WSNs are powered by non-rechargeable batteries. Therefore the techniques and design of new protocol to prolong the lifetime of the network are of great importance. The energy required to route a message to the sink node, for each and every sensor node depends on the distance from the sink node and number hops that message will have to travel. Having several sink nodes, employed effectively within the network filed would help to reduce the energy needed for a message to be delivered and prolong the network lifetime. Though, there are some constraints of employing several sink nodes such as the cost of the device or not being practical to have more than one within the filed. Due to the fact that, the sensed data, collected by the ordinary
SNs are transferred to the sink node, the overall network performance can be influenced by the place of the sink node.

Figure 7.1. A typical Wireless Networks

There are several challenges to be faced in order to locate the optimal spot of the sink node within the network filed. Few of main issues are as follows:

- There exist a huge solution space which means the sink node can be possibly located at anywhere in the network filed.
- Massive number of sensor nodes in the WSNs is another main challenge in locating the sink node.
- There are different routing protocols each having its own energy model and technique to optimize and route data towards the sink node.
- Possible changes in the network topology due to any sort of failure or improvements which might require the sink node to be relocated.
- Optimization of sink node location for different sampling mode such as periodical or event-driven, might require different considerations [5].
- Increment of sensor nodes within the network filed require the sink node to be repositioned in order to improve the lifetime as well as the throughput of the network.

In this chapter a novel algorithm is proposed which intends to find out an optimal location for the sink node so that the sum of distances from all the nodes to the sink
node is minimized. To spot the optimal place our algorithm finds the geometric median of all the locations associated with the sensor nodes. In a discrete set of points, the geometric median could be defined as the location which basically minimizes the sum of distances to all the points. Despite of being a straightforward concept, its computation is a challenge.

### 7.2 RELATED WORK

In [133] the sink has been located on different places and the conclusion indicates that the centre of the network as well as the centre of the quarter having the highest density of nodes are far better choices for the sink location. In [134] the P-median, a well-known NP-hard problem was used to decide the optimal location of the sink node. The result given in [135] shows the optimal placement of the sink would be the centre. Authors in [136] fix the sinks location by taking into account the nodes whose data are conveyed through a node close to the sink. In [137] optimal base-station locations regarding two-tiered WSNs has been proposed. The network lifetime was evaluated by the distances of all the nodes and the sink as well as the average rate of bit stream. In [138] the result from the simulation shows that improvements on data rate and power Efficiency can be accomplished by employing different algorithms to discover a layout for the base station. The sink node position was selected to increase the joint weight of data flows in an effort to reduce the energy consumption of the WSN. In [139] the P-Median Problem model was utilized in order to express the problem of placing several sink nodes by the help of an iterative algorithm. In this chapter a novel algorithm is proposed which attempts to determine an optimal location for the sink node with the intention that sum of distances from all the nodes to the sink node is minimized. In order to spot the ideal place, the geometric median of all the locations associated with the sensor nodes is found.

### 7.3 BOKHARI-GM ALGORITHM STRATEGY TO PLACE THE SINK NODE

Basically if there is a discrete set of points located in a Euclidean Space the geometric median would be defined as the point which actually minimizes the sum of distances to all the points. In statics, geometric median is a significant method to calculate an estimate of a location. Additionally it is a typical problem in facility location, in which
it deals with the issue of locating a facility in an effort to minimize the total cost of transportation [140].

For a set of \( m \) points \( a_1, a_2, \ldots, a_m \) where each \( a_i \in \mathbb{R}^n \), the geometric would be as:

\[
GM = \arg\min_{b \in \mathbb{R}^n} \sum_{i=1}^{m} \| a_i - b \|_2
\]

(7.1)

Where \( \arg\min \) means the argument \( b \) that will minimize the sum.

Although the geometric median’s appearing as a simple and straightforward concept to comprehend, its computation poses a challenge. There exist no exact formula nor any precise algorithm, only numerical approximations are practical towards the solution. However, it can be calculated by the help of an iterative procedure, where in each step the algorithm generates a more accurate result to the problem.

