A BALANCED CLUSTERING APPROACH TO ENHANCE LIFETIME AND THROUGHPUT OF WIRELESS SENSOR NETWORKS

by

Saiful Islam

MASTER OF SCIENCE
IN
INFORMATION AND COMMUNICATION TECHNOLOGY

Institute of Information and Communication Technology
BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY
2019

The thesis titled "A BALANCED CLUSTERING APPROACH TO ENHANCE LIFETIME AND THROUGHPUT OF WIRELESS SENSOR NETWORKS" submitted by Saiful Islam, Roll No.: M10063104, Session: October-2006, has been accepted as satisfactory in partial fulfillment of the requirement for the degree of Master of Science in Information and Communication Technology on February 12, 2019.

BOARD OF EXAMINERS

1.	Dr. Md. Saiful Islam Professor IICT, BUET, Dhaka	Chairman
2.	Dr. Md. Saiful Islam Professor and Director IICT, BUET, Dhaka	Member (Ex- officio)
3.	Dr. Md. Rubaiyat Hossain Mondal Associate Professor IICT, BUET, Dhaka	Member
4.	Dr. Hossen Asiful Mustafa Assistant Professor IICT, BUET, Dhaka	Member
5.	Dr. Mohammod Abul Kashem Professor Dept. of CSE, Dhaka University of Engineering and Technology (DUET), Gazipur	Member (External)

CANDIDATE"S DECLARATION

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

Signature of the Candidate

Saiful Islam
M10063104
IICT, BUET

DEDICATION

I dedicate this thesis to my parents, daughter who is my life and beloved wife.

Table of Contents

List of Tables

List of Figures

List of Abbreviations of Technical Symbols and Terms

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 E_cient 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
IN	Internal Nodes
CBN	Central Border Nodes
BN	Balancing Nodes

Acknowledgement

All praises are for the Almighty Allah 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. Md. Saiful Islam, Professor and Director, 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 deep advice and the freedom he provided to me to do this research. I am grateful to him for his cooperation throughout my thesis work. Additionally, Md. Nurul Islam Khan who were with me 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 valuable comments. I would like to thank to all of my friends and colleagues for their cooperation. I am also grateful to my parents, daughter Rupkatha who is my life told me "You are a student of Nursery Class in BUET and I am in class One of my school" and beloved wife. Without their trust and support, I would have never arrived here.

Abstract

Wireless sensor networks (WSN) is a special type of Micro Electro-Mechanical System (MEMS) which is composed of large number of small, inexpensive and low powered sensor nodes. Lifetime is one of the crucial challenges of WSN. Balanced clustering may be a suitable solution of this challenge. Proper selection of cluster head (CH), cluster formation and a suitable intra-cluster communication technique can create a balanced clustering. Balanced clustering will extend lifetime as well as throughput. Many protocol exist for clustering. They used various matrices like residual energy, position or distance from base station (BS), neighbor set, neighbor information etc., but none of them can create balanced clustering. Moreover, many of them create back transmission which consume further energy and degrade lifetime. In this research, we offered a balanced clustering by selecting CH, cluster formation and a suitable intra-cluster communication technique.

The parameters residual energy (RE), number of neighbor nodes (NNN), one-hop neighbor information (ONI) and distance from nodes to BS (DNB) have been used for selecting a CH. RE information helps to select comparatively higher energetic node as CH, NNN helps to select CH from better density area of nodes of the network, ONI will restrict to select one CH from one cluster and DNB will reduce back transmission path. In cluster formation, energy may be wasted due to too high or too low cluster size. We have restricted it by using central border, internal and balancing nodes. We have used two threshold value (maximum and minimum) for ensuring suitable cluster size. Most of the clustering approach used TDMA for intra-cluster communication. But huge TDMA time slots may be unused due to data un-availability or lower trafic of slots owner‟s node. We have used three steps in intra-cluster communication (ICC) technique for sending more packets. At first, we used "power level adjustment" for sending non-owner node‟s data to CH by exploiting capture effect. Secondly, we used "time slot adjustment" by adjusting window size which ensured whether a non-owner node should send data or not. Lastly, we used "Preamble based CSMA" with waiting time adjusting according to power level of data for sensing channel, it will reduce collision. These three steps, ensure more packets sending to CH and BS, hence throughput increased more and more compare to previous works.

The proposed methods are evaluated by OMNeT++ simulator and compared with LEACH-C, LEACH-MAC and an energy efficient and balanced clustering approach for improving throughput (EEBCAIT). It is found that major improvement of performance in terms of First Node Death (FND), Tenth Node Death (TND), End Node Death (END), Consumption of Energy vs Rounds. Remaining Energy vs Rounds, Alive Nodes vs Rounds, Dead Nodes vs Rounds, CHs vs Rounds, Total Packets vs Rounds, Packets sending per round and improvement using of idle slots per round. We also found great improvement of lifetime and throughput.

CHAPTER 1
Introduction

Wireless sensor networks (WSN) is a special type of Micro Electro-Mechanical System (MEMS) which is composed of large number of small, inexpensive and low powered sensor nodes. WSN are rapidly becoming an integral part of human life. It is a widely used wireless communication technique which consists of huge number of sensor nodes and a sink node known as base station (BS) or super node. BS has enough resources such as energy, computation and memory. WSN formed by deployment of sensors. Deployment process of sensors are not engineered or pre-determined but is a random process. So, WSN needs a suitable self-organizing protocol for operation. Main duties of sensor nodes are to collect data by sensing or measuring area, aggregate, store and forward it to the BS through radio link. BS then sends data to real world for further processing. WSN is self-configured. In many applications, WSN is widely used such as military reconnaissance, vehicular movements, volcanic earthquake timing, weather forecasting, environment monitoring like temperature, humidity, pressure, motion, etc. [1-4]. WSN usually placed in remote and hazardous area [5]; so replacing or recharging of them is very difficult. A sensor can interact, process information locally and communicate with neighbors through radio link. A sensor consists of wireless modules or motes, sensor board and programmable board. WSN operates with low power and its lifetime mainly depends on efficient use of its power.

