AN ENERGY EFFICIENT AND BALANCED CLUSTERING APPROACH FOR IMPROVING THROUGHPUT OF WIRELESS SENSOR NETWORKS

By
Amit Karmaker

MASTER OF SCIENCE

IN

INFORMATION AND COMMUNICATION TECHNOLOGY

INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY

The thesis titled "**AN ENERGY EFFICIENT AND BALANCED CLUSTERING APPROACH FOR IMPROVING THROUGHPUT OF WIRELESS SENSOR NETWORKS**" submitted by Amit Karmaker , Roll No. 0412312015, and Session April, 2012, has been accepted as satisfactory in partial fulfillment of the requirement for the degree of Master of Science in Information and Communication Technology on June 19, 2017.

BOARD OF EXAMINERS

1. **Dr. Mohammad Shah Alam** — **Chairman (Supervisor)**
   Associate Professor
   IICT, BUET, Dhaka

2. **Dr. Md. Saiful Islam** — **Member (Ex-Officio)**
   Professor and Director
   IICT, BUET, Dhaka

3. **Dr. Hossen Asiful Mustafa** — **Member**
   Assistant Professor
   IICT, BUET, Dhaka

4. **Dr. Raqibul Mostafa** — **Member (External)**
   Professor
   Department of EEE
   United International University, Dhaka

Declaration

It is hereby declared that this thesis and any part of it has not been submitted elsewhere for the award of any degree or diploma.

Signature of the Candidate

Amit Karmaker
0412312015
IICT, BUET

Dedication

THIS THESIS IS DEDICATED

TO

MY PARENTS AND MY NIECE

Contents

Declaration ii

Dedication iii

List of Tables vii

List of Figures ix

List of Abbreviation x

List of symbols xi

Acknowledgment xii

Abstract xiii

1 Introduction 1

1.1 Challenges in Wireless Sensor Networks 2

1.2 Motivation 4

1.3 Research Objectives 5

1.4 Outline of Methodology 5

1.5 Organization of Thesis 6

1.6 Summary 6

2 Literature Review 7

2.1 Key Factors of Energy Wastage in WSN 7

2.2 Energy Efficient Routing Protocols 8

2.3 Clustering in WSN 9

2.3.1 Clustering Characteristic 9

2.3.1.1 Cluster Properties 9

2.3.1.2 CH Properties 10
2.3.1.3 Clustering Process Properties 11
2.4 Related Work on Cluster Head Selection 12
2.4.1 Probabilistic Cluster Head Selection 13
2.4.1.1 Random CH Selection 13
2.4.1.2 Hybrid CH Selection 15
2.4.2 Deterministic Cluster Head Selection 19
2.4.2.1 Weight Based Clustering 20
2.5 Related Work on Intra-Cluster Communication 23
2.6 Summary 25

3 Proposed Balanced and Traffic Adaptive Clustering Protocol 26
3.1 Network Model and Assumptions 26
3.2 Radio Model 27
3.3 Protocol Description 28
3.3.1 Setup Phase 29
3.3.1.1 Cluster Head Selection 29
3.3.1.2 Cluster Formation 30
3.3.2 Steady State Phase 31
3.3.2.1 Capture Based Intra-Cluster Mechanism 31
3.3.2.2 Adaptive Contention Using Näive Bayes Classifier . . 35
3.4 Summary 37

4 Results and Performance Evaluation 38
4.1 Performance Metrics 38
4.2 Simulation Environment 39
4.3 Simulation Results with Various Metrics 40
4.3.1 FND, 10^{th} ND and LND with Regard to Number of Round . . 41
4.3.2 Number of Alive Nodes with Respect to Rounds 42
4.3.3 Number of Dead Nodes with Respect to Rounds 43
4.3.4 Remaining Energy of the Network Per Round 45
4.3.5 Energy Consumption of the Network Per Round 46
4.3.6 Number of Selected CH Per Round 46

4.3.7 Number of Packets Sent to Base Station per Round 47
4.4 Summary 49

5 Conclusion and Future Work 50
5.1 Conclusion 50
5.2 Future Work 51

Appendix A 58

Appendix B 73

List of Tables

2.1 Comparative analysis of related protocols 24

3.1 Summary of the used notation 34

3.2 A training set containing data rows 36

3.3 Training data 37

4.1 Network Parameters 39

4.2 Performance comparison of competitive protocols for 50 nodes 48

4.3 Performance comparison of competitive protocols for 75 nodes 49

List of Figures

1.1 General structure of wireless sensor network 1
1.2 Challenges in WSN 2

2.1 Classification of routing protocol in WSN 8
2.2 Working principle of cluster based protocol 9
2.3 The clustering characteristics of WSN (redrawn from) 12
2.4 Phases of clustering protocols 13
2.5 Marked four nodes can not select as cluster head due to number of neighbor less than l 23
2.6 Adaptive frame for intra cluster communication by srikanth et.al. . . 24

3.1 Network model of the proposed protocol 27
3.2 Energy consumption model for sending and receiving l bit packet . . 28
3.3 Overview of the proposed protocol 29
3.4 Communication between a CH and member node during setup phase 31
3.5 When current slot is at the left of the window 33
3.6 When current slot is at the right of the window 33

4.1 Simulated Network in OMNeT++ (a) Node deployment in OMNeT++ (b) Number of node selection for the simulation 40
4.2 Topology after the selection of cluster head in OMNeT++ 40
4.3 FND, 10^{th} ND and LND with respect to round (a) for 50 nodes (b) for 75 nodes 41
4.4 Number of alive nodes in the network with respect to round (a) for 50 nodes (b) for 75 nodes 42
4.5 Number of dead nodes in the network with respect to round (a) for 50 nodes (b) for 75 nodes 43

4.6 Remaining energy of the network with respect to round (a) for 50 nodes (b) for 75 nodes 44
4.7 Energy consumption of the network with respect to round (a) for 50 nodes (b) for 75 nodes 45
4.8 Number of selected CH with respect to round (a) for 50 nodes (b) for 75 nodes 47
4.9 Number of Transmitted packet to BS (a) for 50 nodes (b) for 75 nodes 48

List of Abbreviations

WSN	Wireless Sensor Network
CH	Cluster Head
IoT	Internet of Things
FND	First Node Death
LND	Last Node Death
MAC	Medium Access Control
TDMA	Time Division Multiple Access
CSMA	Carrier Sense Multiple Access
BS	Base Station
LEACH	Low Energy Adaptive Clustering Hierarchy
LEACH-C	Low Energy Adaptive Clustering Hierarchy - Centralized
LEACH-DCHS	LEACH - Distributed Cluster Head Selection
A-LEACH	Advanced - LEACH
I-LEACH	Improved - LEACH
EECS	Energy Efficient Clustering Approach
DB-LEACH	Distance Based - LEACH
DBEA-LEACH	Distance Based Energy Aware - LEACH
EECABN	Energy Efficient Clustering Algorithm Based on Neighbors
LEACH-MAC	LEACH- Medium Access Control
DWCA	Dynamic Weight based Clustering Algorithm
EUCRP	Energy-balanced Unequal Clustering Routing Protocol
CMRP	Cluster-based Multi-hop Routing Protocol
E.E.	Energy Efficiency
RSSI	Received Signal Strength Indicator
FAWAC	Fixed Assignment based Window Access with Capture

List of symbols

$NBR(n)$	Set of neighbor nodes of n
$NBRCNT$	Number of neighbor nodes of n
R	Transmission range
k	Optimum number of cluster head
w_{v_i}	Weighted value of node i
RE_i	Residual energy of node i
CH_{adv}	Cluster Head advertisement
$JOIN_{req}$	Join request for become cluster member
TS_{alloc}	Time slot allocation
N_c	Number of nodes in the cluster
W	Window size
OS	Owners slot
NOS	Non Owner's slot
S_{BW}	Slot behind the window
S_{AW}	Slot after the window
D_{bf}	Number of packet into the queue
Cs	Current Slot
Fs_t	Fixed Slot Transmission
Rs_t	Random Slot Transmission
ST	Successful Transmission
Th	Threshold
Rs_s	Random slot success
Rs_f	Random slot failure

Acknowledgment

All praises are for the Almighty for giving me the strength, without which I could not afford to attempt this research work.

I would like to express my gratitude to my honorable thesis supervisor Dr. Mohammad Shah Alam, Associate Professor, Institute of Information and Communication Technology (IICT), Bangladesh University of Engineering and Technology (BUET), who gave me feedback and the correct guidelines to keep going further in my research. Nothing is comparable to his keen advice and the freedom he provided for me in research. I am grateful to him for his cooperation throughout my thesis work. Additionally, Md. Mahedee Hasan who cheered me up during the long journey until my graduation.

I would like to thank all the members of the board of examiners for their precious time in understanding my work and their insightful comments. I would like to thank to all of my friends and colleagues for their cooperation. Last but not least, I am grateful to my parents, especially my mother. Without their trust and support, I would have never arrived here.

Abstract

Maximizing the lifespan of wireless sensor networks is presently drawing much attention in the research community. To reduce energy consumption, sensor nodes which is far away from the base station avoid sending data directly. As a result, several disjoint clusters are formed and nodes within the cluster send its data through cluster head to avoid long transmission. However, several parameters related to transmission cost need to be considered when selecting a cluster head. While most of the existing research work considers energy and distance as the most stringent parameter to reduce energy consumption, these approaches fail to create a balanced and fair cluster. Consequently, unbalanced cluster formation results in the degradation of overall performance. In this research work, a cluster head selection algorithm is proposed by considering residual energy, number of neighbor nodes, and one hop cluster head information which covers every region of the sensing area in a balanced manner as well as saves significant amount of energy. Furthermore, a capture effect based intra-cluster communication mechanism is proposed that efficiently utilizes the time slot under various traffic conditions. A Näive Bayes Classifier is used to adapt the window size dynamically according to the traffic pattern. Finally, a simulation model using OMNeT++ is developed to compare the performance of the proposed approach with the pioneer clustering approach, LEACH, centralized LEACH-MAC and contemporary DWCA protocol. Simulation results show that proposed approach improved overall performance in terms of energy efficiency, throughput, and network lifetime.

Chapter 1

Introduction

Wireless sensor network (WSN) is composed of large number of small, inexpensive and low power sensor nodes. This self configured sensor nodes are commonly used in military reconnaissance, vehicular movements, volcanic earthquake timing, weather forecasting, environment monitoring like temperature, humidity, pressure, motion, etc. [1–4]. Now-a-days wireless sensor networks is one of the key enablers of Internet of Things (IoT) for collecting surrounding context and environment. In most of the cases, wireless sensor nodes are placed in remote, inaccessible and hazardous area [5], which makes them very difficult to replace or recharge. The major challenges of the sensor nodes are battery power limitations, processing power constraints, duplicate data gathering, and limited memory power of the network.

A WSN comprises of spatially appropriate independent sensor nodes to conveniently sense physical or environmental conditions. These type of networks are fundamentally data collecting networks, where data are extremely associated with the end user [6, 7]. The deployed sensor nodes communicate wirelessly to the base

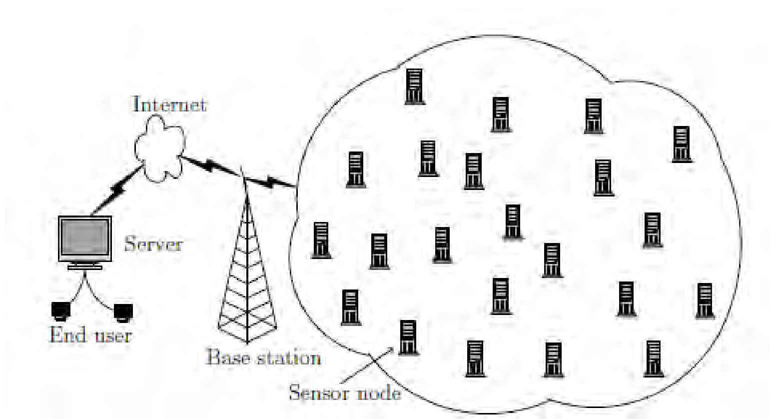


Figure 1.1: General structure of wireless sensor network [5]

station and often try to build a network. The general structure of a wireless sensor network is depicted in Figure 1.1. The WSN may comprise of hundreds or even more number of nodes, which provides reliable monitoring of any applications. The sensed data are transmitted to the base station directly or by multi hop fashion. The base station is connected to the wired world where the data can be collected in large databases for future use.

1.1 Challenges in Wireless Sensor Networks

Before formation of the sensor network and deployment of sensor nodes, the prior and fundamental understanding about connecting and managing the network is needed to achieve beneficial scalability and efficiency. Figure 1.2 shows some important challenges that the wireless sensor networks should overcome and they are described below.

Energy Efficiency

Power consumption is one of the crucial challenges required to be managed in sensor networks. Many researchers are focusing their efforts to improve energy efficiency in these networks. As many of the sensors are battery powered, energy consumption is a very crucial metric and should be managed wisely in order to extend the network lifetime. For example, in the forest fire detection based applications, it is difficult

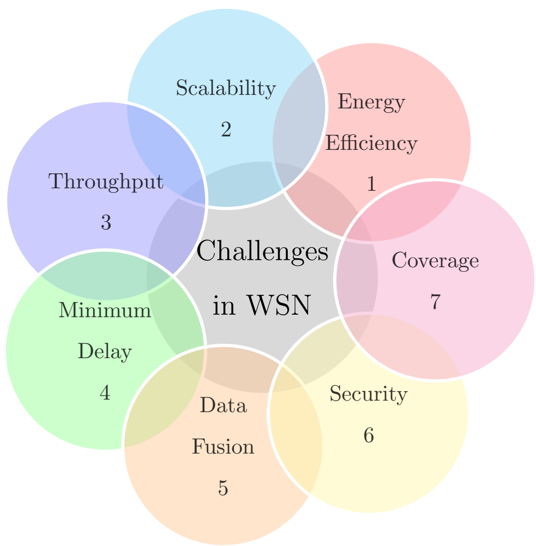


Figure 1.2: Challenges in WSN

to replace the batteries in the forest. Hence, the sensors may fail and might not function if the batteries are exhausted. So, efficient routing may overcome this issue and extend the network lifetime.

Scalability

Scalability is a critical factor especially for sensor networks which contain large number of nodes and can be responsible for degradation of network performance as well. Topological changes in network such as network size and node density should not affect the performance of the network. Hence, routing protocols employed in WSN must be scalable enough to maintain the sensor states when it changes its state from sleep to ideal or vice versa.

Throughput

Most of the times sensor must transmit its data to the BS; the required number of successful packet transmission of a given node per time is determined as throughput.

Minimum Delay

Acquiring the exact information without any distortions is the most primary objective in a WSNs. Also, there should not be any sort of delay. The routing protocols and network topology will ensure the delivery of the data with minimum delay.

Data Aggregation

Data aggregation is the combination of data arriving from different sources by using some functions such as suppression (finding and eliminating duplicates), minimum, maximum and average [8]. As sensor node generates the meaningful data, data from multiple nodes can be aggregated in order to reduce the number of transmissions. This aggregation technique is used to reduce the energy consumption and achieve data transfer optimization in the routing protocols.

Security

As the routing protocols have limited capability, some of these protocols cannot accommodate all the crucial information acquired by the sensor, challenging the security of data. Data is sent to the end users by getting direct access to the messages present in the sensors through internet services. Hence, there is a need to prevent access to the data from unauthorized parties or from any malicious actions.

Coverage

The sensor nodes are deployed randomly based on the required application. Another way of deployment is self organizing systems, where the sensor nodes are scattered and topology is formed in an ad-hoc manner. Uniform distribution of CHs throughout the network is an influenced factor [9]. Arranging cluster head too far or too close from each other can create inefficient energy protocols [10].

1.2 Motivation

Clustering is one of the important routing techniques for wireless sensor network. In clustering, the network is divided into clusters, each of which has one Cluster Head (CH) that is responsible for collecting data from its member nodes. A CH collects information from its members, then aggregates and finally forwards the aggregated information to the Base Station(BS). Clustering is an efficient routing technique that can extend network lifetime which is discussed in details in Chapter 2. However, there are many challenges that still exist in clustering approach and need to be solved. Through the review of some existing cluster based algorithms, a number of issues have been observed which motivated this research work to be carried out. These issues are mainly concerned with balancing load and energy consumption among nodes in a WSN. Selecting CHs is one of the critical decisions in clustered networks since CHs have more responsibilities and functions than member nodes. To form clusters, one of the main concerns is how to distribute the CH over the network in a balanced manner. Due to the nature of wireless sensor network, nodes within the network have various traffic condition. So, using TDMA in communication phase achieves low channel utilization under low contention. Moreover, most of the clustering protocols does not consider node buffer. That's why, it is difficult to find the data transmission behavior as compared with real sensor network scenario. Besides, idle listening by the CH under low traffic may shorten the lifespan of CH. So, in order to design a cluster based algorithm, there are three main aspects that need to be considered, namely, optimum CH selection, balanced cluster formation and efficient cluster communications. Therefore, designing an energy efficient clustering approach by harmonizing the above mentioned three aspects is a challenging issue.

1.3 Research Objectives

The goal of this research work is to improve the network lifetime and maximize the throughput of cluster based wireless sensor network. In order to meet these goals, the following objectives have been identified:

- To develop a new cluster head selection algorithm for Wireless Sensor Networks (WSN) to save significant amount of energy and to cover every region of the sensing area in a balanced manner.
- To devise an intra-cluster communication technique to achieve high throughput and low latency.
- To develop a mechanism for dynamic window size to adapt various traffic loads.
- To compare the performance of the proposed scheme with existing clustering methods.

1.4 Outline of Methodology

The methodology consists of the following stages:

- At first, a new Cluster Head (CH) selection algorithm for WSN will be designed by considering residual energy, number of neighbor nodes and one hop neighbor information.
- Then, a new intra cluster communication technique will be introduced which exploits the benefit of capture effect in order to maximize the utilization of time frame within a cluster.
- In order to adapt various traffic loads, Naïve Bayes Classifier algorithm will be used for dynamic selection of window size for the next frame.