If \( b \) is distinct from the points, \( a_i \) then \( b \) is the geometric median iff it satisfies:

\[
0 = \sum_{i=1}^{m} \frac{a_i - b}{\| a_i - b \|}
\]

(7.2)

Generally, \( b \) would be the geometric median iff there exist vectors \( u_i \) so that:

\[
0 = \sum_{i=1}^{m} u_i
\]

(7.3)

Where for \( x_i \neq b \),

\[
u_i = \frac{a_i - b}{\| a_i - b \|}
\]

(7.4)

And for \( x_i = b \),

\[\| u_i \| \leq 1\]

(7.5)

BOKHARI-GM Algorithm finds the geometric median of all the locations associated with the sensor nodes by first approximating the best location \((x, y)\) which is the centre of gravity. Then the sum of distances from the input points (assumed to be locations of sensor nodes) are computed to the point \((x, y)\). In the next step, the four neighbouring points of \((x, y)\) which are away an experimental distance \( e \) in every directions (up, down, right, left) are found and the sum of distances to each of them is calculated. If any of
the points give a better result, then the \((x, y)\) will be updated and the same procedure with the same value of \(e\) will be carried out. In case if none of the points improves the present value of the \((x, y)\) then the same procedure will be carried out but the value of \(e\) will be reduced as we need to look closer to the current point to find the optimal location for the sink node.

The following algorithm shows BOKHARI-GM approach to find the geometric median which is used to locate the sink node.

1. Reading all the points
2. Calculating the centre of gravity \((x, y)\) and use it as the first approximation to the answer
3. Computing sum of distances to \((x, y)\)
4. Finding the neighbouring points (up, down, right, left) of \((x, y)\), each far away by an experimental value \((e)\)
5. Computing the sum of distances to the neighbours of \((x, y)\)
6. Comparing the results with the \((x, y)\)
7. If none of neighbour improves the solution look closer to the current point (reducing the value of experimental value \(e\)) and go to step (4)
   Else
   Update the value of the \((x, y)\) and go to step (4)
8. Continue till obtaining a precise approximation of the location

### 7.4 Required Energy for Sink Node Placement

Minimizing the overall energy consumption of the WSN is the core objective of our proposed work. The proposed scheme needs no sensor nodes, which are energy constrained to be involved in the main process of computation. Generally the optimal location of the base station is calculated at the deployment stage of the WSN (the same was carried out in this experimental work). However in case of several changes which might take place after the deployment stage, such as changing the number of sensor nodes or their location, optimal position of sink node has to be recalculated. The recalculation will be done by the sink node itself, (assuming to have unlimited power) once recognizing any alterations in the number or position of the sensor nodes. The information regarding the positions of GPS-enabled sensor nodes which are mainly used for routing purposes [141,142] are sent to the base station. Then the base station
(if it is required) will recalculate its own ideal position which can be applied within the network filed accordingly.

7.5 IMPLEMENTATION

In our experiment the sink node each time has been placed in a different location in order to evaluate the performance of the WSN. The following basic characteristics are assumed to simplify the WSN model:

- All SNs have limited energy resources and not rechargeable.
- All nodes are homogeneous.
- The base station is static and has no mobility.
- Position of all the SNs are fixed.
- The base station has unlimited power and aware of the location of all SNs

100 homogeneous sensor nodes have been employed in the simulation and each node has an initial energy of 0.5 Joule, which have been spread over a 100x100 m field, depicted in figure.7.2. To evaluate the performance of the WSN the LEACH [73] which is one of the earliest and well-known hierarchical routing protocols have been engaged. MATLAB [126] is used to carry out the simulation tests. The various parameters used in the simulation are given in the following table.7.1 and the various locations for the base station are presented in the table 7.2.