A WSN contains appropriate independent sensor nodes to sense physical or environmental conditions. These type of networks are basically data collecting networks, in which data are mainly associated with end users [6-7]. The general structure of a wireless sensor network is depicted in Fig. 1.1:

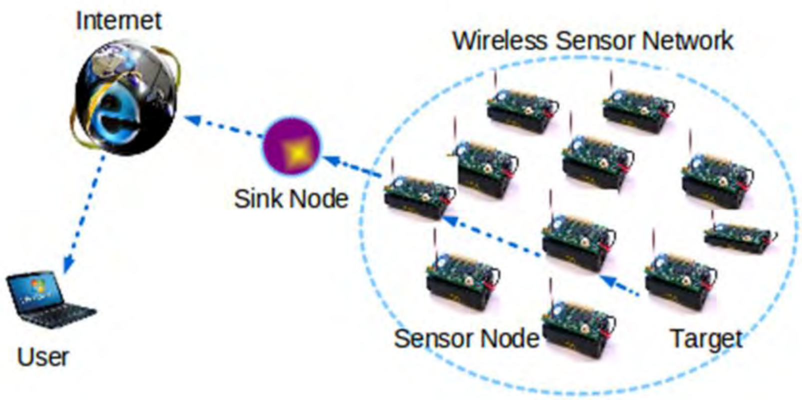


Figure 1.1: General structure of wireless sensor network [5]

1.1 Challenges in Wireless Sensor Networks

Though WSN is a widely used wireless technique, it has many challenges. These challenges should be overcome to achieve maximum benefit. Some important challenges are:

- Energy efficiency
- Coverage
- Security
- Data aggregation
- Minimum delay
- Throughput
- Scalability

Energy Efficiency: The main operational sustainability concern of WSN is its energy resource constraint. As WSN operates with limited energy, efficient uses of energy can maximize its lifetime**.** Most sensors of remotely used WSN are battery powered. Changing or recharging batteries of them is mostly impossible. So, maximizing lifespan of those WSN depends on these limited energies of the batteries. For example, in forest fire detection based applications, batteries of sensors are impossible to change or recharge. So, lifetime of the application depends on efficient uses of its power. Management of energy consumption is one of the main challenges. Efficient routing may overcome this issue and extend the network lifetime. Nowadays, many researchers are focusing on efficient use of power or energy.

Security: Security is another crucial issue of WSN. For unique characteristic of WSN, traditional security method of other networks may not be suitable. Lack of proper security mitigation would cause interruptions of network. Preventing access of data from unauthorized parties or from any malicious actions is very necessary. So, security management is also a major issue of WSN.

Coverage: Sensors need to be deployed for forming a WSN. This deployment may be done randomly or in self-organizing system. Deployment process of sensors are not engineered or pre-determined. So, sensor nodes are scattered and topology formed in an ad-hoc manner. To better perform, a suitable topology needs to cover every area of the network. Uniform distribution of CH is an influenced factor [9]. Position of CHs, too far or too close, is a reason of inefficient energy protocol [10].

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]. Data aggregation reduces number of transmission. So, it is also an energy efficient technique. Sensed data of sensors should be collected at a single point. Basically, base station is that point. It may very on several protocol. Closely located nodes may sense same data. So, there is a possibility of data duplication at center point. A proper data aggregation mechanism can reduce duplication of data as well as transmission.

Minimum Delay: Data should be quickly collected at base station without any sort of delay. It is also a primary objective of WSN. A proper routing protocol as well as network topology can ensure the delivery of the data with minimum delay.

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.

1.2 Literature Review

Amit et al. [50] proposed "**An Efficient Cluster Head Selection Strategy for Provisioning Fairness in Wireless Sensor Networks**". They used number of neighbor nodes, remaining energy and one hop neighbor information to select CH. This CH selection system ensure fairness and balanced situation that cover every region of the network but it cannot create balanced clustering due to back transmission problem. Let a network is divided into five clusters. In one of them, let CN-1 and CN-2 be two candidate nodes to become CH and assume both nodes have same residual energy. CN-1 has more neighbor nodes compared to CN-2. According to this protocol, CN-1 will be CH of that cluster. All member nodes then transmit data to CN-1. CN-1 will collect, aggregate and transfer that data to BS. If CN-2 were selected as CH, then over all path length (including distance between member nodes to CN-2, CN-2 to BS) will less than that of CN-1. So, more energy is consumed by CN-1 as CH compared to CN-2 and cluster will become unbalanced.

Another research work "**An Energy Efficient and Balanced Clustering Approach (EEBCA) for Improving Throughput of WSN"** [51] has been done for cluster formation. According to this method, those nodes that are already selected as CH by BS, broadcast CH advertisement (CH_{adv}) within its radius R. Member nodes receive CH_{adv} message from its neighbor CHs. After collecting all CH_{adv} messages, member nodes select nearest CH nodes according to 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. CHs collect $JOIN_{req}$

from all of their neighbors. Then, it allocates and announce Time Slot (TS_{alloc}) to all requesting nodes. Member nodes receive TS_{alloc}, then both CH and member nodes enter into steady state phase. From this cluster formation approach, it is seen that all non-CH nodes will join to own nearest CH according to RSSI value. It may be a cause of unbalanced clustering due to non-uniform distribution of nodes in the network. CHs with high density nodes area will create cluster with a large number of nodes while CHs with low density nodes area will create a cluster with a tiny number of nodes. CHs with large number of member nodes will consume huge energy to collect data from its members and send it to BS. So, this CH may die shortly. Thus, lifetime of the network will be degraded. Again, when any node will act as CH, it also has an area to collect data. But, there is no information how to collect CHs own data in this technique. So, there is a lack of data collection in this method as well as throughput may not be as expected. **CSMA based intra-cluster communication technique (CBICCT)** [37] proposed for reusing idle TDMA slots. This technique used preamble for sensing channel. If channel is free, then sends data otherwise waits for a random time. After that random time waiting, the node will send preamble for sensing again and so on. But, long preamble creates extra control packet overhead and increases delay. Again, if more than one node want to send different amount of data, and if one node with small amount of data sends preamble and gets channel free, then this node will send that small amount of data while other nodes with more data will be waiting. So, overall sending data or packet will be decreased. So, throughput of the network becomes less. We have tried to overcome this limitation in our protocol.

Capture Based Intra-Cluster Mechanism [51] technique proposed to detect higher powered data**.** In this 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 possibility to send its data successfully. At receiver (CH), authors proposed decode data and get highest powered data by capture effect. The system is good for transmitting data of non-owner data into idle slot. But problem is when multiple non-owner nodes have different amount of data. In this technique, no description is available about multi-level data of multiple non-owner nodes. Suppose an idle slot is available, but more than one non-owner nodes have data to send. Then, according to this technique, all non-owner nodes will send data with same power level. If buffer receives small amount of data of one non-owner node, while another non-owner node is waiting to transmit large amount of data, then overall throughput will be degraded. Again, for sending multiple data by multiple nodes, there is a scope of collision occurrence. So, there is a scope to research to transmit large amount of data of non-owner nodes as well as minimize collision and enlarge throughput of the network.

1.3 Motivation

Routing is an important technique for information sending or collecting from WSN. Several protocols exist for routing. Clustering is one of the pioneer for routing in WSN. Intra-cluster and inter-cluster communication is also a part of routing. An efficient routing protocol can save significant energy of WSN and can prolong lifespan. In clustering, network is divided into several areas known as cluster by selecting a CH. CH is a central node of the cluster and this CH is responsible to collect data from member nodes, aggregate them and finally send aggregated data to base station (BS). Member nodes send data to CH other than BS. This technique saves significant amount of energy. So, selection of CH has to be in a fare manner.