- Then, overall design will be simulated using OMNeT++ simulation framework.
- Finally, performance analysis of the proposed clustering protocol will be carried out through simulation results. Performance improvement in terms of network lifetime, number of CH and throughput will be investigated and compared with existing protocols.

1.5 Organization of Thesis

The subsequent parts of the thesis are organized as follows:

Chapter 2: Literature Review

This chapter gives an overview of routing techniques of WSN and integrates the key research efforts that are available in this field. Specifically, this chapter describes the shortcomings of existing state-of-art clustering techniques and their communication methodologies.

Chapter 3: Balanced and Traffic Adaptive Clustering Protocol

This chapter presents, in detail, the proposed clustering approach. Cluster Head selection algorithm for the proposed protocol is discussed. Later, capture based intra cluster communication mechanism followed by Näive Bayes classifier for dynamic window size is explained.

Chapter 4: Simulation Results

This chapter presents the performance analysis of the proposed clustering protocol through simulation results. Performance improvement in terms of network lifetime, number of CH and throughput is investigated and compared with existing protocols.

Chapter 5: Conclusion and Future Work

This chapter concludes the thesis by summarizing the main ideas and some directions for future research.

1.6 Summary

This chapter presents a very brief introduction as well as the challenges of WSN. Motivation and research methodology are also discussed here to get an overview of the outcome of this research work. Finally, organization of the thesis is described.

Chapter 2

Literature Review

In wireless sensor network, sensor nodes deployed randomly with limited battery life. For the energy efficient transmission of sensed data from sender to receiver, selection of routing technique is one of the main factors. Designing a routing protocol is full of challenges, mainly due to limited power, low bandwidth, low computational power, no conventional addressing scheme, computational overheads and self-organization of the sensor nodes.

2.1 Key Factors of Energy Wastage in WSN

The most challenging concern in WSN design is how to save node energy while maintaining the desirable network behavior. According to Soua et al. [11], there are several source of energy wastage :

- *Collision*: when a node receives more than one packet at the same time, these packets collide. All packets that cause the collision have to be discarded and the retransmission of these packets is required.
- *Overhearing*: when a sender transmits a packet, all nodes in its transmission range receive this packet even if they are not the intended destination. Thus, energy is wasted when a node receives packets that are destined to other nodes.
- *Control packet overhead*: a minimal number of control packets should be used to enable data transmissions.
- *Idle listening*: is one of the major sources of energy dissipation. It happens when a node is listening to an idle channel in order to receive possible traffic.

- *Interference*: is the other major issue in low power and lossy network. It occurs when unwanted signal interfere the original signal.

2.2 Energy Efficient Routing Protocols

Pantazis et al. [12] categorize the routing protocols of WSN into four broad categories:

- Network Structure Scheme
- Communication Model Scheme
- Topology Based Scheme
- Reliable Routing Scheme

Later, Network structure scheme is further divided into Flat and Hierarchical protocol as shown in Figure 2.1. In Flat architecture, all nodes in the network plays identical role. This type of routing is useful for small area network due to minimum overhead to maintain the network. But, in large scale network the main problem is scalability. Whereas, in hierarchical architecture the network is divided into various clusters to achieve energy efficiency, stability, and scalability. This chapter briefly describe the working principle of hierarchical protocol followed by existing literature review.

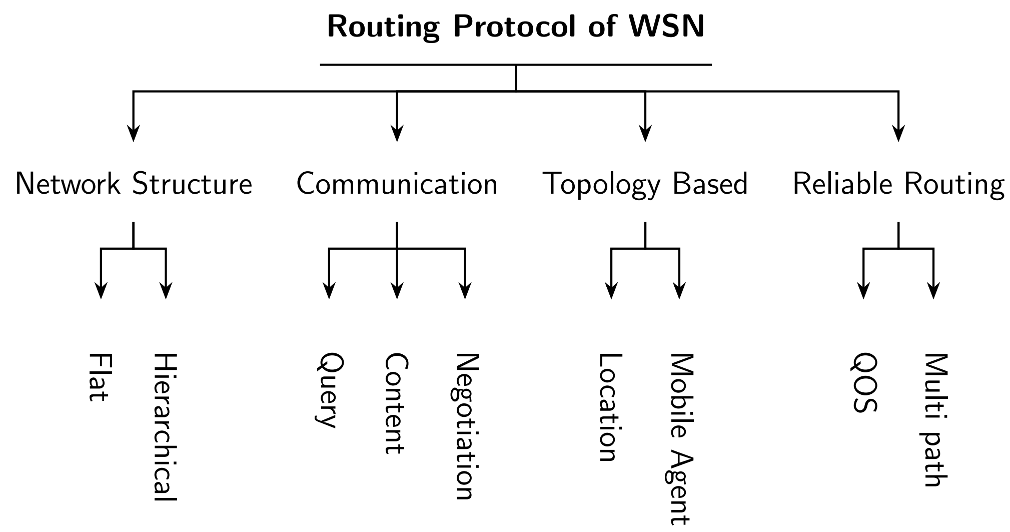


Figure 2.1: Classification of routing protocol in WSN (redrawn from [12])

2.3 Clustering in WSN

Cluster based routing protocol is a cross layer solution which attracts great attention of the researchers in the last few years. Most of the clustering protocol works into two phases: setup phase and steady state phase as shown in Figure 2.2. In setup phase, network is divided into various clusters by selecting the cluster head according to centralized, distributed, or hybrid manner [13]. In steady state phase, member nodes in every clusters sends their sensed data to the cluster head node in their predefined TDMA slot assigned by the CH in the setup phase. After collecting all data, Cluster head nodes aggregate them and send to BS.

2.3.1 Clustering Characteristic

To classify different clustering approaches, various clustering characteristics are used. According to Arjunan and Sujatha [14], the three characteristics of clustering are discussed in this subsection.

- Cluster properties
- CH properties
- Clustering process properties

A proper taxonomy of clustering characteristics can be found in Figure 2.3.

2.3.1.1 Cluster Properties

The specifications of the cluster are defined as cluster properties which include: cluster size, cluster count, intra-cluster communication and inter-cluster communi-

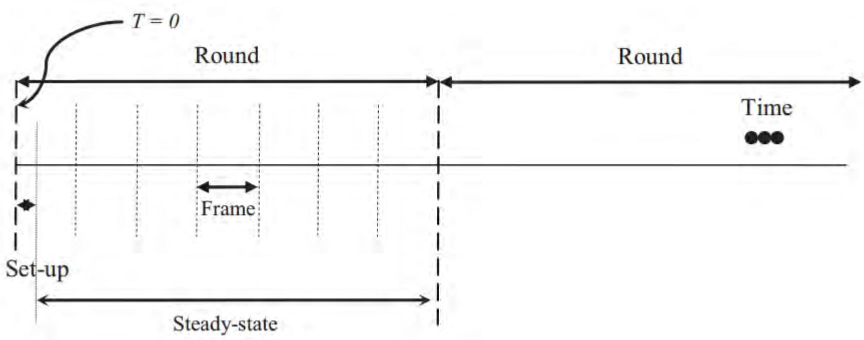


Figure 2.2: Working principle of cluster based protocol

cation.

Cluster count: The number of cluster formed is predefined or variable based on the application requirement. In some cases, the number of cluster is 5% of total number of the nodes deployed. In many applications, the number of clusters are variable when the CHs are randomly selected.

Cluster size: The cluster size can be classified into equal and unequal size cluster. In equal clustering, the region is divided into equal size clusters and the size of the cluster is same throughout the network [15]. In unequal clustering, the cluster size is determined based on the distance to BS. The cluster size is smaller when the distance to BS is small and the size increases as the distance to BS increases.

Intra-cluster communication: Intra – cluster communication involves the data transmission between CH and cluster member within a cluster. Based on the clustering approaches, the communication can be direct or multi-hop. For large scale WSN, multi-hop communication is needed for data transmission within a cluster. The communication mechanism can be contention based, schedule based, or hybrid.

Intra cluster communication: Inter – cluster communication can be direct or multi hop communication. Usually, multi-hop mechanism is preferred for energy efficient data transmissions from CHs to BS through intermediate CHs in large scale WSN. In some applications of small scale WSN, the communication between the CH and BS is single hop transmission.

2.3.1.2 CH Properties

CH performs the following operations: Collecting data from cluster members, aggregating data and forwarding data to BS through direct or multi-hop communication.

Mobility: The CHs can be either stationary or mobile. The mobile CHs can move for a limited distance, although the topology management process of mobile CHs is more difficult than in a network with stationary CHs. For this type of node, velocity and positioning is the crucial parameter for cluster design [16].

Node type: The dispersed CHs across the network can be rich in resources compared to the regular nodes; that is, the network supports the heterogeneity of the nodes. Or, the network can be homogeneous and the CHs are picked from the regular nodes.

Role: The CH receives data from its cluster members, perform data aggregation of the collected sensor data.

2.3.1.3 Clustering Process Properties

The characteristics of Clustering process are listed below:

Clustering methods: There are two methods of clustering: centralized and distributed. In centralized approaches, a central authority like BS or super nodes controls the entire operation (cluster formation, CH selection, etc.,) while distributed approaches have no central authority and widely employed in large scale WSN.

Dynamism: A clustering approach can be either dynamic or static. In dynamic approaches, the CHs are elected based on the current conditions of the network and most of dynamic approaches act in a real times scheme. In static approaches, the CH election and related operations are performed regardless of the current network conditions.

Nature: The clustering process can be proactive, reactive or hybrid in nature. The node continuously senses the data and forwards it to CH. In proactive type, the CH transmits the data to BS continuously. In reactive type, CH transmits the data whenever the sensed value crosses the predefined threshold. In hybrid cases, CH transmits the data to BS at longer regular time intervals and also when the value crosses the threshold value.

CH selection: There are two ways to select CH in WSN: probabilistic methods and deterministic methods. In probabilistic approaches, CHs are selected randomly without any previous consideration. The probabilistic approach is further divided into random and hybrid manner. In deterministic based method, various metrics are used to select CHs like residual energy, node degree, node centrality, expected residual energy, distance to BS, etc. Deterministic approach is further divided into weight, fuzzy and heuristic based method.

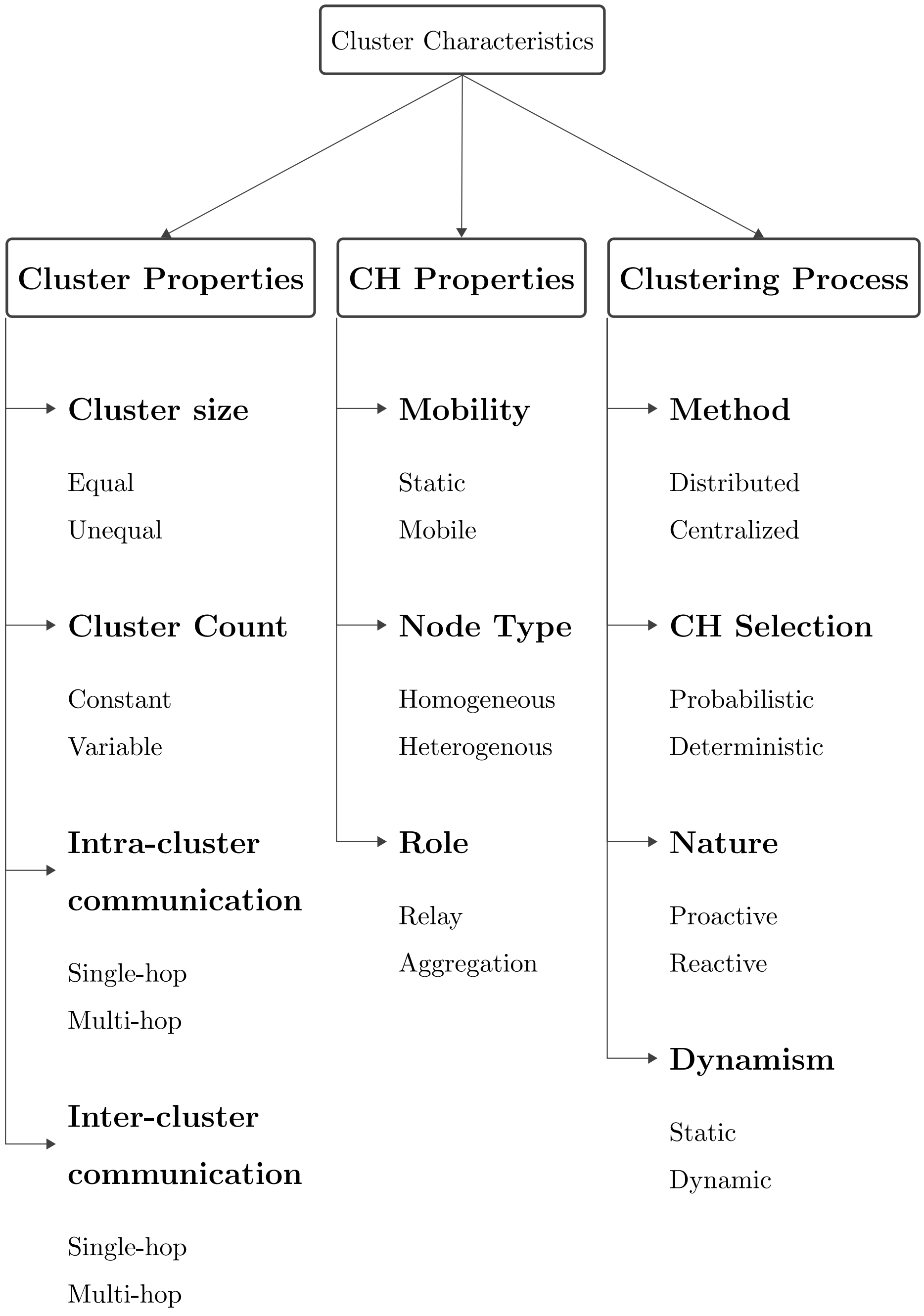


Figure 2.3: The clustering characteristics of WSN (redrawn from [17])

2.4 Related Work on Cluster Head Selection

This section discuss some notable cluster head selection strategy from various literature. According to Fig. 3.1, CH selection divided into deterministic and proba-

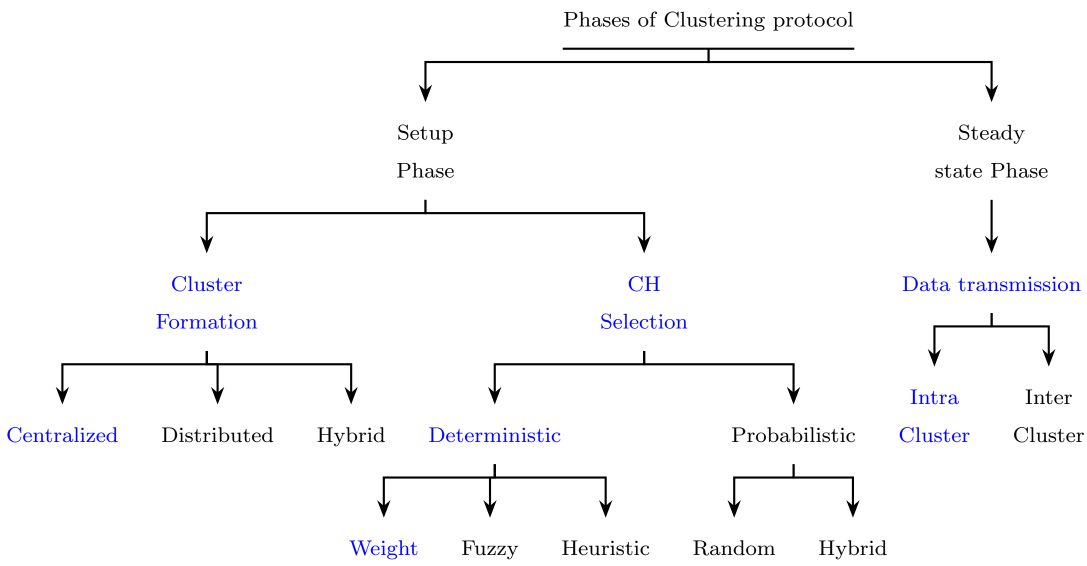


Figure 2.4: Phases of clustering protocols

bilistic approach. So, this section briefly discuss the CH selection strategy in both categories.

2.4.1 Probabilistic Cluster Head Selection

The primary objective in probabilistic clustering algorithms is to prolong the network lifetime as much as possible. Some of these algorithms aimed at randomly selecting the heads. This group conserves the simplicity and produces a minimum overhead for clustering the nodes. In order to create clustering protocol more efficient, the overhead of clustering, including the message and time, should be small. This overhead is incurred because the nodes need the local information to be able to organize themselves into clusters. On the other hand, others utilize some helpful metrics to achieve more goals in addition to the increased network lifetime, including reduced routing delay and fault-tolerance.

2.4.1.1 Random CH Selection

Low Energy Adaptive Clustering Hierarchy (LEACH) [18] is a pioneer clustering routing protocol for WSN. The operation of LEACH consists of several rounds where each round is divided into two phases as discussed earlier. During the setup phase, CH selection, cluster formation and assignment of a Time Division Multiple Access (TDMA) schedule by the CH for member nodes are performed. In CH selection, each node participates in a CH election process by generating a random priority value between 0 and 1.