![Figure 7.2. Arrangement of sensor nodes within filed](image-url)
Table 7.1. Various parameters used in the simulation

<table>
<thead>
<tr>
<th>Sr.</th>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Routing Protocols</td>
<td>LEACH</td>
</tr>
<tr>
<td>2</td>
<td>Base Station position</td>
<td>Divergent</td>
</tr>
<tr>
<td>2</td>
<td>Environment Size</td>
<td>100 x100</td>
</tr>
<tr>
<td>3</td>
<td>Number of nodes</td>
<td>100</td>
</tr>
<tr>
<td>4</td>
<td>Packet Size</td>
<td>4000 bits</td>
</tr>
<tr>
<td>5</td>
<td>Speed of EM wave</td>
<td>3 x 10^8 m/s</td>
</tr>
<tr>
<td>6</td>
<td>Election Probability value of</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>Cluster-heads (P)</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Number of rounds (r_max)</td>
<td>4000 rounds</td>
</tr>
<tr>
<td>8</td>
<td>Initial energy per node (E_0)</td>
<td>0.5 J</td>
</tr>
<tr>
<td>9</td>
<td>E_{elec} = E_{bit}</td>
<td>50 nJ / bit</td>
</tr>
<tr>
<td>10</td>
<td>E_{fs}</td>
<td>0.0013 pJ /</td>
</tr>
<tr>
<td>11</td>
<td>E_{amp}</td>
<td>100 pJ/bit</td>
</tr>
<tr>
<td>12</td>
<td>E_{DA}</td>
<td>5nJ / bit</td>
</tr>
</tbody>
</table>

Table 7.2. Various locations for the base station

<table>
<thead>
<tr>
<th>Sink node</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td>(22.62,51.06)</td>
</tr>
<tr>
<td>Centre</td>
<td>(50,50)</td>
</tr>
<tr>
<td>Random Location (1)</td>
<td>(20,60)</td>
</tr>
<tr>
<td>Random Location (2)</td>
<td>(30,45)</td>
</tr>
<tr>
<td>Random Location (3)</td>
<td>(50,20)</td>
</tr>
</tbody>
</table>

7.6 PERFORMANCE EVALUATION

The number of dead nodes versus number of rounds are illustrated in figure 7.3 and 7.4 as well. By locating the base station at the centre of the network filed (50,50), it is observed that the first node dies on 1384th round and half of the nodes died on 1830th round which gives a better result comparing with the rest of the locations. However considering the round that all the nodes die and the WSN functions no more, it is
realized that the proposed location (22.62, 51.06) offers a better result in which the last node of the WSN dies on 3670\textsuperscript{th} round.

![Figure 7.3](image)

\textbf{Figure 7.3.} Number of Nodes Dead vs. Number of rounds

![Figure 7.4](image)

\textbf{Figure 7.4.} Nodes death Vs. Number of Rounds

Figures 7.5 and 7.6 represent the energy dissipation of the network. Considering the figure7.6 which is a closer look obtained from the figure 7.6, we realize that the proposed sink location (22.62, 51.06) offers a better energy management and balancing after the second halve of the rounds (2100\textsuperscript{th} round) which result the WSN to function for a longer time comparing with the other locations. However the location of the sink
at the centre (50, 50) and the random location 2 (30, 45) gives a better result at the first halve of the rounds.

**Figure 7.5.** Energy Dissipation of WSN

**Figure 7.6.** Closer view of Energy Dissipation of WSN

The figure 7.7 points out that the number of packets received at base station located at the centre (50,50) as well as the random location 2 (30, 45) are slightly higher than the proposed location(22.62, 51.06).
CONCLUSION

The location of the Base Station has a significant effect on the energy dissipation and lifetime of the WSNs. An optimal location for the base station has been investigated within this chapter, in such a way that the sum of distances from all the sensor nodes to the base station is minimized. In an effort to place the sink node within the network BOKHARI-GM algorithm finds the geometric median of all the location associated with the sensor nodes. The optimal spot of the sink node found by our algorithm has been compared with various other options such as centre of WSN filed. Performance evaluation reveals that the proposed location for the sink node prolongs the network lifetime comparing with other possible location within the network filed.
CHAPTER EIGHT

CONCLUSIONS & FUTURE WORK
8.1 CONCLUSIONS

Wireless sensor networks due to variety of beneficial applications and future potential has gained a tremendous attention over the last decade among the researchers. These type of networks pose attention-grabbing and interesting challenges for networking research. Regardless of the goals and purposes of sensor applications, the main important job of the network would be to sense and accumulate data from a potential domain, processing and send the data back to particular sites. Achievement this mission competently requires the design of energy-efficient routing protocols to organize paths in between sensor nodes and the data sink in such a way to prolong network’s life as much as possible. The environment properties where sensor nodes are deployed, in addition to extreme resource and power limitation, make the design of routing protocols very challenging.