Position of CHs, if too close or too far away from each other, can make the cluster as well as whole network unbalanced. So, selection of CH is a critical task. Again, without a fair CH selection, some nodes and CH may be cause in back transmission. There exist some clustering algorithms to solve this issue. After selecting CH, cluster formation is one of the main issues. In non-uniform distributed WSN, clusters size may be too high or too low. So, there is a scope of unbalancing situation. In some cases, CHs are not able to collect data as a member node of the cluster. So there is lack of information collection. A suitable cluster formation protocol can avoid this unbalanced situation and lack of information collection. Again, in WSN most of the energy is spent for communication purpose, i.e., mainly for sending and receiving data. So, an efficient intra-cluster communication mechanism can reduce this communication stage, i.e., can save energy.

Thorough the review of some existing cluster based CH selection, cluster formation and intra-cluster communication 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 as well as throughput among nodes in a WSN.

All of above discussion is concerned with balancing load and energy consumption among nodes in a WSN. So, in order to design a cluster based algorithm, there are some aspect to be considered such as optimum CH selection, reducing back transmission, balanced cluster formation, overcome lack of information collection and intra-cluster communication mechanism. Therefore, designing a balanced clustering approach by agreeing the above mentioned challenging issues motivated us for this research work.

1.4 Research Objectives

The goal of the research is to enhance network lifespan and throughput of cluster based routing of WSN. For mitigation the goals the following aims have been identified:

- To develop a new CH selection algorithm for WSN to save significant amount of energy by reducing back transmission.
- To save CH energy for transmitting data as a member node of same cluster of WSN.
- To fulfill the lack of information collection by considering CH as a member node of same cluster.
- To save member nodes and CH energy by adding nodes with an appropriate CH.
- To increase throughput by developing energy balanced clustering of WSN.
- To develop an intra-cluster communication mechanism for achieving higher throughput.

- To compare the performance of the proposed scheme with existing clustering methods.

1.5 Organization of Thesis

The subsequent parts of the thesis are organized as follows:

Chapter 2: Salient Features of Balanced Clustering

An overview of routing techniques specially shortcoming about unbalancing of several existing protocol of CH selection, Clustering and Intra-Cluster Communication Technique have been described in this chapter. The available key research efforts also are described in this chapter.

Chapter 3: Proposed Protocol

The proposed protocol has been described in this chapter. A new CH selection approach reducing back transmission, creation of energy balanced cluster (EBC) and an intra-cluster communication (ICC) technique are also 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.

CHAPTER 2
Salient Features of Balanced Clustering

Sensors are randomly deployed to form WSN. These sensors have limited energy. So, efficient uses of this limited energy can improve lifetime of WSN. Most of the energy is spent for communication purpose, i.e., for transmitting and receiving data. So, a suitable routing protocol can reduce the transmission path and save energy. Designing of routing protocol is one of the main factors and it is full of challenges like limited power, low bandwidth, low computational memory, 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. Energy is the main concern of WSN design. Soua et al. [11] lists some sources 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**: It 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**: Each node located between transmission range and interference range receives a packet but cannot decode it.

2.2 Energy Efficient Routing Protocols

Some authors tried to categorize routing protocol. Pantazis et al. [12] categorized the routing protocols of WSN into four broad categories:

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

Beside this classification of routing protocol can be divide as follows:

I. Network Structure
   - Flat
   - Hierarchical

II. Communication
   - Query
   - Content
   - Negotiation

III. Topology Based
   - Location
   - Mobile Agent

IV. Reliable Routing
   - QOS
   - Multi path

2.3 Clustering in WSN

Various routing protocols exist in WSN. Clustering is one of the important protocols of them. Cluster based routing protocol is a cross layer solution. Recently, cluster based routing protocol attracts attention of researchers. In clustering, most of protocols work into two phases, i.e., setup phase and steady state phase as shown in Fig. 2.1. In setup phase, network is divided into several areas known as clusters by selecting a CH for each cluster. This clustering and CH selection process can be done according to centralized, distributed or hybrid manner [13]. In steady state phase, member nodes of every cluster send their sensed data to their CH in predefined TDMA slot assigned by the CH in the setup phase. CHs then collect data from member nodes, aggregate and send to BS. Our protocol works mainly in three steps: i) selection of CHs, ii) cluster formation, and iii) intra-cluster communication technique. In this section, we have discussed related CH selection, cluster formation and ICC algorithms.

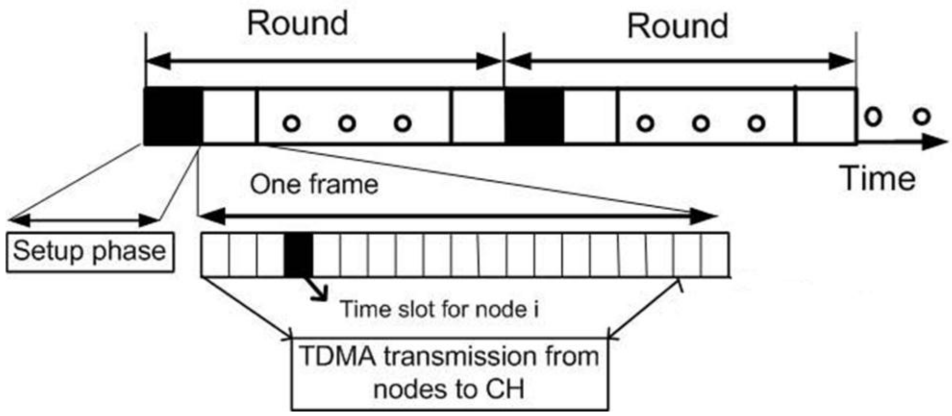


Figure 2.1: Working principle of cluster based protocol

2.3.1 Clustering Characteristic

To classify different clustering approaches, various clustering characteristics are used. According to Arjunan and Sujatha [14], following 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 2.2.