If the generated random number of a sensor node is less than a threshold value $T(n)$, then the node becomes CH. The value of $T(n)$ is calculated using Equation 2.1.

$$T(n) = \begin{cases} \dfrac{p}{1 - p \times (r \; mod \; \frac{1}{p})}, & if \; n \in G \\ 0, & Otherwise \end{cases} \tag{2.1}$$

Where, p denotes the desired percentage of sensor nodes to become CHs among all sensor nodes, r denotes the current round and G is the set of sensor nodes that have not participated in CH election in previous $1/p$ rounds. A node that becomes the CH in round r cannot participate in the next $1/p$ rounds. In this way, every node gets equal chance to become the CH and energy dissipation among the sensor nodes is distributed uniformly. Once a node is selected as the CH, it broadcasts an advertisement message to all other nodes. Depending on the received signal strength of the advertisement message, sensor nodes decide to join a CH for the current round and send a join message to this CH. By generating a new advertisement message based on Equation 2.1, CHs rotate in each round in order to evenly distribute the energy load in the sensor nodes. After the formation of the cluster, each CH creates a TDMA schedule and transmits these schedules to their members within the cluster. The TDMA schedule avoids the collision of data sent by member nodes and permits the member nodes to go into sleep mode. The setup phase is completed if every sensor node knows its TDMA schedule. The steady state phase follows the setup phase. In the steady state phase, transmission of sensed data from member nodes to the CH and CH to the BS are performed using the TDMA schedule. Member nodes send data to the CH only during their allocated time slot. When any one member node sends data to the CH during its allocated time slot, another member node of that cluster remains in the sleep state. This property of LEACH reduces intra cluster collision and energy dissipation which increases the battery life of all member nodes. Additionally, CHs aggregate data received from their cluster members and send it directly to the BS. The CH senses the states of the channel for sending its data. Concept of clustering used by LEACH protocol enforces less communication between sensor nodes and the BS, which increases the network lifetime. But, there exist some disadvantages in LEACH which are as follows:

- In each round, the CH is chosen randomly and the probability of becoming the CH is the same for each sensor node. After completion of some rounds, the probability of sensor nodes with high energy as well as low energy becoming the CH is the same. If the sensor node with less energy is chosen as the CH, then it dies quickly. Therefore, robustness of the network is affected and lifetime of the network degrades.

- LEACH does not guarantee the position and number of CHs in each round. Formation of clusters in basic LEACH is random and leads to unequal distribution of clusters in the network. Further, in some clusters the position of the CH may be in the middle of the clusters, and in some clusters the position of the CH may be near the boundaries of the clusters. As a result, intra cluster communication in such a scenario leads to higher energy dissipation and decreases the overall performance of the sensor network.

In addition to distributed clustering in LEACH, a centralized clustering approach **LEACH - Centralized (LEACH-C)** [19] is proposed by the same authors of LEACH. In LEACH-C, the BS is responsible for cluster formation. At the beginning, each node sends its information including its location and energy level to the BS. The BS then computes the average of the node energies and the nodes with energy below this average are not selected as CH for the current round. When the BS selects the CHs for the current round, it broadcasts the node ID of the CHs to all the nodes in the entire network. The nodes that are not selected as CH join the nearest CH.

2.4.1.2 Hybrid CH Selection

LEACH - Distributed Cluster Head Selection (LEACH-DCHS) [20] is proposed by Handy et al. for prolonging the network lifetime. This is achieved by making two modifications in LEACH protocol. Firstly, they modify the threshold $T(n)$ value for CH selection by multiplying the remaining energy factor shown in Equation

2.2 and then using a new approach to define the network lifetime. LEACH-DCHS presents a hybrid cluster-head selection algorithm with low energy consumption.

$$T(n) = \begin{cases} \dfrac{p}{1 - p \times (r \ mod \ \dfrac{1}{p})} \times \dfrac{En_{current}}{E_{max}}, & if \ n \ \epsilon \ G \\ 0, & Otherwise \end{cases} \tag{2.2}$$

Here, $En_{current}$ is the current energy and E_{max} is initial energy of the node n. But, the problem of this modification is that after a certain numbers of rounds the network gets stuck, although nodes with sufficient energy are available.

In **Advanced - LEACH (A-LEACH)** [21], a new technique for CH selection in every round is proposed. The technique for selection of CH depends on two terms: current state probability (CS_p) and general probability (G_p). Thus, the threshold value to become a CH depends on both terms in each round. So, the threshold value $T(n)$ can be calculated by Equation 2.3 and the value of G_p and CS_p are obtained from Equations 2.4 and 2.5 respectively.

$$T(n) = G_p + CS_p \tag{2.3}$$

here,

$$G_p = \frac{K}{N - k \times (r \ mod \dfrac{N}{K})} \tag{2.4}$$

and,

$$CS_p = \frac{E_{current}}{E_{n-max}} \times \frac{N}{K} \tag{2.5}$$

Where, $E_{current}$ is current energy of a node, E_{n-max} is the initial energy of the network, K is expected number of CHs in a round, r is current round and N is the total number of nodes in the sensor network. So, after putting the value of G_p and CS_p in Equation 2.3, the final threshold value will be represented by Equation 2.6.

$$T(n) = \frac{K}{N - k \times (r \ mod \dfrac{N}{K})} + \frac{E_{current}}{E_{n-max}} \times \frac{N}{K} \tag{2.6}$$

Optimum Cluster - LEACH (K-LEACH) [22] proposed by Thein and Thein, adopted the optimum number of clusters (K_{opt}) to make a dot-product over a traditional LEACH protocol. The optimum number of clusters was derived that gave the factor of coverage area, node density, the effects of transmission energy consumed in both free-space (fs) and multipath (mp) scenarios as well as the distance to the BS (d_{toBS})

$$T(n) = \begin{cases} \dfrac{p}{1 - p \times (r \ mod \ \dfrac{1}{p})} \times \dfrac{E_{current}}{E_{initial}} \times K_{opt}, & if \ n \ \epsilon \ G \\ 0, & Otherwise \end{cases} \tag{2.7}$$

where,

$$K_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}} \frac{\sqrt{\epsilon_{fs}}}{\sqrt{\epsilon_{mp}}} \frac{M}{d_{toBS}^2}$$

Improved - LEACH (I-LEACH) [23] protocol suggested a new idea for selection of the CH. The CH in I-LEACH protocol is selected by considering residual energy, the number of neighboring nodes and position of the node from the BS. All sensor nodes generate a random number between 0 and 1 like LEACH in each round. The improved threshold $T(n)$ has been derived as shown in Equation 2.8 . Comparing the randomly generated number to $T(n)$, if the number is less than $T(n)$, that node will become the CH for the current round, otherwise it remains in normal node.

$$T(n) = \begin{cases} \dfrac{p}{1 - p \times (r \ mod \ \dfrac{1}{p})} \times \dfrac{E_c}{E_{avg}} \times \dfrac{NBR_n}{NBR_{avg}} \times \dfrac{d_{to}BS_{avg}}{d_{to}BS_n}, & if \ n \ \epsilon \ G \\ 0, & Otherwise \end{cases} \tag{2.8}$$

Where, E_c is the current energy of a sensor node and E_{avg} represents average energy of the network. NBR_n and NBR_{avg} are the number of neighbors for n and the average number of neighboring nodes in the network respectively. $d_{to}BS_{avg}$ and $d_{to}BS_n$ denote the average distance of sensor nodes to the BS and distance of individual sensor nodes from the BS respectively. The main disadvantages of this protocol is non-uniform distribution of CHs, which increases the total energy dissipated in the network.

A competition based clustering approach is proposed in EECS [24]. In this scheme, to select the CHs, each node becomes a candidate node based on a probability and announces its status to all its neighbor nodes with in a competition range. Each candidate node, after waiting to receive the announcement from other competition nodes,checks for if there is a candidate with a greater residual energy. If there is one, the node leaves the competition, otherwise, the candidate elects itself as new CH. EECS produces more control overhead complexity because all nodes have to compete with each other for becoming CHs.

Nguyen et al. [25] proposed **Distance Based - LEACH (DB-LEACH)** and **Distance Based Energy Aware - LEACH (DBEA-LEACH)**. In DB-LEACH, a node is more likely to be selected as a cluster head if the distance of it from the BS is nearly equal to the average distance of the network sensor nodes to the BS. In CH nodes selection phase of DBLEACH algorithm, each sensor node generates a random number between 0 and 1. Then, the random number is compared with improved threshold obtained from Equation 2.9.

$$T(n) = \begin{cases} \dfrac{p}{1 - p \times (r \ mod \ \dfrac{1}{p})} \times \dfrac{|d_{toBSavg} - d(i, BS)|}{d_{toBSavg}}, & if \ n \ \epsilon \ G \\ 0, & Otherwise \end{cases} \tag{2.9}$$

Where,

$$d_{toBSavg} = \frac{\sum_1^N d(i, BS)}{N}$$

where, $d(i, BS)$ is distance of sensor node i from the BS, $d_{toBSavg}$ denotes the average distance of the network sensor nodes to the BS, N is total number of sensor nodes. This protocol assumes that the distance from the sensor nodes to the BS and the parameter $d_{toBSavg}$ have been calculated before the network starts operation.

In DBEA-LEACH, in order to select the appropriate CH nodes in the CH nodes selection phase, DBEA-LEACH algorithm takes important factors such as position of the sensor node relative to the BS and the amount of residual energy of each sensor node. Similar to DB-LEACH, DBEA-LEACH establishes a new threshold based on distance. In addition, it introduces current energy and initial energy of the node in Equation 2.10 to CH election probability so as to ensure the nodes with higher remaining energy have greater probability to become CHs than those with the low remaining energy. The CH nodes selection directly affects the performance factors of WSN such as load distribution, energy efficiency, and network lifetime.

$$T(n) = \begin{cases} \dfrac{p}{1 - p \times (r \ mod \ \dfrac{1}{p})} \times \dfrac{|d_{toBSavg} - d(i, BS)|}{d_{toBSavg}} \times \dfrac{E_i}{E_{init}}, & if \ n \ \epsilon \ G \\ 0, & Otherwise \end{cases} \tag{2.10}$$

2.4.2 Deterministic Cluster Head Selection

In contrast with probabilistic approaches, deterministic approaches use standard metrics for selecting CHs. The commonly used traditional metrics are residual energy, node degree, expected residual energy, distance to BS, node centrality, etc., which are attained locally. This information is usually updated by exchanging message between its neighbors. This approach is called deterministic approaches because the clusters with elected CHs are more controllable. This is further classified into three categories: Weight based, Fuzzy based and heuristic based clustering algorithm. In weight based approach, a weight is calculated at each node based on some metrics such as residual energy, node degree, distance to BS, etc. The node with maximum weight is elected as a cluster head. Fuzzy logic is used to elect CHs in situations where uncertainties are more. The cluster head is chosen based on fuzzy input parameters. The input parameters can be residual energy, node degree, distance to BS, node centrality, etc., and the output fuzzy parameters are cluster size and probability of becoming CHs. In the recent years, heuristic based clustering algorithms provide optimal solution in the process of selecting CHs and cluster size. Many optimization algorithms such as Genetic Algorithm (GA), Ant Colony Optimization (ACO), Artificial Bee Colony Optimization (ABC), Particle Swarm Optimization (PSO), Bacterial Foliage Algorithm (BFA), Differential Evolution (DE), etc., are used in WSN. Each algorithm defines various metrics in fitness function to achieve better performance. Heuristic approaches are centralized and a central authority like BS controls all operations in the network. In exceptional cases, some approaches works in distributed manner using agent nodes. Compound algorithm uses different metric like connected graph in clustering methods.

2.4.2.1 Weight Based Clustering

Energy Efficient Clustering Algorithm Based on Neighbors(EECABN): Zhou et al. [26] introduced a centralized clustering approach based on neighbors status. In the method, a combined weight for electing the CHs is introduced which is composed of the following factors: the distance between the node and the BS, the distance between the node and its neighboring nodes within communication range R, and the residual energy of the node. Accordingly, the weight is defined as

$$W(i) = E(i)\frac{h_{max}}{max(h(i), \epsilon)} \sum_{j \in B} \frac{1 - \dfrac{d(i,j)}{R}}{max(\dfrac{E(j)}{E_{max}}, \epsilon)} \tag{2.11}$$

where in node i, $E(i)$, $h(i)$ and R are the residual energy, the distance to the BS, and the communication range in which other nodes can communicate to i successfully with enough signal level, respectively, B is the set of neighbors for i, j is one of the neighbors within R, $d(i,j)$ is the distance between i and j, the variable h_{max} is the distance between the BS and the farthest node in the network, E_{max} is the initial energy, and ϵ is a constant to limit the lower bound of $E(j)/E_{max}$ or $h(i)$, while a node has a very low residual energy or is too close to the BS. In the method, the nodes are divided into strong, with an energy higher than the average energy of all the nodes in the network (E_{ave}), and weak, with an energy smaller than E_{ave}. The CHs are elected from the strong nodes as follows. The BS collects the information of the nodes, computes the weights and selects the nodes with the highest weight as the CHs. Isolated CHs are the CHs with no members.

Weighted - LEACH (W-LEACH) is a new data aggregation algorithm presented by Abdulsalem et al. [27] for WSNs that can handle uniform and non uniform networks. They have assigned a weight w_i based on residual energy e_i and the density d_i to each sensor S_i. The d_i is the ratio between all alive nodes in the range r of a sensor node S_i with all alive nodes in the network. The w_i can be calculated using Equation 2.12.

$$W_i = \begin{cases} e_i \times d_i : & if\ d_i > d_{thres} \\ d_i : & Otherwise \end{cases} \tag{2.12}$$

Where, d_{thres} is a density threshold. Sensors with less than d_{thres} density are selected for the data transmission. So, all the nodes of a cluster not needed to activate and take part in each round of communication, like LEACH. In this way, authors increase the average lifetime of sensor nodes and enhance the network lifetime. Abdulsalam and Ali, have extended their work by introducing a dynamic W-LEACH using CH density d_{CH} . The d_{CH} is calculated using Equation 2.13.

$$d_{CH} = \frac{number\ of\ alive\ sensors\ in\ Cluster_i}{total\ number\ of\ alive\ sensors} \tag{2.13}$$

LEACH - Medium Access Control (LEACH-MAC) [28] protocol is designed to mitigate the randomness problem by restricting the number of cluster head advertisements. The optimal number of CHs k is calculated based on Equation used in K-LEACH. When the CH selection process starts, a variable CH_{heard} initializes to 0 and is incremented by 1 if it receives a CH advertisement message. In the threshold function, nodes select a uniform random time from the time interval 0 to total adv time, where total *advtime* is the time required for the CH transmission and reception. Suppose the selected time is R_t, so the CH advertisement sending time t_{advCH} can be calculated using Equation 2.14.

$$t_{advCH} = \frac{R_t}{Current\ Energy} \tag{2.14}$$

Now, node checks the value of CHheard variable at time t_{advCH} that it has updated at the time the advertisement was received. If the value of CHheard variable is less than the optimal number of clusters, then it will declare itself as a CH and sends a CH advertisement; otherwise it declares itself as a normal node. When calculating the optimal CHs number, LEACH-MAC assumes that CHs are distributed evenly, but the algorithm does not consider how to distribute CHs evenly, so actual CHs distribution is obviously different from the ideal, and the suitable CHs number is different from the calculated number [29, 30].

Energy-balanced Unequal Clustering Routing Protocol (EUCRP), proposed by wang et al. [31] also minimize the CH advertisement like LEACH-MAC. They first select temporary CH by considering residual energy by using threshold function. The candidate cluster heads calculate the waiting time T according to the

Equation 2.15 . After waiting time T, they broadcast message that they become the cluster head. The other candidate cluster heads in its competitive range drop out of the race.

$$T = \lambda \times T_{CH} \times \frac{E_0}{E_r} \times \frac{d_i}{d_{max}} \tag{2.15}$$

where, λ is random number between $0.9 \sim 1.0$, T_{CH} represent cluster head competition maximum time. The main drawback of this protocol is that, when nodes in a particular region have high residual energy, then nodes from other region cannot create energy efficient cluster.

Another weight based protocol **Dynamic Weight based Clustering Algorithm (DWCA)** is proposed by Essa. A et al [32] , where they used number of neighbor nodes, distance to BS and remaining energy for selecting candidate cluster head node. They used Equation 2.16 to calculate the weight value.

$$W_v = w_1 \times neighbors + w_2 \times distance\ to\ BS + w_3 \times remaining\ energy \tag{2.16}$$

Where, $w_1 = 0.2$, $w_2 = 0.2$ and $w_3 = 0.6$. Then, each node sends its weighted value to its neighbors. Those node who have highest value among the neighbors select itself as a candidate CH. Each initially selected candidate CH compete for the final selection of CH within its range (R_{comp}); among the range those node that have highest weighted value finally selected as CH. The main drawback of this protocol is that a lot of control packet exchange is needed for the selection of CH. Moreover, considering distance to BS as a weight metric with remaining energy and number of neighbor nodes may lead an unfair selection of CHs.

Cluster based Multi hop Routing Protocol (CMRP) [33], considers three conditions for selecting cluster head. One problem in this protocol is that if the number of neighbor is less than the optimum number of nodes, then it cannot select cluster head.

$$l = \frac{n - m}{m} \tag{2.17}$$

where l is the number of neighbor, n and m is the number of active node and optimum cluster respectively. Sensor nodes are deployed randomly in the region and the density of the nodes are not equal.

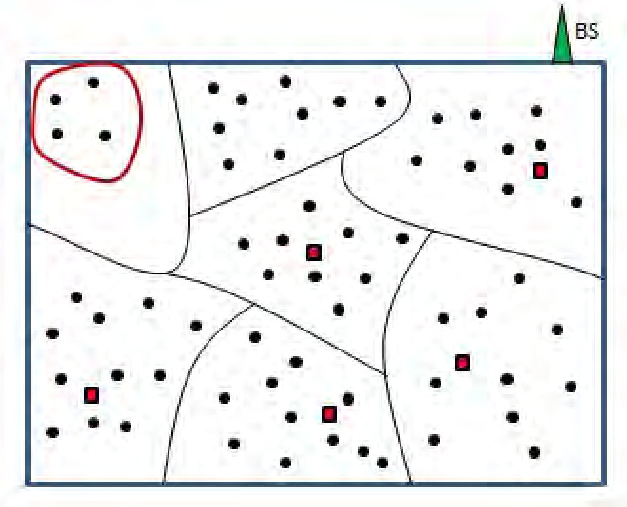


Figure 2.5: Marked four nodes can not select as cluster head due to number of neighbor less than l

According to Figure 2.5, there is a probability when some sensor nodes are separated from other nodes and the number of neighbor is less than l. Then, these separated nodes can't be selected as Cluster Head by the Base Station. In this scenario, separated nodes need to join one of the CH which is far away from the ordinary nodes. For a long distance communication, signal of the member node will get faded and it will also require significant amount of energy to transmit the data [34]. Some protocol [35, 36] use extra relay node to avoid long distance communication but selecting the relay node increase complexity.