In chapter five of the thesis a novel two-step fuzzy logic based routing protocol has been introduced. In our proposed algorithm, the priorities of CHs were calculated by making use of fuzzy logic, with overall six fuzzy input parameters considered. In first-step of the proposed algorithm, the main factors of each SN such as residual energy, density and distance to BS are checked and according to the output, most suitable candidates are selected. Furthermore in the second-step of FL system the vulnerability index, centrality and distance between CHs are checked and according to the output the CHs will be elected. The well-known Mamdani fuzzy inference system was used in the proposed protocol that includes 27 set of rules in each step and in order to calculate the priority values the centroid defuzzification method was used. Theoretically there exist numerous algorithms based of fuzzy logic. But it was realized that they do not take into
account all the important information of the sensor nodes so as to guarantee the optimal selection of the CHs. Comparison of our proposed routing protocol against some other similar approaches using fuzzy logic such as CHEF [99] GUPTA [90] along with the earliest clustering algorithm LEACH [73] indicates that, the proposed algorithm performs better in case of fair distribution and balancing of the overall energy consumption. The result of simulation shows a smaller variance of energy for the proposed algorithm which also validates its reasonable balance and energy consumption in the network.

In chapter 6 of our research the concept of intelligent approach, fuzzy logic control is used to enhance the process of cluster head selection in the existing SEP [126] protocol based on three variables, distance of nodes form BS, density of nodes and the battery level of nodes to improve the lifetime of and throughput of the WSN. The proposed algorithm, Stable Election Protocol Using Fuzzy Logic (BOKHARI-SEPFL), shows more stability and runs for a longer time. The lifetime of the network has been extended by 73.2% and 42.4%, in comparison with the SEP and SEP-E protocols respectively. The result also declares the throughput of the network to be improved by 68.54% and 33.4% compared with the mentioned protocols.

In chapter 7 of the thesis a novel algorithm BOKHAR-GM is proposed which intends to find out an optimal location for the sink node so that the sum of distances from all the nodes to the sink node is minimized. Due to the fact that, the sensed data, collected by the ordinary SNs are transferred to the sink node, the overall network performance can be influenced by the place of the sink node. To spot the optimal place the geometric median of all the locations associated with the sensor nodes were found. Performance evaluation reveals that the proposed location for the sink node prolongs the network lifetime comparing with other possible location within the network filed.

**8.2 FUTURE WORK**

In future by running additional simulations, considering a large number of heterogeneous nodes where sensor nodes do not have the same characteristics such as initial energy, processing power etc. the efficiency of the proposed protocols can be further examined. The fuzzy set can be modified and adjusted more accurately to achieve a further improvement in network’s performance. Additional experiments with more than one sink node and larger WSNs, have to be performed as in reality wireless
sensor networks contain hundreds or thousands of sensor nodes. Alternative defuzzification approaches could be used to further evaluate the performance of the proposed routing protocols. Fuzzy logic control could be deployed to improve and investigate the optimal location of the sink node by considering the important information of all the sensor nodes. Finally our proposed protocols can be examined and evaluated against other hierarchal routing protocols such as APTEEN [76], Self-organized [77] and EWC [78].


[130] Jun Huang; Yu Meng; Xuehong Gong; Yanbing Liu; Qiang Duan, "A Novel Deployment Scheme for Green Internet of Things," in Internet of Things Journal, IEEE, vol. 1, no. 2, pp. 196-205, April 2014


[139] Dingding Luo; Hai Zhao; Yuanguo Bi; Kai Lin; Xiyuan Zhang; Zhenyu Yin; Peigang Sun, "Multiple Sink Nodes' Deployment Based on PMP in WSNs," in