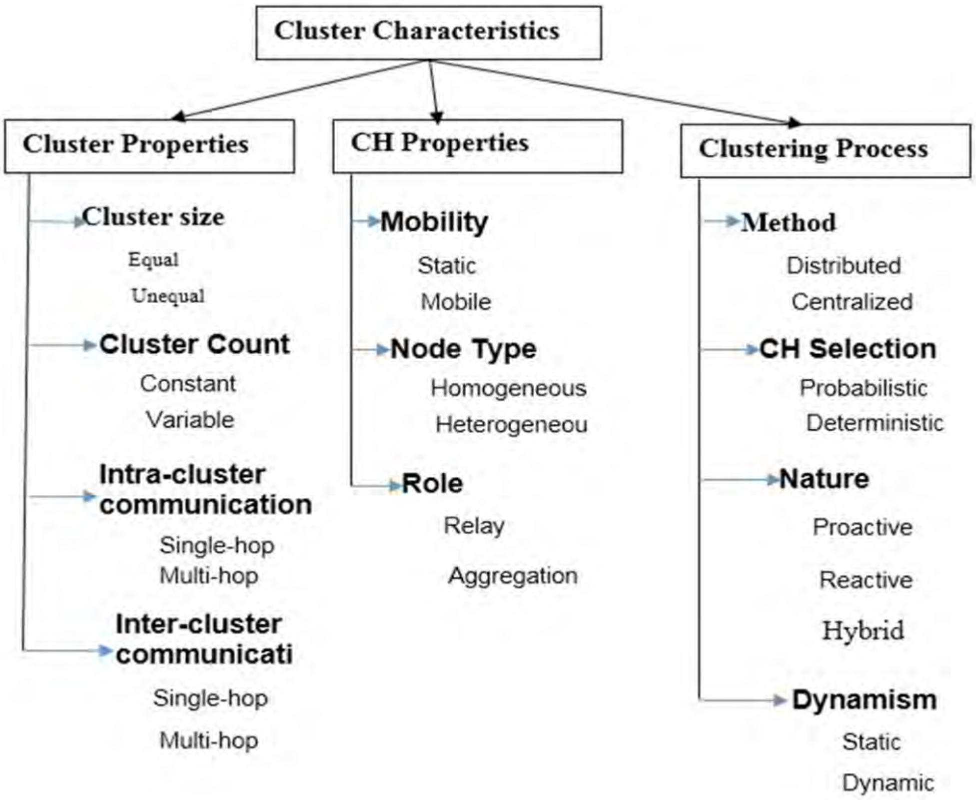


Figure 2.2: The clustering characteristics of WSN (redrawn from [17])

2.3.1.1 Cluster Properties

Cluster properties or specifications are cluster size, number of cluster, intra-cluster and inter cluster communication.

Cluster count: The number of cluster formed in WSN is known as cluster count. It may be predefined or variable that depends on application. Sometimes it may be a fixed number say 5% of total nodes in many applications. The number of clusters are variable when the CHs are randomly selected. Some clusters may contain no nodes, i.e., empty cluster in various algorithms.

Cluster size: Cluster size may be equal or unequal. In equal, network region is divided into equal size sub-regions or clusters and the size of the cluster is same throughout the network [15]. In unequal clustering, cluster size depends on distance of BS as well as it may depend on other parameters. The smaller distance from BS creates small size cluster and larger distance from BS creates larger size of cluster. It may also differ according to algorithms. Cluster size may be also depending on number of nodes that reside in a cluster. Larger number of member nodes of a cluster creates larger size of cluster while smaller number of member of nodes creates small size cluster.

Intra-cluster communication: Intra-cluster communication actually communication between member nodes and CH of a cluster. It may be direct or multi-hop. Single-hop intra-cluster is suitable for small scale application and multi-hop is needed for large scale application. A suitable intra-cluster communication technique can save energy as well as enhance lifetime and throughput of WSN. The communication mechanism can be contention based, schedule based or hybrid.

Inter cluster communication: Inter cluster communication means communication among clusters or clusters to BS. It may be single or multi-hop based on application. In small scale application, this communication occurs between CHs to BS i.e. single hop and large scale application multi-hop mechanism is preferred for energy efficient data transmission among CHs and BS.

2.3.1.2 CH Properties

Main responsibilities of CH are to collect data from cluster members, combine and forward collected data to BS through direct of multi-hop communication.

Mobility: The CHs can be either fixed or portable. The portable CHs can move for a limited distance. The topology for portable CHs is much difficult than that of fixed. For this type of node, velocity and positioning is the crucial parameter for cluster design [16].

Node type: The discrete CHs across the network can be expensive 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 sensors data and forward them to next CH or BS.

2.3.1.3 Clustering Process characteristics

The characteristics of Clustering process are listed below:

Clustering methods: Two types of clustering methods are in WSN centralized and distributed. In centralized method, a central authority (BS or super node) controls entire operation like cluster formation, CH selection etc. In distributed method, there is no central authority control.

Dynamism: Clustering approach may be in dynamic or static. In dynamic approaches, CH and cluster size and all other related operations are selected based on current situation while in static approach all of these operations are performed regardless of the current network.

Nature: The clustering process can be proactive, reactive or hybrid in nature. Main duties of nodes are to sense data and send it to CH. When CH transmits data to BS continuously, it is known as proactive. CH transmits data BS, if data is sensed and available is known as reactive. 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: CH selection process can be divided into two process, probabilistic and deterministic. In probabilistic, CH selection done randomly and without any previous consideration. It is further divided into random and hybrid manner. In deterministic method, CH selection process done with consideration of various parameters like residual energy, distance to BS, node centrality, node degree, etc.

Deterministic approach is further divided into weight, fuzzy and heuristic based method.

2.4 Related Work on Cluster Head Selection

We have discussed some important cluster head selection techniques from various literature. According to Fig. 2.3, CH selection is divided into deterministic and probabilistic approach. So, this section briefly discusses the CH selection strategy in both categories.

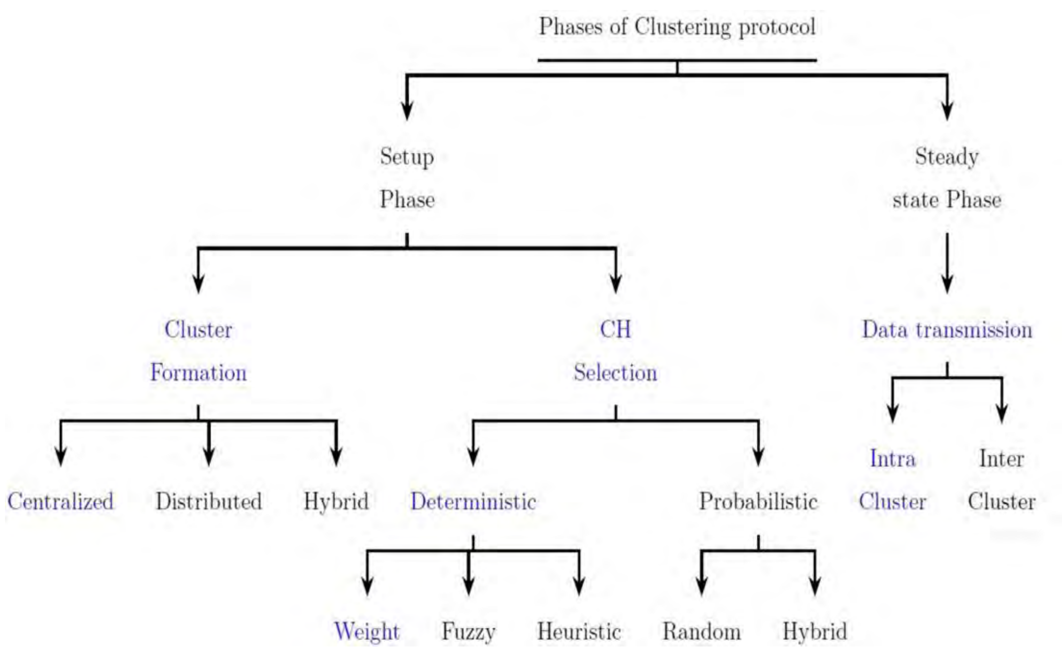


Figure 2.3: Phases of clustering protocols

2.4.1 Probabilistic Cluster Head Selection

The main goals of WSN is to enhance of lifetime and throughput. Cluster head selection using probabilistic approach tries to meet the goal. Some of these algorithms aimed at randomly selecting the CHs. This approach is very simple, has less overhead and converge the network. To ensure more efficiency of clustering, this overhead should be small.