2.5 Related Work on Intra-Cluster Communication

At the setup phase of LEACH protocol, CH sends the TDMA schedule to all of its member node. Member nodes sensed the environment and send the data to the CH during their allocated time slot only. A lot of successors of LEACH protocol work over setup phase but they consider steady state phase as same as traditional LEACH protocol. For high traffic, schedule based communication mechanism is efficient than the contention based protocol. But, in low or moderate traffic lots of slot unused in the intra-cluster communication phase. Some researchers work over data communication phase in clustering. A CSMA based intra-cluster communication mechanism is reported in [37] where a sender initiated communication mechanism is proposed within cluster to prolong the lifetime of CH. But, long preamble cre-

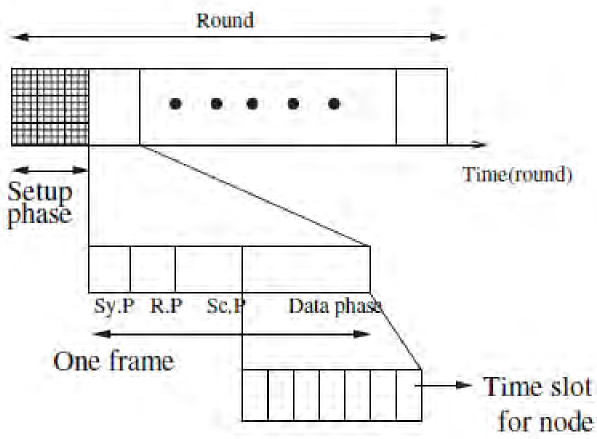


Figure 2.6: Adaptive frame for intra cluster communication by srikanth et.al.

ates extra control packet overhead and increase delay. Control message overhead is considered as one of the important part of cluster complexity for the performance metrics of the cluster based WSN [38]. Srikanth et al. [39] introduce a hybrid intra-cluster communication mechanism for mobile nodes where the CH first collect the requisition of packet from the member nodes. According to the packet request, CH broadcast the TDMA schedule among the nodes as shown in Figure 2.6. But it

Table 2.1: Comparative analysis of related protocols

Protocol	**Year**	**Clustering**	**Overhead**	**E.E**	**Scalability**	**Coverage**
LEACH	2000	Distributed	High	Mod.	Low	Unbalanced
LEACH-C	2002	Centralized	Low	High	Low	Unbalanced
LEACH-DCHS	2002	Distributed	High	High	Low	Unbalanced
A-LEACH	2008	Distributed	Mod.	High	High	Unbalanced
EECABN	2011	Centralized	Mod.	High	Low	Unbalanced
I-LEACH	2013	Distributed	High	High	Mod.	Balanced
W-LEACH	2013	Distributed	High	High	High	Unbalanced
DB-LEACH	2014	Distributed	High	High	Low	Unbalanced
CMRP	2015	Centralized	Mod.	High	Low	Unbalanced
LEACH-MAC	2016	Distributed	High	High	Mod.	Unbalanced
DWCA	2017	Distributed	High	Mod.	High	Balanced

increases extra control packet exchange. In addition with this, the authors did not mention the mechanism by which the member node sends their requisition. Because for high traffic condition, this requisition increase latency.

Table 2.1 gives a comparative analysis of the related protocols that already discussed in this chapter. Table 2.1 shows year wise comparison among LEACH and it's successors considering most important five parameters: clustering scheme, overhead, energy efficiency, scalability and coverage.

2.6 Summary

It has been observed from the literature study that quite a good number of clustering schemes have been proposed in the last fifteen years. However, the existing schemes are expensive from communication, energy, and time perspectives. Most of the existing protocols do not consider overall region of the network while selecting the CH. Also, the fixed schedule based communication mechanism did not gives the flexibility with different traffic condition. This provides a motivation for designing an energy efficient and balanced clustering algorithms that could reduce the overall network energy consumption and can increase throughput.

Chapter 3

Proposed Balanced and Traffic Adaptive Clustering Protocol

The goal of the proposed protocol is to make balanced cluster and to provide good performance in clustering with various traffic condition. With an unbalanced network, it is difficult to get the benefit from clustering advantages. So, it is necessary to select cluster head such way that it can form balanced cluster and can cover full region in fair manner.

This section will start with the basic assumptions and radio model used in this protocol and presents an overview of the proposed protocol.

3.1 Network Model and Assumptions

The network is represented by an undirected graph, $G = (V, B, E)$, where the set of vertices, V, represents the nodes, and E is the set of edges, $(u, v) \epsilon E$ if both u and v are in the communication range of each other. B denotes the base station that is supposed to be energy unlimited node. The set of nodes to which a node, v_i, is directly connected (v_i's neighbors), is denoted by N_{vi}. This model represented by G reflects a network topology when nodes use a regular transmission range. However, nodes are supposed to be able to communicate directly with the BS by increasing their power when needed (to initially transmitting energy information or when selected as CHs). The network is assumed stationary after deployment and nodes are unaware of their location. Traffic is periodic, with all nodes generating data packets of the same length and at a fixed rate. Data aggregation is used by CHs, i.e., a CH compresses all the data received from its members into a single message and then transmits it to the BS. A clustering Scheme S is defined as a set of clusters, $S = (CH_i, m_1, m_2, m_j)$, that covers the network (the graph G). Every

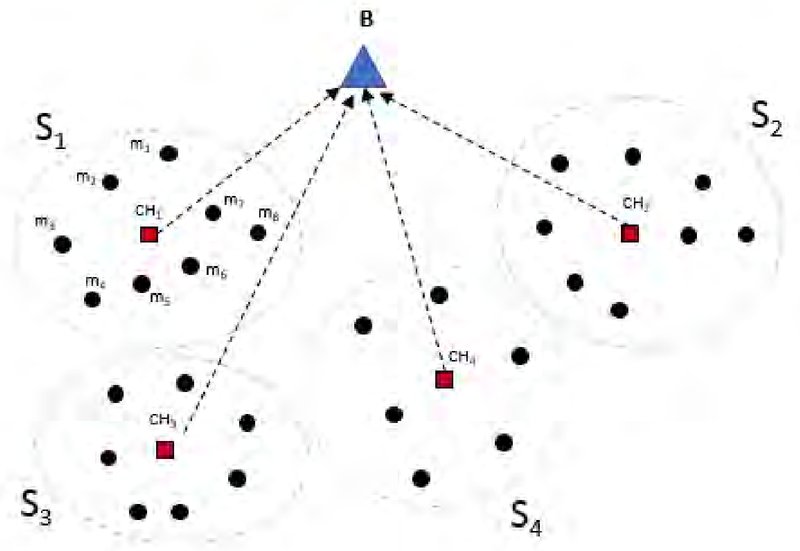


Figure 3.1: Network model of the proposed protocol

tuple (CH_i, m_1, m_2, m_j) represents a cluster i with a cluster-head CH_i, and j members (m_1, m_2, m_j), as shown in Figure 3.1. In this protocol, the following key characteristics are assumed for the network:

- The network is homogeneous, where sensor nodes have same initial energy.
- Nodes are randomly deployed in the network and has unique ID.
- Each sensor node is location unaware and can control the transmission range.
- Base station is located outside the network and have enough computation power.
- Each sensor node have enough power to communicate with BS.

3.2 Radio Model

For energy consumption analysis, the following first order radio model [23] is used which is depicted in Figure 3.2.

The energy consumption of transmitting l bit over distance d is given by the following equation:

$$E_{TX(l,d)} = \begin{cases} E_{elec} \times l + \epsilon_{fs} \times l \times d^2, \ for\ d < d_0 \\ E_{elec} \times l + \epsilon_{mp} \times l \times d^4, \ for\ d \geq d_0 \end{cases} \tag{3.1}$$

Here, E_{elec} is energy required for processing per bit, ϵ_{fs} and ϵ_{mp} is the energy taken for transmitting one bit to achieve an acceptable bit error rate in free space and multipath model respectively. d is the distance between a sensor node and its respective cluster head or between CH and BS.

Where threshold d_0 is calculated by the following equation:

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \tag{3.2}$$

and the energy consumption by receiving l bit is given by

$$E_{RX} = E_{elec} \times l \tag{3.3}$$

where, E_{RX} is the energy dissipated per bit at receiver.

3.3 Protocol Description

A general overview of the proposed protocol is depicted in Figure 3.3. In case of designing clustering protocol, the first step towards the development is to select the cluster heads. Other than CH, remaining nodes join the nearest CH based on distance in the cluster formation phase. In the steady state phase, each frame is divided into TDMA slot, ACK slot and decision of window size for next frame slot.

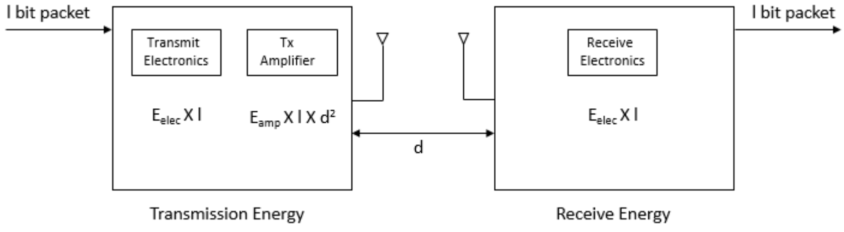


Figure 3.2: Energy consumption model for sending and receiving l bit packet

3.3.1 Setup Phase

Cluster formation can be generally classified into distributed and centralized techniques . In the distributed approach, the decisions related to the formation of the clusters are made by the sensor nodes without the use of a central entity (or BS). In centralized clustering, the control message for cluster computation is received from a central base station (BS) based on information collected from all the sensors in the network [40]. This research work focus on the centralized cluster head selection scheme. Here, base station selects CH node according to the parameter value. BS has enough resource such as energy, computation, and memory. So, with an efficient algorithm, it can select cluster head in fair and balanced way.

3.3.1.1 Cluster Head Selection

When nodes are randomly deployed into the network, it first selects its radius to find its neighbors by Equation 3.4

$$R = \sqrt{\frac{M \times M}{\pi \times k}} \tag{3.4}$$

Where M×M is the area of the network and k is the optimum cluster. By using Equation 3.5 optimum number of clusters can be found [19].

$$k = \frac{\sqrt{N}}{\sqrt{2 \times \pi}} \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \frac{M}{d^2_{toBS}} \tag{3.5}$$

Where N is the number of sensor nodes and d_{toBS} is the average distance from node to base station.

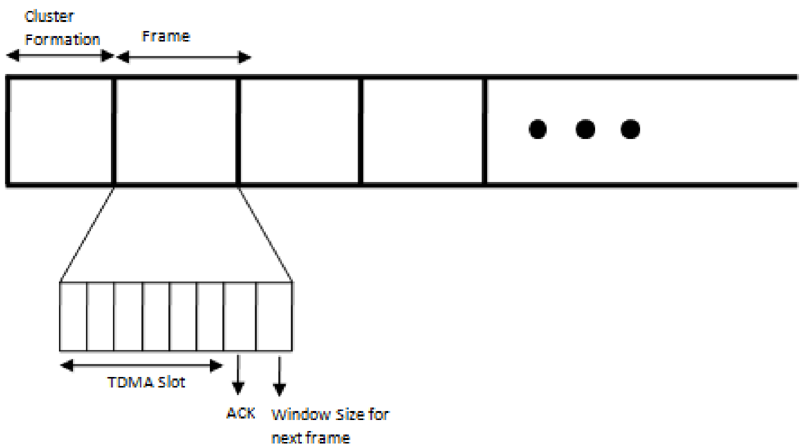


Figure 3.3: Overview of the proposed protocol

After calculating its radius each node sends 'HELLO' message to find its neighbor and set $NBR(n)$ and $NBRCNT(n)$, where $NBR(n)$ is the set of all neighbor of n and $NBRCNT(n)$ is the number of neighbor nodes of n.
All nodes then send their neighbor information and Residual Energy (RE) information to the Base Station to sort the weighted value (w_{v_i}) which is calculated by the following equation:

$$w_{v_i} = RE_{v_i} + \alpha \times NBRCNT_{v_i} \tag{3.6}$$

To incorporate residual energy with the number of neighbor nodes, a constant α, which value is taken as 0.1 into the analysis. After sorting w_{v_i}, the node which has higher weight is selected by the base station as a cluster head and added into CH list. When base station go through the second highest node, it first checks its neighbor list with the selected CH list. If one of its neighbors is already in the selected CH list, then BS considers it as a normal node; otherwise, it is added to CH list.

By sampling properties of each node and checking its neighbor node, finally base station finds a balanced set of cluster head and then, informs the determined CH node. Algorithm 1 gives the detailed procedure of cluster head selection mechanism.

3.3.1.2 Cluster Formation

Those nodes that are already selected as CH by BS broadcast Cluster head advertisement (CH_{adv}) within its radius R. Member nodes receive CH_{adv} message from its neighbor CH nodes. After collecting all CH_{adv} message, member nodes select nearest CH nodes via RSSI value. The distance between two nodes can be estimated based on the received signal[41].

Member node sends $JOIN_{req}$ message to the nearest CH by performing CSMA/CA. CH collects $JOIN_{req}$ from all of its neighbor. Then, it allocate (TS_{alloc}) and announce time slot to all requesting nodes. Member nodes receive TS_{alloc} and both CH, and member nodes enter into steady state phase. The detailed flow diagram of cluster formation is shown in Figure 3.4.

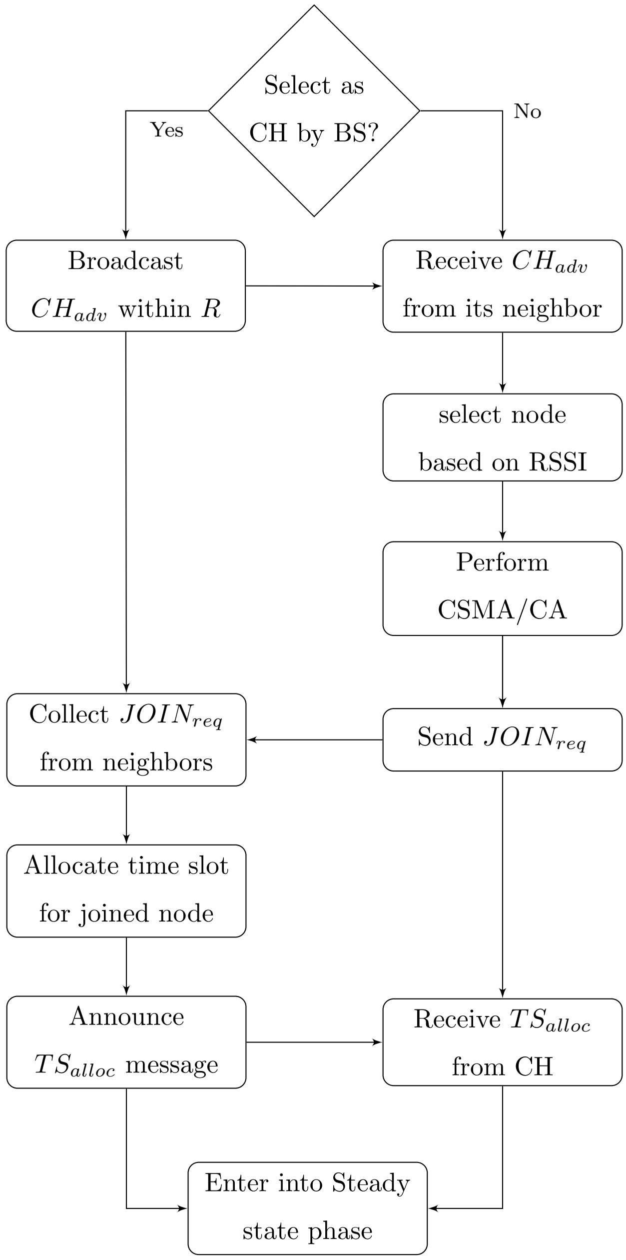


Figure 3.4: Communication between a CH and member node during setup phase

3.3.2 Steady State Phase

In this section, an intra-cluster communication mechanism without using extra control packet is introduced which adapt the slot as per traffic condition.

3.3.2.1 Capture Based Intra-Cluster Mechanism

The basic idea of capture effect is, when multiple sender transmits packet simultaneously, then the receiver can decode the comparatively high power packet. The term

Algorithm 1 CH selection algorithm

$NBRCNT = 0$
$NBR = \emptyset$
for $\forall V \in G$ **do**
    Calculate R by using Equation 3.4
    Sends *hello* message within R
    Update NBR & $NBRCNT$
end for
$CH = \phi$ ▷ Initially set of CH is empty
for (each vertex $v_i \in V$) **do**
    calculate its weight w_{v_i} by using E.q. 3.6
end for
SORT (w_{v_1}) ▷ sort the weighted value in descending order
for i=1 to N **do**
    if $NBR_{v_i} \cap$ CH is ϕ **then**
        $CH \leftarrow v_i \cup CH$
    else
        v_i is a normal node
    end if
end for

capture effect was first used in 1976 by Metzener et al. [42]. Capture effect first exploited in WSN after 29 years of capture based ALOHA network by Whitehouse et al. [43]. Unlike FAWAC, as presented in [44], the advantages of the capture effect by considering the buffer size of each node is exploited and adapted the window size as per previous round status. The proposed mechanism consider the window size as same as FAWAC, then it adapt the window size as per traffic pattern. The basic idea of capture effect is, when multiple sender transmits packet simultaneously, then the receiver can decode a single packet whose power level is higher than the other packet. Within any cluster, the member node has its own slot for sending their sensed data. In the proposed mechanism, the owner of each slot always sends data with high power and non-owner node tries to send data with low power outside the window. If there is no traffic for the owner, then non-owner nodes have the pos-

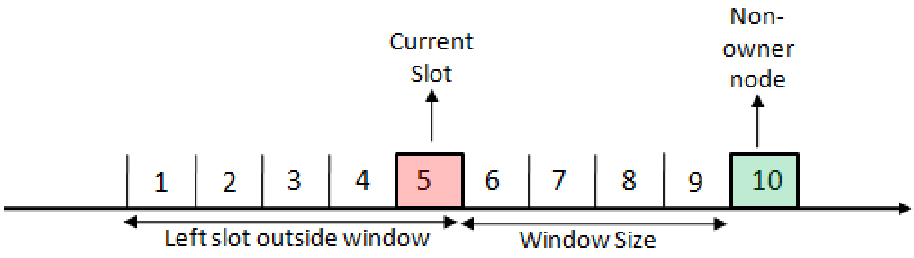


Figure 3.5: When current slot is at the left of the window

sibility to send its data successfully. The detailed procedure is given in Algorithm 2.