2.4.1.1 Random CH Selection

Low Energy Adaptive Clustering Hierarchy (LEACH) [18] is a pioneer of clustering system. LEACH is a distributed algorithm where nodes can take decisions without any centralized control. In this system, WSN works in several rounds. Each round is divided into two phases: setup phase and steady state phase. In setup phase, network is divided into several number of cluster (k), each cluster contains a CH, cluster formation and Time Division Multiple Access (TDMA) scheduling 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.

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

Where, p is the desired percentage of nodes that become CH, r is 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. When a node is selected as CH, it broadcasts a message to all nodes. According to received signal strength of the advertisement message, non-CH nodes decide to join to a nearest CH and send join request message to that 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. When cluster formation is finished, CHs create TDMA schedule for its member nodes and permits member nodes to go to sleep mode. After knowing each member node‟s time slot, the setup phase is completed then the steady state phase starts. At steady state phase, member nodes of a cluster transmit its sensed data to CH at previously allocated TDMA time slot. After sending sensed data of a member nodes, it goes to sleep mode. This technique avoids collision at intra-cluster communication that increases battery lifetime of WSN 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. As directly communication between nodes to BS is very less, LEACH system saves significant amount of energy which increase network lifetime. But, there exists some disadvantages in LEACH which are as follows:

- CH is chosen randomly in each round and probability of becoming CH of each sensor is same. After several rounds, residual energy level of sensors will be different. So, after several rounds, lower energy nodes have a probability becoming CH. If it happens, then this node will be died out shortly and lifetime of WSN may be degrade.
- The position and number of CHs of each round may be varied in LEACH. LEACH is a random system and it may create unequal distribution of clusters in network. Moreover, some CHs position may be in the center which is good, and some of CHs may be at the near of the boundaries of the cluster which consume huge energy in intra-cluster communication. So, overall lifetime of the network may be decreased.

A centralized clustering approach is offered by same authors in [19], **LEACH-Centralized (LEACH-C)**. In this system, BS is responsible for cluster formation. At first, all nodes send their residual energy and location to the BS. Then BS computes the average energy level and select CH from those nodes whose energy level is more than average at current round. Node having less energy level than average level will not be selected as CH at current round. After selecting CHs, BS will broadcast node ID of CHs to all nodes in the network. Non-CH nodes will join nearest CH as a member node and clustering process is completed.

2.4.1.2 Hybrid CH Selection

Handy et al. proposed deterministic CH selection named **LEACH - Distributed Cluster Head Selection (LEACH-DCHS)** [20]. They achieved increasing lifetime of the network by selecting CH in two processes.

A first approach increases the lifetime of a LEACH network as they included of the remaining energy level available in each node. They achieved new threshold $T(n)_{new}$ by reducing the threshold *T(n)*, denoted in Eq. 2.2(a), relative to the node‟s remaining energy. Therefore, *T(n)* is multiplied with a factor representing the remaining energy level of a node:

$$\boldsymbol{T(n)_{new}} = \begin{cases} \dfrac{\boldsymbol{p}}{\boldsymbol{1 - p \times (r \ mod \frac{1}{p})}} \dfrac{\boldsymbol{E_{n\_current}}}{\boldsymbol{E_{n\_max}}} & \boldsymbol{, if \ n \epsilon G} \\ \boldsymbol{0} & \boldsymbol{, Otherwise} \end{cases} \qquad \boldsymbol{(2.2(a))}$$

where $E_{n\_current}$ is the current energy and E_{n_max} the initial energy of the node. Authors shown a disadvantage: "After a certain number of rounds, the network is in trapped, although there are still nodes available with enough energy to transmit data to the base station. The reason for this is a cluster-head threshold which is too low, because the remaining nodes have a very low energy level". They have overcomed this limitation by including a factor that increases the threshold for any node that has not been cluster head for the last 1/P rounds as equation 2.2(b):

$$\boldsymbol{T(n)_{new}} = \begin{cases} \dfrac{\boldsymbol{p}}{\boldsymbol{1 - p \times (r \ mod \ \frac{1}{p})}} \left[\dfrac{\boldsymbol{E_{n\_current}}}{\boldsymbol{E_{n\_max}}} + \left(\boldsymbol{r_s div \frac{1}{p}}\right)\left(\boldsymbol{1 - \frac{E_{n\_current}}{E_{n\_max}}}\right) \right] & \boldsymbol{, if \ n \epsilon G} \\ \boldsymbol{0} & \boldsymbol{, Otherwise} \end{cases} \qquad \boldsymbol{(2.2(b))}$$

Where r_s is the number of consecutive rounds in which a node has not been cluster-head. When r_s reaches the value 1/P the threshold $T_{(n)new}$ is reset to the value it had before the inclusion of the remaining energy into the threshold-equation.

In Advanced - LEACH (A-LEACH) [21], authors offered a new cluster head selection algorithm improving LEACH by the following equation (2.3). 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.

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

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

$$CS_p = \frac{E_{current}}{E_{n\_max}} \times \frac{N}{k} \tag{2.5}$$

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.

$$T(n) = \frac{k}{N - k \times (r \bmod \frac{N}{k})} + \frac{\boldsymbol{E_{current}}}{\boldsymbol{E_{n\_max}}} \times \frac{\boldsymbol{N}}{\boldsymbol{k}} \tag{2.6}$$

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.

$$T(n) = \frac{k}{N - k \times (r \bmod \frac{N}{k})} + \frac{\boldsymbol{E_{current}}}{\boldsymbol{E_{n\_max}}} \times \frac{\boldsymbol{N}}{\boldsymbol{k}} \tag{2.6}$$

Optimum Cluster - LEACH **(K-LEACH)** [22] offered by Hein and Thein to ensure an even energy load distribution over the whole network. They adopted the optimum number of clusters (K_{opt}) according to equation 2.7 to make a dot-product over a traditional LEACH protocol.

$$T(n)_{new} = \begin{cases} \frac{p}{1 - p \times (r \ mod \frac{1}{p})} \times \frac{E_{n\_current}}{E_{n\_max}} \times k_{opt} & , if \ n \epsilon G \\ 0 & , Otherwise \end{cases} \quad (2.7)$$

Where

$$k_{opt} = \frac{\sqrt{N}}{\sqrt{2 \times \pi}} \sqrt{\frac{\epsilon f_s}{\epsilon_{mp}}} \frac{M}{d^2{}_{toBS}}$$

Here, network area is M×M, optimal number of cluster is k, N is total nodes, d^2_{toBS} is average distance from BS to a node. ϵf_s and ϵ_{mp} are energy needed for transmitting one bit to d_0 an allowable bit error rate in free space and multipath model respectively.

Improved - LEACH (I-LEACH) [23] offered a new protocol for CH selection. In this system, sensor node generates a random number between 0 and 1. Then the random number is compared with improved threshold obtained from (2.8). If the random number is less than the T(n) threshold, then the sensor node is selected as CH node and leaves *G* set to prevent it from reelection as CH node in the current round. *G* is a set of sensor nodes that will determine which of the sensor nodes is not selected as a CH node in the last *1/p* rounds.

$$T(n) = \begin{cases} \frac{p}{1 - p \times (r \ mod \frac{1}{p})} \times \frac{E_{cur}}{E_{avg}} \times \frac{NBR_n}{NBR_{avg}} \times \frac{dtoBS_{avg}}{dtoBS_n}, & if \ n \epsilon G \\ 0 \ , & Otherwise \end{cases} \quad (2.8)$$

where, E_{cur} is current sensor node power; E_{avg} is the average energy of the network in the current round; NBR_n is the number of neighbors for n; NBR_{avg} is the average number of neighboring nodes in the network; $dtoBS_{avg}$ is the average distance of the network sensor nodes to the BS; *and* $dtoBS_n$ is distance of sensor nodes from the BS. One major problem of the method is that; it creates non-uniform distribution of CH which consumed more energy in the network.

Ye M. at el offered a competition based clustering approach **EECS** [24]. In this system, each node becomes a candidate node to be a CH based on a probability and announce its status to its neighbors. Each candidate node, after waiting to receive the announcement from other competitor 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.

Distance Based -LEACH (DB-LEACH) and Distance Based Energy Aware -LEACH (DBEA-LEACH) proposed by same authors Nguyen et al. [25]. In DB-LEACH, if distance of a node from BS is near equal to average distance of nodes from BS, then it may be selected as CH. DB-LEACH used following equation as 2.9:

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

Where

$$dtoBS_{avg} = \frac{\sum_1^N d(i,BS)}{N}$$

DBEA-LEACH offered by same authors that established a new threshold based on distance. In addition, it introduced current energy and initial energy of the node to CH election probability so as to ensure these nodes with higher remaining energy have greater probability to become CHs than that with the low remaining energy. The CH nodes selection directly affected the performance factors of WSN such as load distribution, energy efficiency, and network lifetime. DBEA-LEACH used new threshold according to equation 2.10.

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

Here, E_i is the residual energy of candidate node i at the current round. E_{init} denotes as the initial energy of the node before the transmission.

2.4.2 Deterministic Cluster Head Selection

Deterministic approaches use standard metrics for selecting CHs such as residual energy, node degree, expected residual energy, distance to BS, node centrality, etc. As clusters with elected CHs are more controllable, so it is called deterministic approach. Deterministic approach is further divided into three types: Weight based, Fuzzy based and heuristic based clustering algorithm.

Weight based: 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 based: Where uncertainties are more, fuzzy logic is used. In this method, some fuzzy input such as residual energy, node degree, distance to BS, node centrality, etc., are used and some fuzzy output is received. Fuzzy output may be cluster size and probability of becoming CHs.

Heuristic based: Recently, heuristic based clustering algorithm is being used. 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.

Energy Efficient Clustering Algorithm Based on Neighbors (EECABN):

EECABN is a centralized system proposed by Zhou et al. [26]. It works based on neighbor‟s status. A combined weight factor such as 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 used in this method for electing CHs. Accordingly, the weight is defined as:

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

For a node i, $W(i)$ is its weight, $E(i)$ the residual energy, and $h(i)$ the distance to the sink node. Within the area of radius R, other nodes can communicate to node i successfully with enough signal level, and the radius R is called neighbor range of i. B is the set of neighbors for node i, node j one of the neighbors within neighbor range from node i, and $d(i,j)$ their distance. The variable $h_{\max}$ is the longest distance from the sink node to the farthest node in the whole network, and $E_{\max}$ is the highest node energy level. ε is a constant constrains the lower limit of ratio of $\frac{E(j)}{E_{max}}$ or $h(i)$, while a node has very low residual energy or very short distance to the sink node. In this method, nodes are divided into two categories. The nodes which consists more than average energy is known as strong and CHs election process done from these nodes. Those nodes have less than average energy known as weak and these nodes remain absent from CHs election process. 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)

Abdulsalem et al. [27] proposed a new data aggregation algorithm **Weighted - LEACH (W-LEACH)**. This algorithm is able to handle uniform and non-uniform networks. Authors 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.

$$W_i = \begin{cases} e_i \times d_i; & if\ d_i > d_{thres} \\ d_i; & Otherwise \end{cases} \quad (2.12)$$

where $d_i = 1 + \frac{\text{number of alive sensors in range r}}{n}$ for 1 α i α n, and d_{thresh} is a density threshold to define the set of sensors in very low density areas. So, all the nodes of a cluster not needed to activate and take part in each round of communication, like LEACH. As all nodes need not to send data to CH all time, so network lifetime will be increased. 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.

$$d_{CH} = \frac{\text{number of alive sensors in Cluste}_i}{\text{total number of alive sensors}} \quad (2.13)$$

LEACH - Medium Access Control (LEACH-MAC) [28] protocol is designed to mitigate the randomness problem by restricting the number of cluster head advertisements. Optimal number of CH (k) is same as k-LEACH. At starting of CH selection process, a variable CH_{heard} initializes
to 0 and is incremented by 1 if it receives a CH advertisement message.

After passing through the threshold function, nodes select a uniform random time in the interval 0 to total_adv_time, where total_adv_time is the total time required for cluster head transmission and reception. Now the selected time R_t is divided by the current (residual) energy so that higher energy nodes can send the advertisement earlier than low energy nodes. Time of sending CH advertisement as following equation 2.14:

$$t_{\text{adv\_CH}} = \frac{R_t}{\text{Current Energy}} \quad (2.14)$$

At the time t_{adv_CH}, node checks the number of advertisements heard by itself through the CH_{heard} variable. If the value of CH_{heard} variable is less than optimal value, i.e., 5 then it will send the cluster head advertisement otherwise declare itself as 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. Authors have tried to 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.

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

where, λ is a random number between 0.9 to 1.0; T_{CH} represents 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.