In Figure 3.5 and 3.6, 10 member nodes are considered who want to send their data to CH and window size is 4. Basically, this mechanism consider $(\lfloor \frac{n}{2} \rfloor - 1)$ as window size. Here, slot is divided into two categories: left side slot outside window and right side slot outside window. In clustering protocol, CH collects all data from the TDMA schedule, aggregates them and sends to BS. So, if a node has a single data into D_{bs}, then it does not send its data within non owners slot. According to Figure 3.5, suppose the current slot is 5 and node 10 has data to send. In this case, node 10 only sends its data if it has more than one data into its buffer; otherwise it wait for its own slot. On the other hand, if the current slot is right outside the window for a node, then it does not consider the buffer size for sending its data. According to Figure 3.6, the current slot is 5 but node 3 has data to send. To send its data, in traditional clustering protocol, it has to wait until next frame, but here, without considering buffer size, node 3 immediately tries to send it data into the 5^{th}

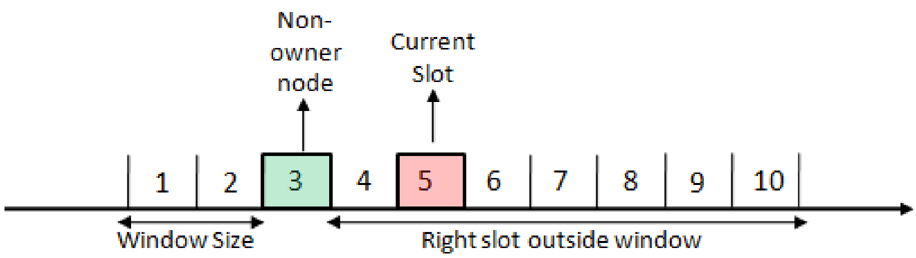


Figure 3.6: When current slot is at the right of the window

Table 3.1: Summary of the used notation

Symbol	Description
N_c	Number of nodes in the cluster
W	Window size
OS	Owners slot
NOS	Non Owner's slot
S_{BW}	Slot behind the window
S_{AW}	Slot after the window
D_{bf}	Number of packet into the queue
Cs	Current Slot
Fs_t	Fixed Slot Transmission
Rs_t	Random Slot Transmission
ST	Successful Transmission
Th	Threshold
Rs_s	Random slot success
Rs_f	Random slot failure

Algorithm 2 Slot Reuse

```
When data generate outside the window
if (Cs is S_bw && D_bf ≥ 2) then
    sends data as non owner node
else if (Cs is S_aw) then
    sends data as non owner node
else
    waits for its own slot
end if
```

slot. This not only increase the number of packets send to BS but also decrease the per packet delay.

3.3.2.2 Adaptive Contention Using Näive Bayes Classifier

When the window size is low, then the intra cluster behaves like contention based protocol and when the window size is high then it behaves like schedule based protocol. In this subsection, all CHs handle the contention by varying the window size according to the successful transmission value of the previous frame. Näive Bayes Classifier is a classification technique based on Bayes' Theorem with an assumption of independence among predictors [45, 46]. Based on the predetermined attributes x_1, x_2 .... x_n, bayes theorem can find the most possible hypothesis and according to maximum value of the hypothesis it can conclude the most desirable output.

$$P(c|X) = P(c_j)\prod_i P(x_i|c_j) \tag{3.7}$$

Training data

The classification algorithm is trained with a set of training data in the Näive Bayes classifier. Generally, the training data is obtained by either experimental results or intuitive information. In this research work, the intuitive information for training set is used. The training set is shown in Table 3.2.

Näive Bayes classifier application

Inputs of the classifier are slot status, transmission status and new window status. Attributes of slot status are divided into two categories: Fixed slot is greater than Random slot ($FS \geq RS$) and fixed slot is less than random slot ($FS < RS$) . Attributes of Transmission status are divided into three categories: Successful transmission is less than threshold value ($ST < Th$), successful transmission is greater than threshold ($ST > Th$) and successful transmission is equal to threshold ($ST = Th$). Finally, the attributes of new window status are divided into two categories: Random slot success is greater or equal to the Random slot failure ($RS_s \geq RS_f$) and Random slot success is less than Random slot failure ($RS_s < RS_f$). The prior probability can be calculated from Table 3.2 and the results are shown in Table 3.3. The traffic condition is predicted by using prior probability value in

Table 3.2: A training set containing data rows

Slot	Transmission	New window status	Class
FS≥RS	ST>Th	$RS_s \geq RS_f$	High
		$RS_s < RS_f$	High
	ST<Th	$RS_s \geq RS_f$	Low
		$RS_s < RS_f$	High
	ST=Th	$RS_s \geq RS_f$	Medium
		$RS_s < RS_f$	High
FS<RS	ST>Th	$RS_s \geq RS_f$	High
		$RS_s < RS_f$	High
	ST<Th	$RS_s \geq RS_f$	Low
		$RS_s < RS_f$	High
	ST=Th	$RS_s \geq RS_f$	Medium
		$RS_s < RS_f$	Medium

Equation 3.7. For example, when FS $\geq$ RS, ST $>$ Th and $RS_s < RS_f$ the most probable state (High) can be derived from the prediction algorithm as follows:

Individual Probability P(High) $= \frac{7}{12}$
P(Medium) $= \frac{3}{12}$
P(Low) $= \frac{2}{12}$

Individual probability of each class

P(High| FS≥RS, $ST\geq$ Th , $RS_s < RS_f$) $= \frac{4}{6} \times 1 \times \frac{2}{6} = \frac{2}{9}$

P(Medium| FS≥RS, $ST\geq$ Th , $RS_s < RS_f$) $= \frac{1}{6} \times 0 \times \frac{2}{6} = 0$

P(Low| FS≥RS, $ST\geq$ Th , $RS_s < RS_f$) $= \frac{1}{6} \times 0 \times \frac{2}{6} = 0$

Table 3.3: Training data

Instance	**Attributes**	**High**	**Medium**	**Low**
Slot	FS≥RS	$\frac{4}{6}$	$\frac{1}{6}$	$\frac{1}{6}$
	FS<RS	$\frac{3}{6}$	$\frac{2}{6}$	$\frac{1}{6}$
Transmission	ST>Th	1	0	0
	ST<Th	$\frac{2}{4}$	0	$\frac{2}{4}$
	ST=Th	$\frac{1}{4}$	0	$\frac{3}{4}$
Window Status	$RS_s \geq RS_f$	$\frac{2}{6}$	$\frac{2}{6}$	$\frac{2}{6}$
	$RS_s < RS_f$	$\frac{5}{6}$	0	$\frac{1}{6}$

Now probable state can be calculated as follows:

P(FS≥RS, $ST>$ Th , $RS_s < RS_f$ | High)= $\frac{2}{9} \times \frac{7}{12} = \frac{7}{54}$

P(FS≥RS, $ST>$ Th , $RS_s < RS_f$ | Medium)= $0 \times \frac{3}{12} = 0$

P(FS≥RS, $ST>$ Th , $RS_s < RS_f$ | Low)= $0 \times \frac{2}{12} = 0$

So, the maximum probable state is High. According to the status of the class, this mechanism increases the value of window size when the class is high and decreases when the class is low.

3.4 Summary

In this chapter, an energy efficient and balanced clustering protocol for wireless sensor networks has been proposed. In this protocol, BS takes the decision of cluster head selection based on the residual energy, number of neighbor nodes and one hop CH information. Along with this, a capture based intra cluster communication mechanism is proposed with adaptive window selection for various traffic condition.

Chapter 4

Results and Performance Evaluation

In this section, performance of the proposed protocol evaluated through simulation study. For performance analysis, the proposed protocol is compared with pioneer clustering protocol (LEACH), one of the recent centralized Cluster Head selection algorithm (LEACH-MAC), and contemporary weight based clustering protocol (DWCA) by using OMNeT++ simulator [47]. A wireless sensor network of 50 and 75 nodes deployed randomly over a square area of 200 × 200 m^2. The sink node is placed in one side of the network. The simulation parameters are demonstrated in Table 4.1.

4.1 Performance Metrics

For performance evaluation, the following metrics are used in this chapter:

Network lifetime: The lifetime of a WSN represents the total amount of time over which the network remains operational [48]. The lifetime of the network emphatically relies on individual life span of each single node that constitutes the WSN. It is fundamentally reliant upon two main considerations: firstly, the measure of the energy it spent over rounds and also, the measure of energy needed for data aggregation and data sending. The lifetime is individually corresponding to the measure of data received from the network. Thus, the lifetime of sensor network is the amount of time that the sensor nodes will stay alive in the network. This work also used First Node Dies (FND), 10^{th} Node Dies (10^{th}ND) and Last Node Dies (LND) metrics as the network lifetime. FND is the time interval from the start of the operations until the first node dies, 10^{th}ND is the time interval from the start of the operations until the 10^{th} node is death and LND is the time interval from the start of the operations until the last node dies.

Table 4.1: Network Parameters

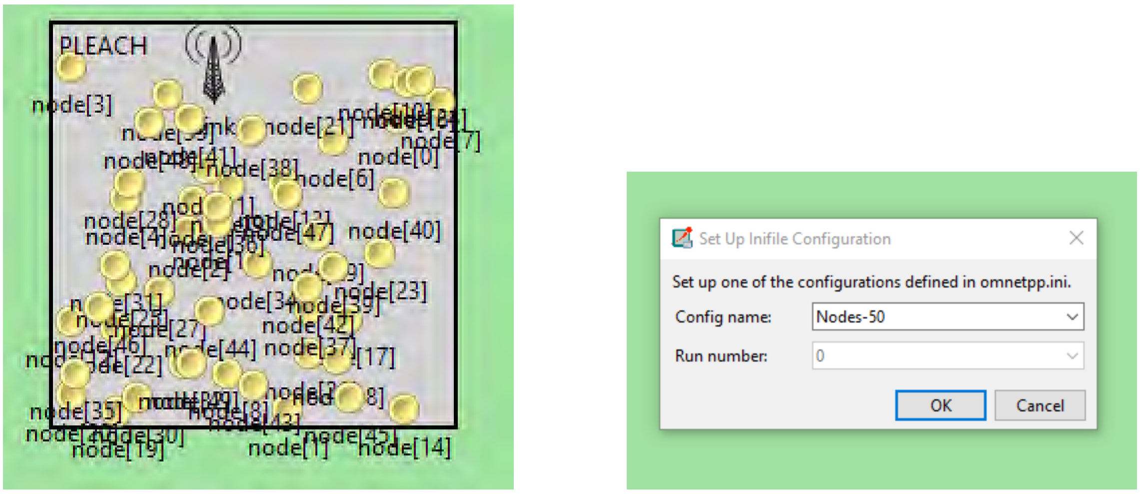
Parameter	**Value**
Network Area	200 × 200
Number of Nodes	50/75
Data Packet	512 bytes
Control Packet	32 bytes
Electronics energy, E_{elec}	50 nJ/bit
Free space loss, ϵ_{fs}	10 pJ/bit/m^2
Multipath loss, ϵ_{mp}	0.0013 pJ/bit/m^4
Aggregation energy, E_{DA}	5 nJ/bit/signal

Number of clusters: The lifetime of the network is affected based on the number of clusters created in each round. The clusters are created depending on the number of CHs selected by the algorithm. The selected cluster head nodes will exhaust energy rapidly, because they handle data aggregation and data forwarding. It is well observed that, after a certain rounds, number of cluster reduces the lifetime rather than increasing it. So, it is necessary to minimize the number of cluster heads to extend the lifetime of the network.

Throughput: The throughput of the network depends upon the whole number of packets transmit through CHs to BS and/or sink node by network in its whole life.

4.2 Simulation Environment

The Proposed network structure is presented in Figure 4.1. The proposed protocol simulated through various number of nodes. Fig.4.1b shows the dynamic selection window for the node deployment. According to Fig.4.1a, 50 sensor nodes are randomly deployed into the network and base station placed outside the network. Each node contain an unique id. The overall network area is 200 × 200 m^2. Topology after the selection of cluster head for first round is shown in Figure 4.2. Sample code of this simulation is attached in the Appendix A.



(a) (b)

Figure 4.1: Simulated Network in OMNeT++ (a) Node deployment in OMNeT++ (b) Number of node selection for the simulation

4.3 Simulation Results with Various Metrics

The simulation results is compared in terms of energy consumption per round, number of round required for total energy dissipation of first node, 10^{th} node and last node, remaining energy of the network over round, number of dead and alive node remain in the network after certain round, number of packet send to BS and number of CH per round.

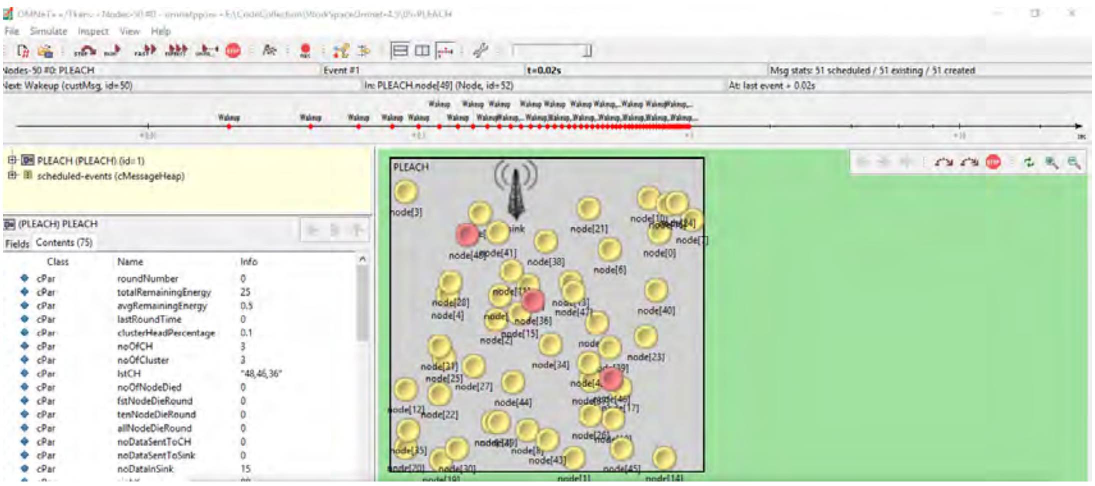
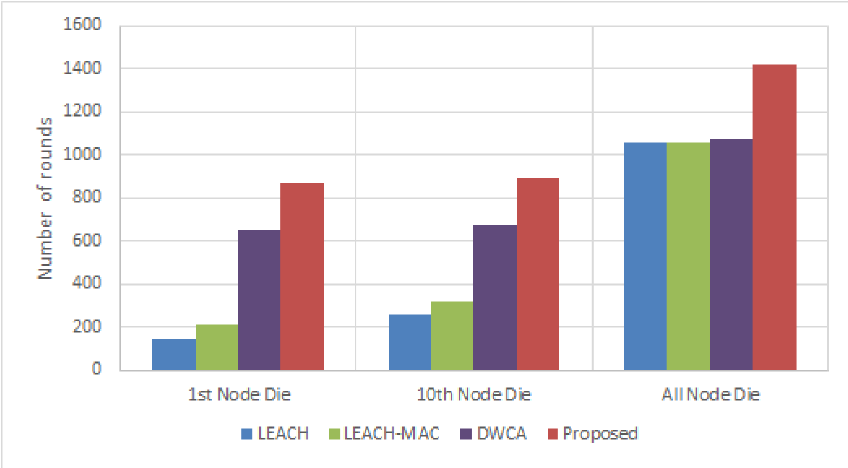
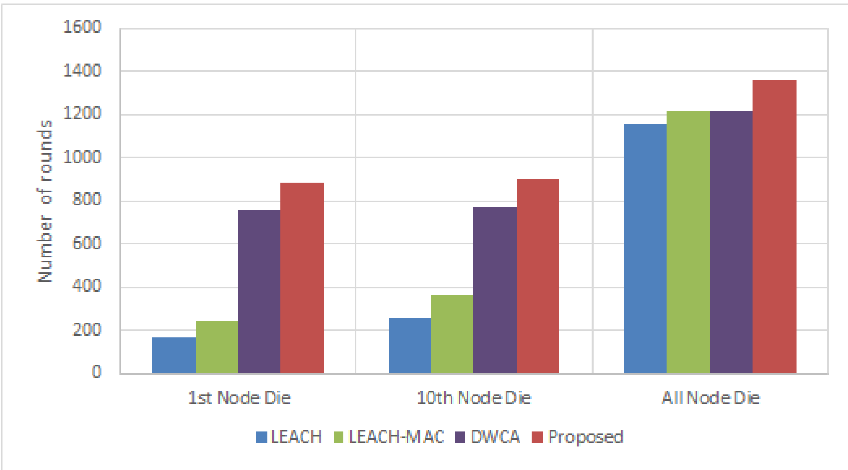


Figure 4.2: Topology after the selection of cluster head in OMNeT++



(a)