Essa. A. et al [32] offered another weight based protocol Dynamic Weight based clustering **Algorithm (DWCA)** considering number of neighbor nodes, distance to BS and remaining energy for selecting candidate cluster head node using equation 2.16.

$$W_v = w_1 \times \, neighbors + \, w_2 \times distance\ to\ BS + w_3 \times remaining\ energy \qquad (2.16)$$

Each node calculates W_v according to this equation and broadcasts this value to its neighbors. Among these neighbors, the node whose W_v is the highest, selects itself as a candidate CH. Each initially selected candidate CH competes for the final selection of CH within its range (R_{comp}); among the range those nodes that have highest weighted value are 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. 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. 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 protocols [35, 36] use extra relay node to avoid long distance communication but selecting the relay node increases complexity.

Amit [51] proposed "An Energy Efficient and Balanced Clustering Approach for Improving Throughput of Wireless Sensor Networks" and same author [50] offered an **Efficient Cluster Head Selection Strategy for Provisioning Fairness (ECHSSPF)** in Wireless Sensor Networks which is a faired technique to select CH using three parameters. Authors have used number of neighbor nodes (NBR), remaining energy (RE) and one hop neighbor information to select CH. When nodes are randomly deployed into network, it first selects its radius to find its neighbor by the Equation 3.4. After calculating radius, each node sends „hello" message to find its neighbors 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 BS. BS then sort the weighted value (WGTV) which is calculated by the following equation 2.18:

$$\mathrm{WGTV}_{vi} = \mathrm{RE}_{vi} + \alpha \times \mathrm{NBRCNT}_{vi} \tag{2.18}$$

To incorporate residual energy with neighbor nodes, authors considered a constant α and set its value 0.1 in their analysis. After sorting WGTV, the node which have highest WGTV, BS selects it as a cluster head and adds it into CH list. When base station goes through the second highest node, it checks its neighbor list with selected CH list. If one of its neighbor is already in the selected CH list, then BS considers it as a normal node otherwise adds it into CH list. By sampling each nodes properties and checking its neighbor node, finally base station finds a balanced set of cluster head. BS then informs the determined CH node by sending packet.

This CH selection system ensure fairness and balanced situation that cover every region of the network. It cannot create balanced clustering due to back transmission problem. Let discuss this problem by an example. In figure 2.4, a network is divided into five clusters. In one of them, let CN-1 and CN-2 be two candidate nodes to become CH and assume both nodes have same residual energy. CN-1 has more neighbor nodes compared to CN-2. As [51] CN-1 will be CH of that cluster. All member nodes then transmit data to CN-1. CN-1 will collect, aggregate and transfer that data to BS. But, if CN-2 were selected as CH, then over all path length (including distance between member nodes to CN-2, CN-2 to BS) will less than that of CN-1. So, more energy of CN-1 will be consumed compare to CN-2 and cluster will be unbalanced. Considering these unbalanced condition, we have tried to select CH to make energy efficient clustering which is described in next chapter.

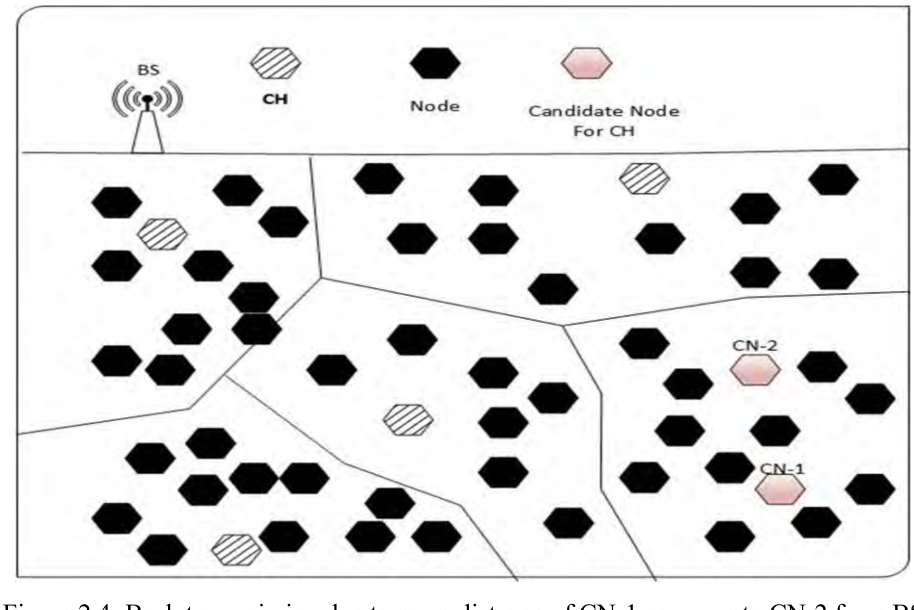


Figure 2.4: Back transmission due to more distance of CN-1 compare to CN-2 from BS

Protocols we described above are summarized with comparative study in Table 2.1. This table shows year wise comparison among LEACH and it's successors considering most important five parameters: clustering scheme, overhead, energy efficiency, scalability and coverage.

2.5 Related Work on Cluster Formation

After selection CH, cluster formation is the next step. Non-CH nodes join to CH is the main task of cluster formation. Many authors used traditional method as LEACH. In this section, we have described some cluster formation approaches.

2.5.1 CH-Leach

An Energy Efficient and Adaptive Clustering for Wireless Sensor Network (CH-Leach) using Leach Protocol [49] based on k-means procedure. At first, authors generate *k* pointes cluster center, the number of the cluster required k, then set the data points. In this case, the data points will assign to location of the nodes, in every cluster area required. The cluster center (centroid) in each area will assign to nearest node by calculating the mean value of the nodes location (data points) in each cluster. Then, this node will become the cluster-head if it has enough energy, means above the threshold set. At last, repeat the steps when assigning node that is near to the cluster center not able to act as cluster head.

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	Unbalance
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
ECHSSPF	2017	Centralized	High	High	High	Balanced

However, k-means procedure suffers from empty cluster and unequal cluster size problem [16]. [13] uses k-means algorithm for clustering. CH selection is made by Gauss elimination algorithm. This approach also suffers from unequal cluster size. Due to unequal cluster size, uneven load distribution occurs in the network. Because a CH holds more member nodes (MNs) will chomp more energy, and dies faster in compare to another CH holding less MNs [14]. Comparative analysis of various protocols is summarized in the Table 2.1.

2.5.2 EEBCA

An Energy Efficient and Balanced Clustering Approach (EEBCA) for Improving Throughput of WSN proposed by Amit [51] offered a clustering methods. According to this method, 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 according to 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 allocates (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. From this clustering approach, it is seen that all non-CH nodes will join to own nearest CH according to RSSI value. It may be a cause of unbalanced clustering due to non-uniform distribution of nodes in the network. CHs with high density nodes area will create cluster with a large number of nodes while CHs with low density nodes area will create a cluster with a tiny number of nodes. CHs with large number of member nodes will consume huge energy to collect data from its members and send it to BS. So, this CH may be died out shortly. So, lifetime of the network will be degraded. Again, when any node will act as CH, it also has an area to collect data. But, there is no information how to collect CHs own data in this technique. So there is a lack of data collection in this method as well as throughput may not be as expected.