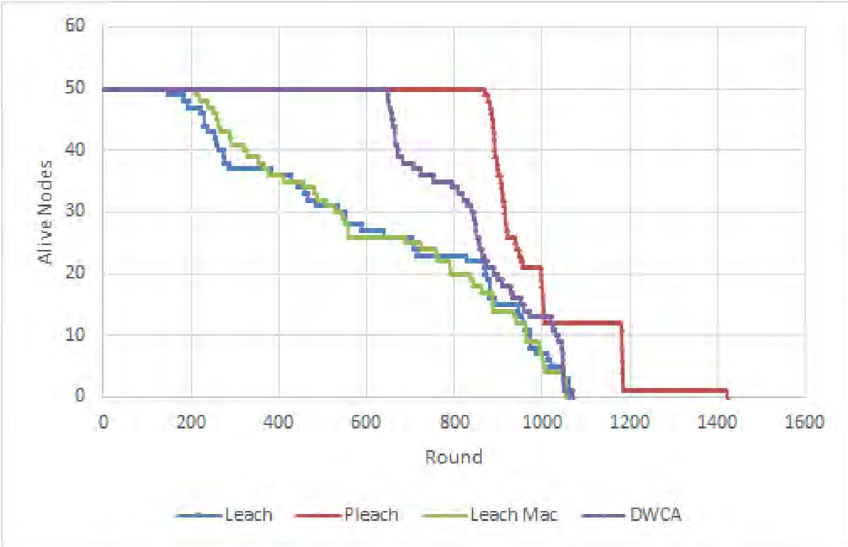
(b)

Figure 4.3: FND, 10^{th} ND and LND with respect to round (a) for 50 nodes (b) for 75 nodes

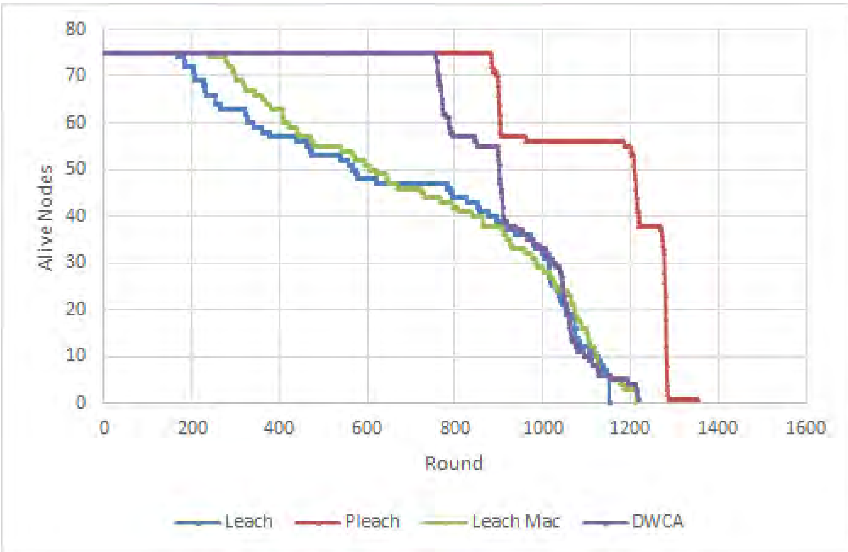
4.3.1 FND, 10^{th} ND and LND with Regard to Number of Round

The term FND, 10^{th} ND and LND was first used by Handy et al. [20]. Figure 4.3 demonstrates that the proposed protocol is more efficient than LEACH and LEACH-MAC in terms of FND, 10^{th}ND and LND. Figure 4.3a indicates that for 50 nodes both LEACH and LEACH-MAC first node dies within 200 rounds and in DWCA, first node dies after 600 rounds, whereas in proposed protocol the first node dies after 800 round. On the other hand, all node dies after 1400 rounds in the proposed protocol, whereas in LEACH, LEACH-MAC, and DWCA all node dies after 1000 rounds. Figure 4.3b shows the result of same metrics for 75 nodes.

4.3.2 Number of Alive Nodes with Respect to Rounds



(a)

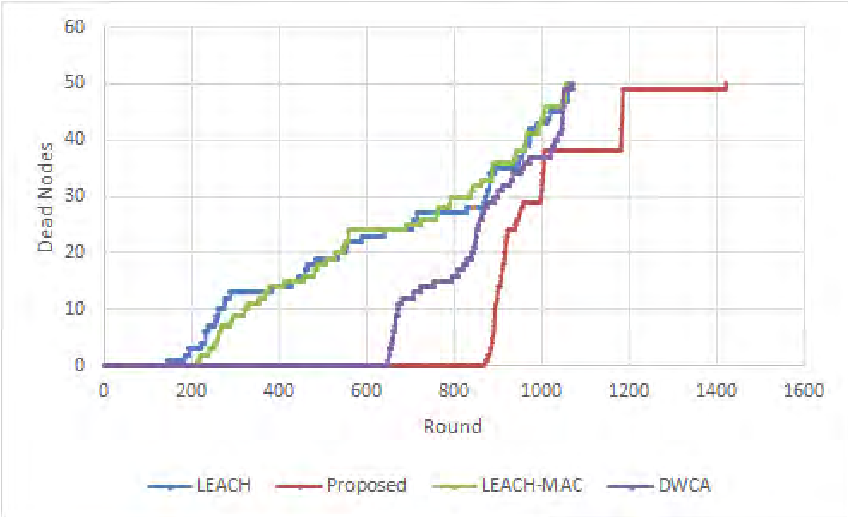


(b)

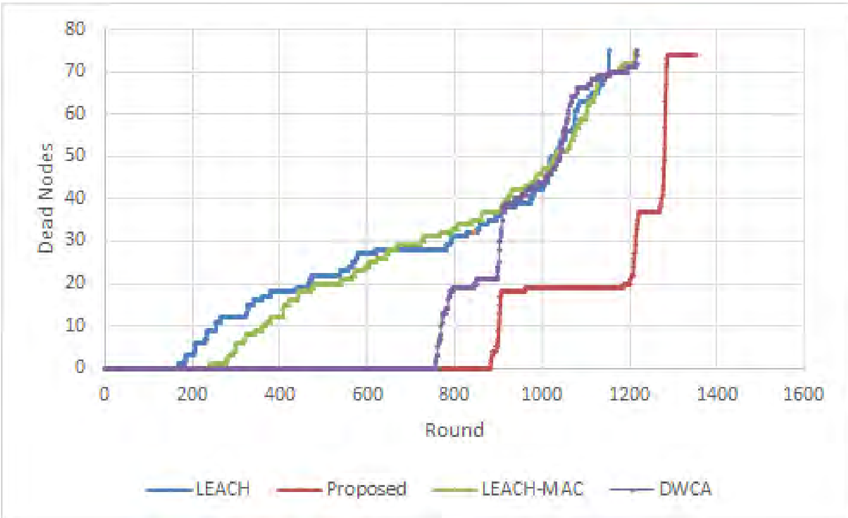
Figure 4.4: Number of alive nodes in the network with respect to round (a) for 50 nodes (b) for 75 nodes

Figure 4.4 shows the number of alive nodes with respect to operational iterative rounds. For both 50 nodes (Figure 4.4a) and 75 nodes (Figure 4.4b), the experiment is applied on same 200 $\times$ 200 region. That is why, number of alive node slightly improved for 75 nodes experiment. Performance results show that proposed protocol has maximum network lifetime as compared to other protocols. This enhancement in stability period is resulted because of the efficient deployment of nodes and centralization of clustering formation.

4.3.3 Number of Dead Nodes with Respect to Rounds



(a)

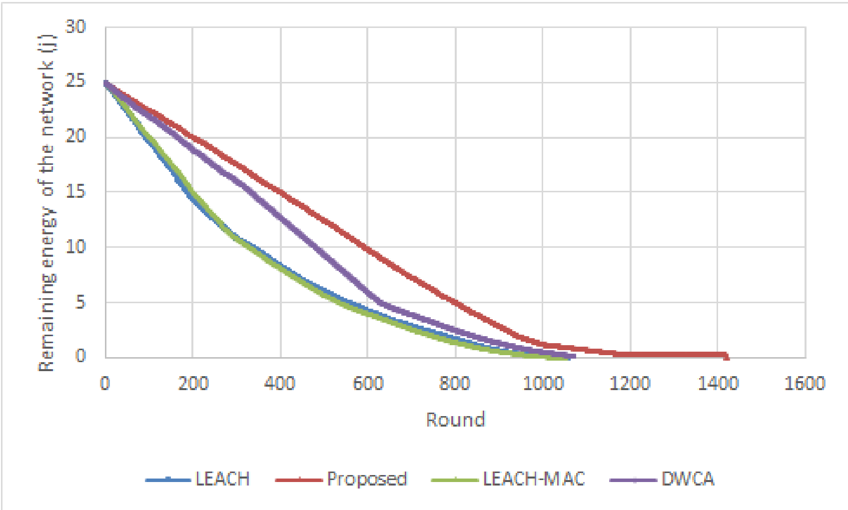


(b)

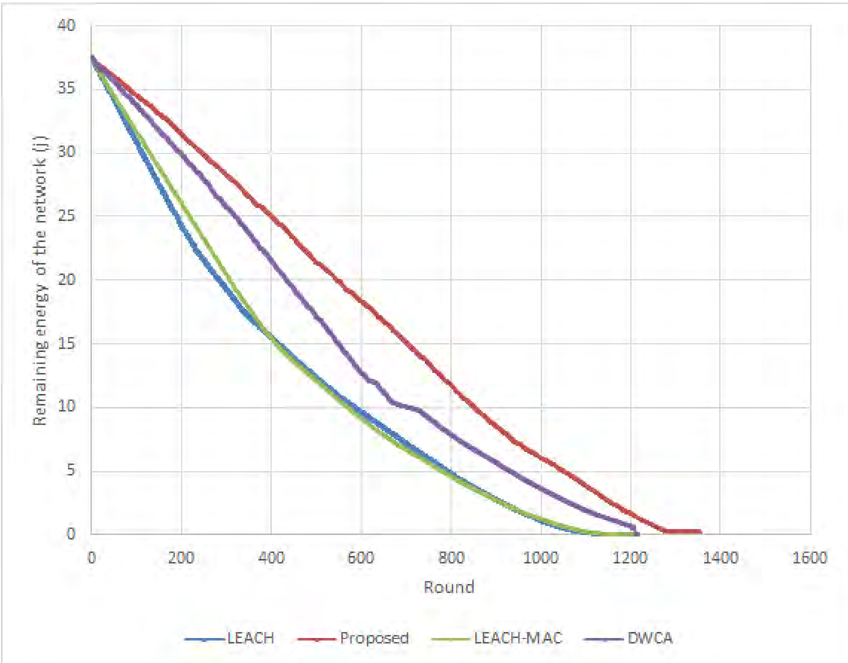
Figure 4.5: Number of dead nodes in the network with respect to round (a) for 50 nodes (b) for 75 nodes

In Figure 4.5, the increment of dead nodes explains the rate of instability as network operation proceeds. In this scenario, the proposed protocol performs well and indicates significant resistance in order to control the rate of dead nodes and to avoid drastic degradation of network performance with the passage of network operation. In the case of proposed protocol, last node dies after 1400 rounds. Another significant feature of proposed protocol is that instability period starts later as compared to other routing protocols. In proposed protocol, nodes do not die instantaneously during the instable period as it happens in the case of LEACH, LEACH-MAC, and DWCA.

It shows that proposed protocol is more resilient during the instability period and it continues to send the sensed data from the network field as long as possible.

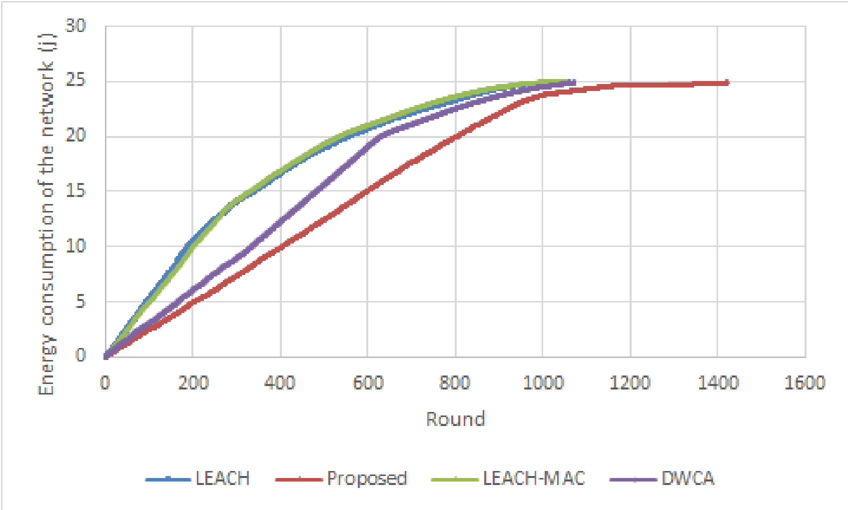


(a)

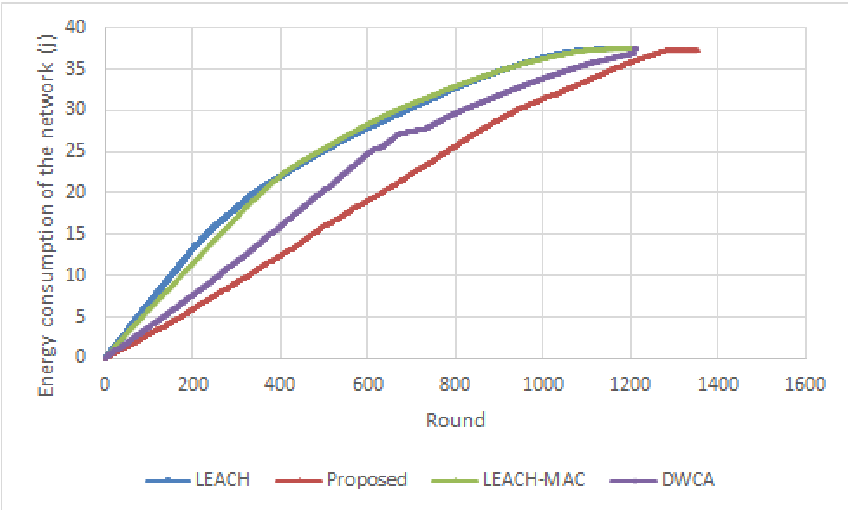


(b)

Figure 4.6: Remaining energy of the network with respect to round (a) for 50 nodes (b) for 75 nodes



(a)



(b)

Figure 4.7: Energy consumption of the network with respect to round (a) for 50 nodes (b) for 75 nodes

Since proposed protocol is dealing with nodes according to their radius, it minimize the multi path transmission within cluster. Nodes joining the CH must have the distance from CH smaller than d_0. This is the primary reason of prolonged lifespan of WSNs using proposed protocol. The proof is attached in Appendix B.

4.3.4 Remaining Energy of the Network Per Round

Figure 4.6 demonstrates the performance comparison in terms of overall remaining energy of the network with respect to rounds. For remaining energy comparison, Y-axis contains the value of total energy of 50 and 75 nodes. The remaining energy of the network ends after 1400 rounds for 50 nodes in the proposed protocol, whereas, for LEACH, LEACH-MAC, and DWCA remaining energy of the network ends immediate after 1000 rounds.

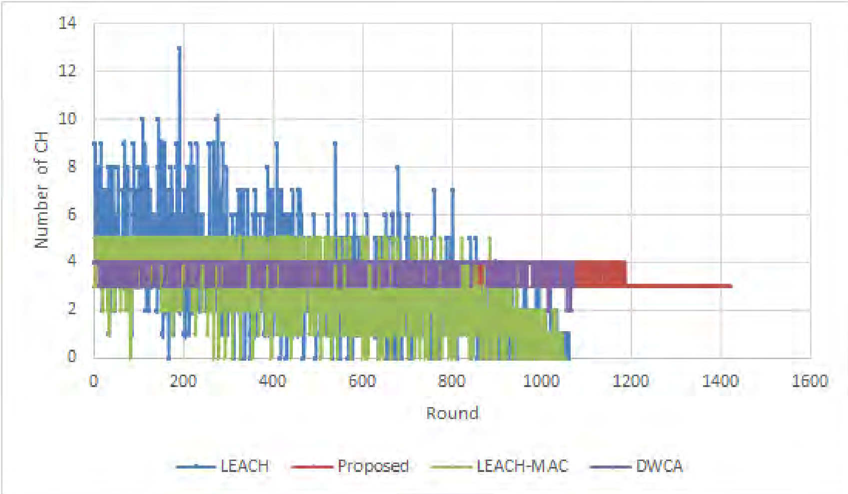
4.3.5 Energy Consumption of the Network Per Round

Energy consumption of network per round is shown in Figure 4.7. It is observed that the proposed scheme consumes less energy than previous scheme as residual energy per round is much higher in our protocol as shown in Figure 4.7. A major constituent of energy consumption is communication process. Almost 70 percent of whole network's energy is consumed in communication. So, a proper communication model is very much necessary for any energy efficient clustering protocol. In our proposed scheme, an adaptive intra-cluster communication mechanism is proposed which reduce the idle listening by CH by using the idle slot.

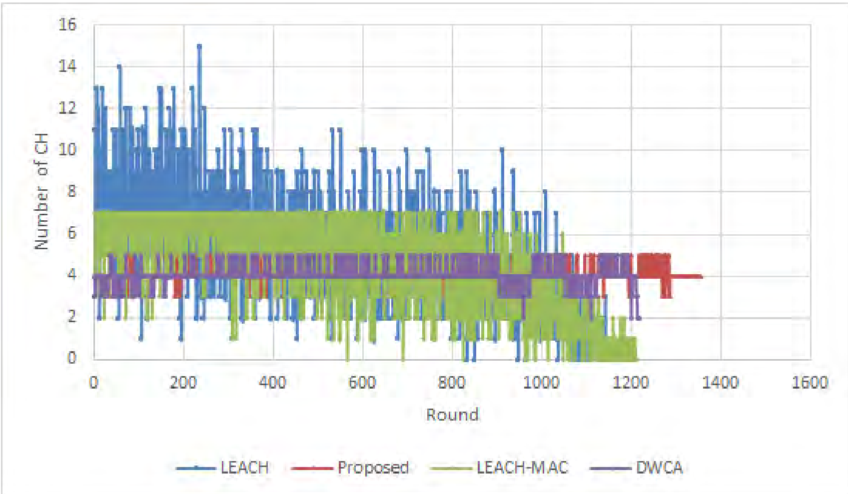
4.3.6 Number of Selected CH Per Round

Figure 4.8 exhibits the amount of cluster heads made in each round. The number of CHs selected in each round is not equal. According to the transmission range and available energy of a node, the clusters are created. If any node does not come under the close proximity of any CH, then that turns into an isolated CH. Furthermore, the lifetime of the network is very subject to the number of cluster heads appointed. At the point when the number of cluster heads is more, it consumes a lot of energy in data forwarding, which decreases the lifetime of the network.

The proposed protocol outperforms other protocols as it consistently provides optimal number of CHs in every round. LEACH and LEACH-MAC perform zigzag manner for Cluster Head selection and fail to provide guaranteed number of CHs, and it is because of the distributed nature of their CHs selecting algorithm. In DWCA, nodes compete for cluster head selection within a particular region, so the number of cluster head does not fluctuate like LEACH and LEACH-MAC. Both Figure 4.8a and Figure 4.8b shows that, in the proposed protocol, the number of CH varies from $4 \sim 5$. On the other hand, there is no restriction of CH selection for distributed approach. In LEACH, for the very first round the number of CH relies on $10 \sim 14$ nodes that creates extra energy consumption.



(a)



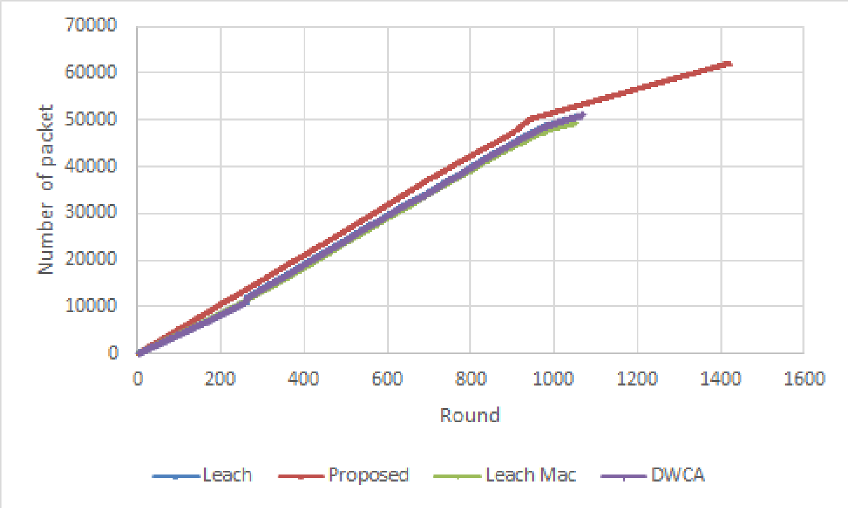
(b)

Figure 4.8: Number of selected CH with respect to round (a) for 50 nodes (b) for 75 nodes

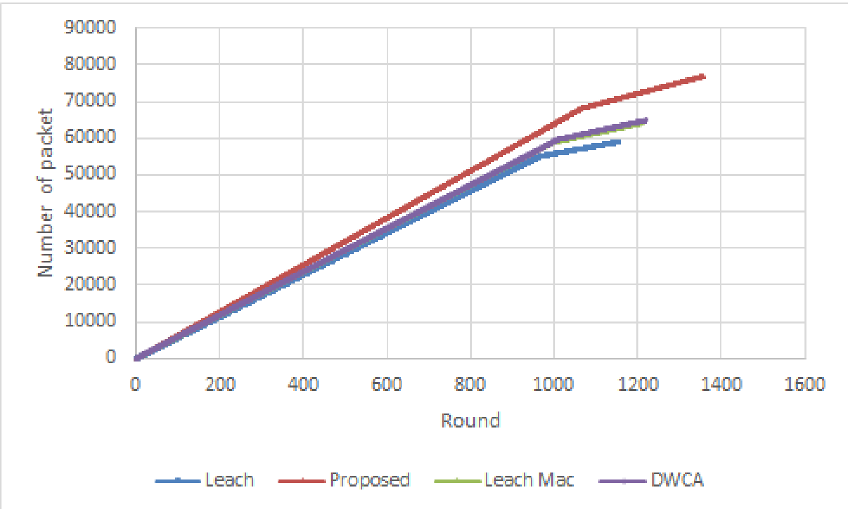
4.3.7 Number of Packets Sent to Base Station per Round

Figure 4.9 shows the total number of packets sent to BS for both 50 and 75 node scenario. Due to proper CH selection and communication model, network throughput of the proposed scheme is much greater than previous schemes.

According to the simulation results, Table 4.2 and 4.3 shows the comparative performance analysis of the competitive protocols in terms of stability period, instability period, network lifetime and CH fluctuation.



(a)



(b)

Figure 4.9: Number of Transmitted packet to BS (a) for 50 nodes (b) for 75 nodes

Table 4.2: Performance comparison of competitive protocols for 50 nodes

Clustering Protocol	**Stability Period**	**Instability Period**	**Network Lifetime Round**	**CH Fluctuation**
LEACH	1 - 182	183 - 1072	1 - 1072	0 ∼ 13
LEACH-MAC	1 - 210	211 - 1074	1 - 1074	0 ∼ 5
DWCA	1 - 649	650 - 1072	1 - 1072	2 ∼ 5
Proposed	1 - 846	847 - 1412	1 - 1412	3 ∼ 4

Table 4.3: Performance comparison of competitive protocols for 75 nodes

Clustering Protocol	**Stability Period**	**Instability Period**	**Network Lifetime Round**	**CH Fluctuation**
LEACH	1 - 191	192 - 1186	1 - 1186	$0 \sim 15$
LEACH-MAC	1 - 221	222 - 1204	1 - 1204	$0 \sim 7$
DWCA	1 - 756	757 - 1218	1 - 1218	$1 \sim 6$
Proposed	1 - 864	865 - 1373	1 - 1373	$3 \sim 5$

4.4 Summary

In this chapter, the overall protocol performance is discussed through various metrics. The protocol has been tested on an energy consumption model that can be viewed as a realistic network. The empirical simulations has been conducted to achieve the adequate performance. The simulation results outperform its competent approaches.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

A lot of clustering protocol has been introduced in the last few years. Most of them consider average energy, location and distance for selecting cluster head in both randomized and centralized approach. But, most of them cannot create a fully balanced cluster. In this research work, energy information and neighbor information along with neighbor's CH status is used for cluster head selection. This thesis presents a new cluster head selection technique by considering residual energy, number of neighbor nodes and one hop cluster head status. Centralized algorithm is used for the selection of CHs, means BS takes the decision of CHs. BS controls CH fluctuation by considering one hop CH information that keep the number of CH at optimum rate. The significance of using one hop neighbor information for selecting CH is to avoid high number of CH in a particular region.

This research also improves intra-cluster communication by exploiting capture effect. Within TDMA schedule, the owner nodes send their data with high power and non-owner nodes compete beside window size with low power. The capture based intra-cluster communication not only increases the throughput but also decreases per packet latency. The capture based mechanism is helpful for low or moderate traffic but for the high traffic, this paper adapts the window size for capture. In each round, every CH calculate the window size by using Näive Bayes Classifier. When the window size is large, then system behaves like schedule based approach and when the window size is low, the system behaves like contention based approach.

OMNeT++ simulator is used to compare the results of the proposed scheme with LEACH, one of the recently proposed centralized clustering protocol (LEACH-

MAC), and weight based clustering protocol (DWCA). All results show that the network performance of the proposed scheme is better than previously proposed clustering techniques. This improvement is achieved by using balanced selection of CH and capture based traffic adaptive intra-cluster communication technique.

5.2 Future Work

The proposed protocol that generally manage the cluster head selection, cluster formation and intra-cluster communication can be reached out to some other areas of clustering like mobility of nodes and security in clustering. A further study will be useful to discover the behavior of mobile nodes in the proposed protocol and its resulting investigation. An alternate direction is to analyze the security features within the cluster. So, in a single sentence, the mobility and security features can be added to the proposed protocol to make the approach more efficient and reliable.

Bibliography

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] M. Dong, K. Ota, and A. Liu, "RMER: Reliable and energy-efficient data collection for large-scale wireless sensor networks," *IEEE Internet of Things Journal*, vol. 3, no. 4, pp. 511–519, 2016.

[3] J. Ben-Othman and B. Yahya, "Energy efficient and qos based routing protocol for wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 70, no. 8, pp. 849–857, 2010.

[4] J. Wang, Z. Zhang, F. Xia, W. Yuan, and S. Lee, "An energy efficient stable election-based routing algorithm for wireless sensor networks," *Sensors*, vol. 13, no. 11, pp. 14301–14320, 2013.

[5] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[6] Y. Ozturk and V. Nagarnaik, "A scalable distributed dynamic address allocation protocol for ad-hoc networks," *Wireless Networks*, vol. 17, no. 2, pp. 357–370, 2011.

[7] R. R. Rout and S. K. Ghosh, "Enhancement of lifetime using duty cycle and network coding in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 2, pp. 656–667, 2013.

[8] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad hoc networks*, vol. 3, no. 3, pp. 325–349, 2005.

[9] M. M. Zanjireh and H. Larijani, "A survey on centralized and distributed clustering routing algorithms for wsns," in *Vehicular Technology Conference (VTC Spring), 2015 IEEE 81st*, pp. 1–6, IEEE, 2015.

[10] M. M. Zanjireh, A. Shahrabi, and H. Larijani, "ANCH: A new clustering algorithm for wireless sensor networks," in *Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on*, pp. 450–455, IEEE, 2013.

[11] R. Soua and P. Minet, "A survey on energy efficient techniques in wireless sensor networks," in *Wireless and Mobile Networking Conference (WMNC), 2011 4th Joint IFIP*, pp. 1–9, IEEE, 2011.

[12] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *IEEE Communications surveys & tutorials*, vol. 15, no. 2, pp. 551–591, 2013.

[13] A. Patra and S. Chouhan, "Energy Efficient Hybrid multihop clustering algorithm in wireless sensor networks," in *Communication, Networks and Satellite (COMNETSAT), 2013 IEEE International Conference on*, pp. 59–63, IEEE, 2013.

[14] S. Arjunan and S. Pothula, "A survey on unequal clustering protocols in wireless sensor networks," *Journal of King Saud University-Computer and Information Sciences*, 2017.

[15] M. Aldeer, R. Howard, and A. Al-Hilli, "Minimizing energy consumption in transmit-only sensor networks via optimal placement of the cluster heads," in *Proceedings of the Eighth Wireless of the Students, by the Students, and for the Students Workshop*, pp. 36–38, ACM, 2016.

[16] J. Corn and J. Bruce, "Clustering algorithm for improved network lifetime of mobile wireless sensor networks," in *Computing, Networking and Communications (ICNC), 2017 International Conference on*, pp. 1063–1067, IEEE, 2017.

[17] M. M. Afsar and M.-H. Tayarani-N, “Clustering in sensor networks: A literature survey,” *Journal of Network and Computer Applications*, vol. 46, pp. 198–226, 2014.

[18] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*, pp. 10–pp, IEEE, 2000.

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

[20] M. Handy, M. Haase, and D. Timmermann, “Low energy adaptive clustering hierarchy with deterministic cluster-head selection,” in *Mobile and Wireless Communications Network, 2002. 4th International Workshop on*, pp. 368–372, IEEE, 2002.

[21] M. S. Ali, T. Dey, and R. Biswas, “ALEACH: Advanced LEACH routing protocol for wireless microsensor networks,” in *Electrical and Computer Engineering, 2008. ICECE 2008. International Conference on*, pp. 909–914, IEEE, 2008.

[22] M. C. M. Thein and T. Thein, “An energy efficient cluster-head selection for wireless sensor networks,” in *Intelligent systems, modelling and simulation (ISMS), 2010 international conference on*, pp. 287–291, IEEE, 2010.

[23] Z. Beiranvand, A. Patooghy, and M. Fazeli, “I-LEACH: An efficient routing algorithm to improve performance & to reduce energy consumption in Wireless Sensor Networks,” in *Information and Knowledge Technology (IKT), 2013 5th Conference on*, pp. 13–18, IEEE, 2013.

[24] M. Ye, C. Li, G. Chen, and J. Wu, “EECS: an energy efficient clustering scheme in wireless sensor networks,” in *Performance, Computing, and Communications Conference, 2005. IPCCC 2005. 24th IEEE International*, pp. 535–540, IEEE, 2005.

[25] T. G. Nguyen, C. So-In, and N. G. Nguyen, “Two energy-efficient cluster head selection techniques based on distance for wireless sensor networks,” in *Computer Science and Engineering Conference (ICSEC), 2014 International*, pp. 33–38, IEEE, 2014.

[26] W. Zhou, “Energy efficient clustering algorithm based on neighbors for wireless sensor networks,” *Journal of Shanghai University (English Edition)*, vol. 15, pp. 150–153, 2011.

[27] H. M. Abdulsalam and L. K. Kamel, “W-LEACH: Weighted Low Energy Adaptive Clustering Hierarchy aggregation algorithm for data streams in wireless sensor networks,” in *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*, pp. 1–8, IEEE, 2010.

[28] P. K. Batra and K. Kant, “LEACH-MAC: a new cluster head selection algorithm for wireless sensor networks,” *Wireless Networks*, vol. 22, no. 1, pp. 49–60, 2016.

[29] Y. Jia, C. Zhang, and K. Liang, “A Distributed Multi-competitive Clustering Approach for Wireless Sensor Networks,” *International Journal of Wireless Information Networks*, pp. 1–8, 2017.

[30] M. Elhoseny, A. Farouk, N. Zhou, M.-M. Wang, S. Abdalla, and J. Batle, “Dynamic Multi-hop Clustering in a Wireless Sensor Network: Performance Improvement,” *Wireless Personal Communications*, pp. 1–21, 2017.

[31] J. Wang, Y. Cao, J. Cao, H. Ji, and X. Yu, “Energy-Balanced Unequal Clustering Routing Algorithm for Wireless Sensor Networks,” in *International Conference on Computer Science and its Applications*, pp. 352–359, Springer, 2016.

[32] A. Essa, A. Y. Al-Dubai, I. Romdhani, and M. A. Eshaftri, “A New Dynamic Weight-Based Energy Efficient Algorithm for Sensor Networks,” in *Smart Grid Inspired Future Technologies: First International Conference, SmartGIFT 2016, Liverpool, UK, May 19-20, 2016, Revised Selected Papers*, pp. 195–203, Springer, 2017.

[33] S. Sharma and S. K. Jena, "Cluster based multipath routing protocol for wireless sensor networks," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 2, pp. 14–20, 2015.

[34] S. Rani, S. H. Ahmed, R. Talwar, and J. Malhotra, "Can Sensors Collect Big Data? An Energy Efficient Big Data Gathering Algorithm for WSN," *IEEE Transactions on Industrial Informatics, 2017*, 2017.

[35] M. Tarhani, Y. S. Kavian, and S. Siavoshi, "SEECH: Scalable energy efficient clustering hierarchy protocol in wireless sensor networks," *IEEE Sensors Journal*, vol. 14, no. 11, pp. 3944–3954, 2014.

[36] N. Wang, Y. Zhou, and W. Xiang, "An Energy Efficient Clustering Protocol for Lifetime Maximization in Wireless Sensor Networks," in *Global Communications Conference (GLOBECOM), 2016 IEEE*, pp. 1–6, IEEE, 2016.

[37] A. Karmaker and M. M. Hasan, "A CSMA based intra cluster communication technique for saving cluster head energy," in *Computer and Information Technology (ICCIT), 2013 16th International Conference on*, pp. 267–270, IEEE, 2014.

[38] A. Zeb, A. M. Islam, M. Zareei, I. Al Mamoon, N. Mansoor, S. Baharun, Y. Katayama, and S. Komaki, "Clustering Analysis in Wireless Sensor Networks: The Ambit of Performance Metrics and Schemes Taxonomy," *International Journal of Distributed Sensor Networks*, vol. 12, no. 7, p. 4979142, 2016.

[39] B. Srikanth, M. Harish, and R. Bhattacharjee, "An energy efficient hybrid mac protocol for wsn containing mobile nodes," in *Information, Communications and Signal Processing (ICICS) 2011 8th International Conference on*, pp. 1–5, IEEE, 2011.

[40] K. S. Hamza and F. Amir, "Centralized Clustering Evolutionary Algorithms for Wireless Sensor Networks," in *Proceedings of the 10th International Conference on Informatics and Systems*, pp. 273–277, ACM, 2016.

[41] R. K. Tripathi, *Base station positioning, nodes' localization and clustering algorithms for wireless sensor networks.* PhD thesis, INDIAN INSTITUTE OF TECHNOLOGY KANPUR, 2012.

[42] J. Metzner, "On improving utilization in ALOHA networks," *IEEE Transactions on Communications*, vol. 24, no. 4, pp. 447–448, 1976.

[43] K. Whitehouse, A. Woo, F. Jiang, J. Polastre, and D. Culler, "Exploiting the capture effect for collision detection and recovery," in *Embedded Networked Sensors, 2005. EmNetS-II. The Second IEEE Workshop on*, pp. 45–52, IEEE, 2005.

[44] S. Shimamoto, H. Kudo, and Y. Onozato, "Fixed assignment based window access with capture (fawac) for personal satellite communications," in *Global Telecommunications Conference, 1994. GLOBECOM'94. Communications: The Global Bridge., IEEE*, vol. 2, pp. 745–749, IEEE, 1994.

[45] K. P. Murphy, "Naive bayes classifiers," *University of British Columbia*, 2006.

[46] K. P. Murphy, *Machine learning: a probabilistic perspective.* MIT press, 2012.

[47] A. Varga, "Omnet++," *Modeling and tools for network simulation*, pp. 35–59, 2010.

[48] H. Yetgin, K. T. K. Cheung, M. El-Hajjar, and L. Hanzo, "A survey of network lifetime maximization techniques," *IEEE Communications Surveys & Tutorials*, 2017.

Appendix A

Sample Simulation Code

omnet.ini

```
[General]
network = PLEACHPckg.PLEACH
[Config Nodes-50]
description = "50 Wireless nodes"
PLEACH.noOfWirelessNode = 50
PLEACH.clusterHeadPercentage = 0.1
```

PLEACH.ned

```
package PLEACHPckg;
network PLEACH
{
    parameters:
        int roundNumber = default(0);
        double totalRemainingEnergy = default(0);
        double avgRemainingEnergy = default(0);
        double lastRoundTime = default(0);
        double clusterHeadPercentage = default(0.1);
        int noOfCH = default(0);
        int noOfCluster = default(0);
        string lstCH = default("0");
        int noOfNodeDied = default(0);
        int fstNodeDieRound = default(0);
```

```
        int tenNodeDieRound = default(0);
        int allNodeDieRound = default(0);
        int noDataSentToCH = default(0);
        int noDataSentToSink = default(0);
        int noDataInSink = default(0);
        int sinkX = default(80); //old = 230
        int sinkY = default(20); //old = 40
        int networkStatus = default(0);
        int netSizeX = default(200);
        int netSizeY = default(200);
        int noPacketSentToSink = default(0);
        int sendPacketToCH = default(0);
        double dToBS = default(0);
        int noOfWirelessNode @prompt("Number of Nodes") = default(2);
        @display("bgb=200,200");
    submodules:
        sink: Sink;
        node[noOfWirelessNode]: Node {
       @display("i=misc/node_vs,gold;p=230,140");
        }
}
```

Node.h

```
#ifndef NODE_H_
#define NODE_H_
#include <omnetpp.h>
#include <Sink.h>
#include <custMsg_m.h>
#include<vector>
class Node: public cSimpleModule {
public:
    int X;
```

```
int Y;
int netSizeX;
int netSizeY;
int noOfWirelessNode;
double batteryPower;
double G;
double clusterHeadPercentage;
char type;
double distanceToBS;
double sleepTime;
double Efs; //10*0.000000000001;
double Emp; //0.0013*0.000000000001;
double Do;
double CHETx;
double CHERx;
double NETX; //Node ETX
double NERX; //Node ERX
double ETX;
double ERX;
double EDA;
int CHIndex;
double roundInterval;
double deadTime;
int roundNumber;
double optimalClusterFactor;
int M; //Network size
double R; //Radius
double K; //optimum cluster
double dToBS;
int energyMarker;
double alpha;
double efactor;
```

```
    std::vector<int> neighborNode;
    double WGTV;
    double thresholdEnergy;
    int noDataSentToCH;
    int noDataSentToSink;
    int noDataInSink;
    custMsg *wakeup;
    custMsg *data;
    cQueue *dataQueue;
    cQueue *chDataQueue;
    cModule *senderNode;
    cModule *receiverNode;
    Node *tempTargetModule;
    Sink *tempBaseModule;
    cModule *tempModule;
    Node *tempNode;
    cModule *tempSrcModule;
    Node *tempSrcNode;
    cModule *tempDestModule;
    Node *tempDestNode;
    cModule *calModule;
    Node *calNode;
    cModule *calSinkModule;
    Sink *calSinkNode;
    cModule *sinkModule;
    Node();
    virtual ~Node();
private:
    void SetCoordinate();
    void CalculateAvgDistanceToBS();
    void CalculateNeighborNode();
    void SetEnergyMarker();
```

```
    void CalculateWGTV();
    void ClusterHeadSelection(int roundNo);
    void ClusterFormation(int roundNo);
    int CalculateDistanceToBS(int nodeindex);
    custMsg* CreateCustMsg(const char *name);
    int CalculateDistance(int senderIndex, int receiverIndex);
    void SendDataToCH();
    void SendDataToSink();
protected:
    virtual void initialize();
    virtual void handleMessage(cMessage *msg);
    void finish();
};
#endif /* NODE_H_ */
```

Node.cc

```
#include <Node.h>
#include <Sink.h>
#include<RANDOM.H>
#include<custMsg_m.h>
#include<algorithm>
#include<string>
#include<stdlib.h>
#include<cmath>
Define_Module(Node);
Node::Node() {
    this->X = 0;
    this->Y = 0;
    batteryPower = 0.5;
    this->G = 0.0;
    this->type = 'N'; //Normal Node
    sleepTime = 0.10;
```

```
    Efs = 10 * 0.000000000001;
    Emp = 0.0013 * 0.000000000001;
    Do = sqrt(Efs / Emp);
    CHETx = 0.0;
    CHERx = 0.0;
    NETX = 0.0; //Node ETX
    NERX = 0.0; //Node ERX
    ETX = 50 * 0.000000001;
    ERX = 50 * 0.000000001;
    EDA = 5 * 0.000000001;
    CHIndex = 0;
    roundInterval = 1.0;
    optimalClusterFactor = 100; //Long value as inifinity
    alpha = 0.1;
    efactor =  0.0;
    dToBS = 109.06;
    thresholdEnergy = 0.23; //for 50 nodes
    noDataSentToCH = 0;
    noDataSentToSink = 0;
}
void Node::initialize() {
    dataQueue = new cQueue;  //Queue for incoming message
    chDataQueue = new cQueue;
    wakeup = CreateCustMsg("Wakeup");
    noOfWirelessNode = getParentModule()->par("noOfWirelessNode");
    clusterHeadPercentage = getParentModule()->par("clusterHeadPercentage");
    this->netSizeX = getParentModule()->par("netSizeX");
    this->netSizeY = getParentModule()->par("netSizeY");
    SetCoordinate();
    this->K = (sqrt(noOfWirelessNode) / sqrt(2 * PI)) * sqrt(Efs / Emp)
    this->R = sqrt((double) (netSizeX * netSizeX) / (PI * K));
    if (simTime().dbl() == 0.0 && getIndex() == (noOfWirelessNode - 1)) {
```

```
        CalculateAvgDistanceToBS();
        CalculateNeighborNode();
        SetEnergyMarker();
        CalculateWGTV();
        //endSimulation();
        ClusterHeadSelection(0);
        ClusterFormation(0);
    }
    scheduleAt(
            simTime().dbl()
                    + ((double) noOfWirelessNode - getIndex())
                            / (double) noOfWirelessNode, wakeup);
}
void Node::handleMessage(cMessage *msg) {
   int networkStatus = getParentModule()->par("networkStatus");
    double lastRoundTime = getParentModule()->par("lastRoundTime");
    roundNumber = getParentModule()->par("roundNumber");
    int noOfNodeDied = getParentModule()->par("noOfNodeDied");
    custMsg *inMsg = check_and_cast<custMsg *>(msg);
    //Handle incoming data message
    if (strcmp("DataMsg", inMsg->getFullName()) == 0) {
        //Power consumtion for incomming message
        if (this->type == 'C' && this->batteryPower >= 0) {
            //For Cluster head
            this->batteryPower = this->batteryPower - this->CHERx;
            if (this->batteryPower <= 0) {
                noOfNodeDied++;
                getParentModule()->par("noOfNodeDied") = noOfNodeDied;
                if (noOfNodeDied == 1) {
                    getParentModule()->par("fstNodeDieRound") = roundNumber;
                }
                if (noOfNodeDied == 10) {
```

```
                getParentModule()->par("tenNodeDieRound") = roundNumber;
            }
            if (noOfNodeDied == noOfWirelessNode) {
                getParentModule()->par("allNodeDieRound") = roundNumber;
            }
        }
    }        inMsg->setPacketReachTime(simTime().dbl());
    chDataQueue->insert(inMsg);
}
//Initial setup
if (roundNumber <= 0 && getIndex() == noOfWirelessNode - 1) {
    roundNumber = roundNumber + 1;
    lastRoundTime = simTime().dbl();
    getParentModule()->par("roundNumber") = roundNumber;
    getParentModule()->par("lastRoundTime") = lastRoundTime;
    getParentModule()->par("networkStatus") = 2;
}
//Send data to CH
if (networkStatus == 2) {
    if (this->type == 'N') {
        SendDataToCH();
    }
    if (getIndex() == noOfWirelessNode - 1) {
        getParentModule()->par("networkStatus") = 3;
    }
}
//Send data to Sink
if (networkStatus == 3) {
    if (this->type == 'C') {
        SendDataToSink();
    }
    if (getIndex() == noOfWirelessNode - 1) {
```

```
            getParentModule()->par("networkStatus") = 1;
        }
    }
    //Setup state
    if (roundNumber > 0
            && simTime().dbl() >= lastRoundTime + this->roundInterval
            && getIndex() == noOfWirelessNode - 1 && networkStatus == 1) {
        roundNumber = roundNumber + 1;
        lastRoundTime = simTime().dbl();
        getParentModule()->par("roundNumber") = roundNumber;
        SetEnergyMarker();
        CalculateWGTV();
        ClusterHeadSelection(roundNumber);
        ClusterFormation(roundNumber);
        getParentModule()->par("networkStatus") = 2;
    }
    if (msg->isSelfMessage()) {
        custMsg *dataMessage = CreateCustMsg("DataMsg");
        dataQueue->insert(dataMessage);
        scheduleAt(simTime().dbl() + sleepTime, msg->dup());
    }
}
void Node::finish() {
    WriteOneTenAllNodeDeadHistory();
}
void Node::ClusterHeadSelection(int roundNo) {
    int noOfCH = 0;
    noOfCH = getParentModule()->par("noOfCH");
    int noOfNodeDied = getParentModule()->par("noOfNodeDied");
    WriteDeadNodeHistory(noOfNodeDied, roundNo - 1);
    WriteNetworkEnergyHistory(roundNo - 1);
    CountSinkPacket(roundNo - 1);
```

```
CountCH(roundNo - 1, noOfCH);
CountThroughput(roundNo - 1);
getParentModule()->par("noOfCH") = 0; //Reset cluster head
getParentModule()->par("noOfCluster") = 0;
double p = clusterHeadPercentage;
int roundNumber = roundNo;*/
double distance = 0.0;
int noOfCluster = 0;
double optimalRandValue = dblrand();*/
//Initially reset all node as normal node
for (int i = 0; i < noOfWirelessNode; i++) {
    tempModule = getParentModule()->getSubmodule("node", i);
    tempNode = check_and_cast<Node *>(tempModule);
    tempNode->type = 'N';
    tempNode->CHIndex = 0;
}
cModule *tempCurModule;
Node *tempCurNode;
cModule *tempNextModule;
Node *tempNextNode;
int sortedNode[noOfWirelessNode];
for (int i = 0; i < noOfWirelessNode; i++) {
    sortedNode[i] = i;
}
for (int i = 0; i < noOfWirelessNode; i++) {
    for (int j = 0; j < noOfWirelessNode - i - 1; j++) {
        int jValue = sortedNode[j];
        int jNextValue = sortedNode[j + 1];
        tempCurModule = getParentModule()->getSubmodule("node", jValue);
        tempCurNode = check_and_cast<Node *>(tempCurModule);
        tempNextModule = getParentModule()->getSubmodule("node",
                jNextValue);
```

```
            tempNextNode = check_and_cast<Node *>(tempNextModule);
            if (tempCurNode->WGTV > tempNextNode->WGTV) {
                int temp = sortedNode[j]; //Node index
                sortedNode[j] = sortedNode[j + 1];
                sortedNode[j + 1] = temp;
            }
        }
    }
    //Since it is ascending order
    cModule *tempNeigborModule;
    Node *tempNeigborNode;
    for (int i = noOfWirelessNode - 1; i >= 0; i--) {
        bool isCH = true;
        int sortedValue = sortedNode[i];
        tempCurModule = getParentModule()->getSubmodule("node", sortedValue);
        tempCurNode = check_and_cast<Node *>(tempCurModule);
        for (int j = 0; j < tempCurNode->neighborNode.size(); j++) {
            int neighborNodeIndex = tempCurNode->neighborNode[j];
            tempNeigborModule = getParentModule()->getSubmodule("node",
                    neighborNodeIndex);
            tempNeigborNode = check_and_cast<Node *>(tempNeigborModule);
            if (tempNeigborNode->type == 'C') {
                isCH = false;
                break;
            }
        }
        if (isCH == 1) {
            distance = CalculateDistanceToBS(i);
            tempCurNode->distanceToBS = distance;
            tempCurNode->type = 'C';
            noOfCluster = getParentModule()->par("noOfCluster");
            getParentModule()->par("noOfCluster") = noOfCluster + 1;
```

```
        int noOfCH = getParentModule()->par("noOfCH");
        noOfCH = noOfCH + 1;
        getParentModule()->par("noOfCH") = noOfCH;
        char buffer[33];
        itoa(sortedValue, buffer, 10);
        std::string strCH(buffer);
        if (noOfCH <= 1) {
            getParentModule()->par("lstCH") = strCH;
        } else {
            std::string prevList = getParentModule()->par("lstCH");
            getParentModule()->par("lstCH") = prevList + "," + strCH;
        }
        if (tempCurNode->distanceToBS > tempCurNode->Do) {
            tempCurNode->CHETx = (tempCurNode->ETX + tempCurNode->EDA)
                    * (4000)
                    + tempCurNode->Emp * 4000
                            * (distance * distance * distance * distance);
            }
            if (tempCurNode->distanceToBS <= tempCurNode->Do) {
                tempCurNode->CHETx = (tempCurNode->ETX + tempCurNode->EDA)
            * (4000)
            + tempCurNode->Efs * 4000 * (distance * distance);
            }
            tempCurNode->CHERx = (ERX + EDA) * 4000;
            tempCurNode->CHETx = tempCurNode->CHETx - efactor;
            tempCurNode->CHERx = tempCurNode->CHERx - efactor;
    }
}
 for (int i = 0; i < noOfWirelessNode; i++) {
 tempModule = getParentModule()->getSubmodule("node", i);
 tempNode = check_and_cast<Node *>(tempModule);
 if (tempNode->batteryPower > 0.0) {
```

```
      temp_rand = dblrand(); //Generate a random variable 0 to 1
      if (tempNode->G <= 0) {
      double rechargeValue = p / (1 - p * (roundNumber % (int) (round)(1 / p)));
      if (temp_rand <= rechargeValue) {
      tempNode->type = 'C'; //Cluster node
      tempNode->optimalClusterFactor = (double) optimalRandValue
      initialCH++;
      tempNode->G = round(1 / p) - 1;
      distance = CalculateDistanceToBS(i);
      tempNode->distanceToBS = distance;
      noOfCluster = getParentModule()->par("noOfCluster");
      getParentModule()->par("noOfCluster") = noOfCluster + 1;
      if (tempNode->distanceToBS > tempNode->Do) {
      tempNode->CHETx = (tempNode->ETX + tempNode->EDA)
      * (4000) + tempNode->Emp * 4000
      * (distance * distance * distance * distance);
      }
      if (tempNode->distanceToBS <= tempNode->Do) {
      tempNode->CHETx = (tempNode->ETX + tempNode->EDA) * (4000)
      + tempNode->Efs * 4000 * (distance * distance);
      }
      tempNode->CHERx = (ERX + EDA) * 4000;
      }
      }
      }
      }
}

void Node::ClusterFormation(int roundNo) {
    for (int i = 0; i < noOfWirelessNode; i++) {
        tempModule = getParentModule()->getSubmodule("node", i);
        tempNode = check_and_cast<Node *>(tempModule);
```

```
int minDistance = 32000;    //infinity
cModule *tempCHModule;
Node *tempCHNode;
int tempDistance = 0;
if (tempNode->type == 'N' && tempNode->batteryPower > 0) {
    for (int ch = 0; ch < noOfWirelessNode; ch++) {
        tempCHModule = getParentModule()->getSubmodule("node", ch);
        tempCHNode = check_and_cast<Node *>(tempCHModule);
        if (tempCHNode->type == 'C') {
            tempDistance = std::min(minDistance,
                    CalculateDistance(i, ch));
            if (tempDistance < minDistance) {
                minDistance = tempDistance;
                tempNode->CHIndex = ch;
                //min_dis_cluster=c;
            }

        } else {
            continue;
        }
    }
}
if(tempNode->type == 'C')
{
    continue;
}
if (minDistance > this->Do) {
    tempNode->NETX = ETX * (4000) + Emp * 4000 * (minDistance
    * minDistance * minDistance * minDistance);
}
if (minDistance <= Do) {
    tempNode->NETX = ETX * (4000)
```

```
            + Efs * 4000 * (minDistance * minDistance);
        }
        tempNode->NETX = tempNode->NETX - efactor;
    }
}
```

Appendix B

Node joining the CH must have the distance from CH smaller than d_0

First calculate the value of optimum cluster head k by using Equation 3.5

$$k = \frac{\sqrt{50}}{\sqrt{2 \times 3.1416}} \times \frac{\sqrt{10}}{\sqrt{0.0013}} \times \frac{200}{(109.73)^2}$$
$$= 4.1152$$

Now, calculate the radius R by using Equation 3.4,

$$R = \sqrt{\frac{200 \times 200}{3.1416 \times 4.71}}$$
$$= 55.6587$$

and,

$$d_0 = \sqrt{\frac{10}{0.0013}}$$
$$= 87.7$$

Algorithm 1 check every node's neighbor list. If any of it's neighbor already selected as CH, then BS deny it to make as CH. Otherwise, BS select the member node as a CH. That's why every node must join with a cluster having distance within it's radius R, otherwise it creates it own cluster. So, from the above derived value of R and d_0, it verify the statement mathematically that, "**Node joining the CH must have the distance from CH smaller than d_0**"