2.6 Related Works on Intra-Cluster Communication

We have discussed earlier that in LEACH protocol whole operation completed in two phases setup phase and steady state phase. After selecting CHs and constructing clusters, CH allocate time slots for its member nodes and broadcast it to all members. At steady state phase, member nodes send it collected data to respective CH at allocated time slot. A lot of successors of LEACH protocol work over setup phase but they consider steady state phase as same as traditional LEACH protocol. For large scale application this type of mechanism is efficient than the contention based. But, in small and medium scale application many slots may be idle while another node may wait for its slot for data transmission. Some researchers work over data communication phase in clustering. Here, we have discussed about some Intra-Cluster Communication Technique.

2.6.1 CBICCT

Amit et al. [37] offered a **CSMA based intra-cluster communication technique (CBICCT)**. This technique used preamble for sensing channel. If channel is free, then send data otherwise wait for a random time. After that random time waiting, the node will send preamble for sensing again and so on that shown in figure 2.5.

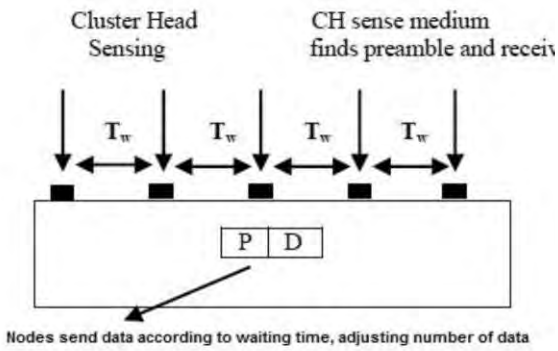


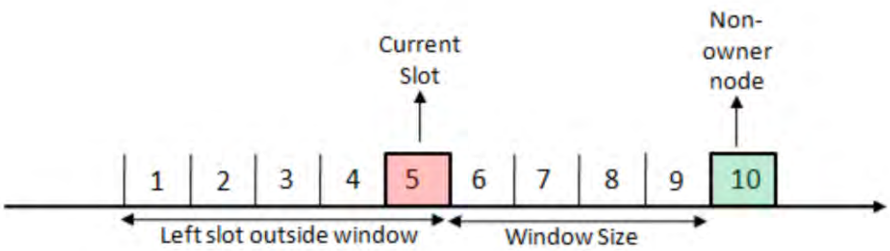
Figure 2.5: Preamble based CSMA (Redrawn from [15])

But, long preamble creates extra control packet overhead and increase delay. Again, if more than one nodes want to send different amount of data, and if one node with small amount of data send preamble and get channel free, then this node will send that small amount of data while other nodes with more data are waiting. So overall sending data or packet will be decreased. So throughput of the network becomes degrade. We have tried to overcome this limitation in our protocol.

2.6.2 Capture Based Intra-Cluster Mechanism

Amit et at. [51] offered Capture Based Intra-Cluster Mechanism. Basic idea of capture effect is, when multiple sender transmits packet simultaneously, then the receiver can decode the comparatively high power packet consider the window size as same as FAWAC[44], 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 this 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 possibility to send its data successfully.

In Figure 2.6 and 2.7, 10 member nodes are considered who want to send their data to CH and window size is 4. Basically, this mechanism consider $((\lfloor \frac{n}{2} \rceil - 1))$ 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 to transmit, then it does not send its data within non owners slot. According to Figure 2.6, 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 waits 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 2.7, 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 5th slot.



Th

e

sys

Figure 2.6: When current slot is at the left of the window(Redrawn from[51])

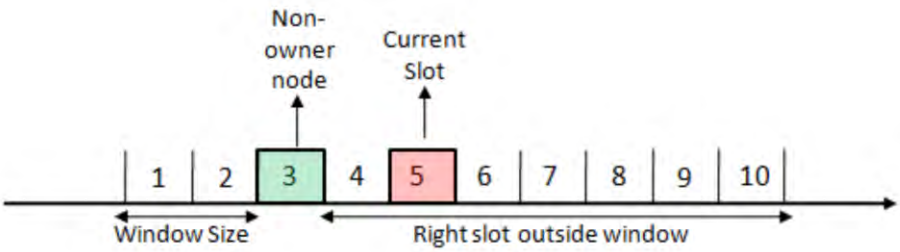


Figure 2.7: When current slot is at the right of the window (Redrawn from[51])

tem is good for transmitting data of non-owner data into idle slot. But problem is, when multiple non-owner nodes has different data. In this technique, no description is available about multi-level data of multiple non-owner nodes. Suppose an idle slot is available but more than one non-owner nodes have data to send. Then according to this technique, all non-owner node will send data with same power level. If buffer receives small amount of data of one non-owner node, while another non-owner node is waiting to transmit large amount of data, then overall throughput will be degraded. So, there is a scope for research to transmit large amount of data of non-owner nodes and enlarge throughput of the network.

2.7 Summary

We observed from the literature review that a good number of study/ research have been done about CH selection, Cluster formation and Intra-cluster communication technique in recent years. We also observed that, inspite of existing some good researches, there exists some lacks of proper CH selection, Cluster formation and Intra-cluster communication technique. We have tried to overcome these limitations which described in next chapter.

CHAPTER 3
Proposed Methodologies for Balanced Clustering

The goal of the proposed protocol is to extend lifetime as well as throughput of WSN. For these, we have tried to select Cluster Heads (CHs), cluster formation and efficient intra cluster communication technique in a balanced way. Getting clustering advantage is impossible from unbalanced situation. Proper selection of cluster head and cluster formation technique can help to create balanced cluster and it can cover every region of the network. Moreover, efficient intra-cluster communication can increase the number of packets sent to BS; it also decreases the per packet delay. 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

We can represent our network model by a graph G. Let G= (V, B, E) be an undirected graph where V is the set of nodes, B is the base station which is supposed to have unlimited resources and E is the set of edges (u,v) ϵ E if both u and v are in the communication range of each other. Let the set of nodes those are directly connected to v_i (v_i"s neighbors) is denoted by N_{vi}. G mirrors a network topology. Furthermore, nodes have ability to communicate directly to base station by increasing its power when necessary. Nodes are randomly deployed and create a stationary network. After deployment of nodes, network will be stationary. Traffic is periodic, data packet produced by nodes has same length and has a fixed rate. Cluster Heads (CHs) receive data from nodes and aggregate them into a single message then transfer to Base Station (BS). A clustering Scheme S is defined as a set of cluster $S = \{CH_i, m_1, m_2, m_j\}$ that cover the network (G). Every 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:

- All nodes are randomly organized over the network.
- All nodes are same type, i.e., all nodes consist of equal communication abilities and equal initial energy.
- Energy level of the nodes can differ according to communication distance.
- The BS is fixed with enough resources.

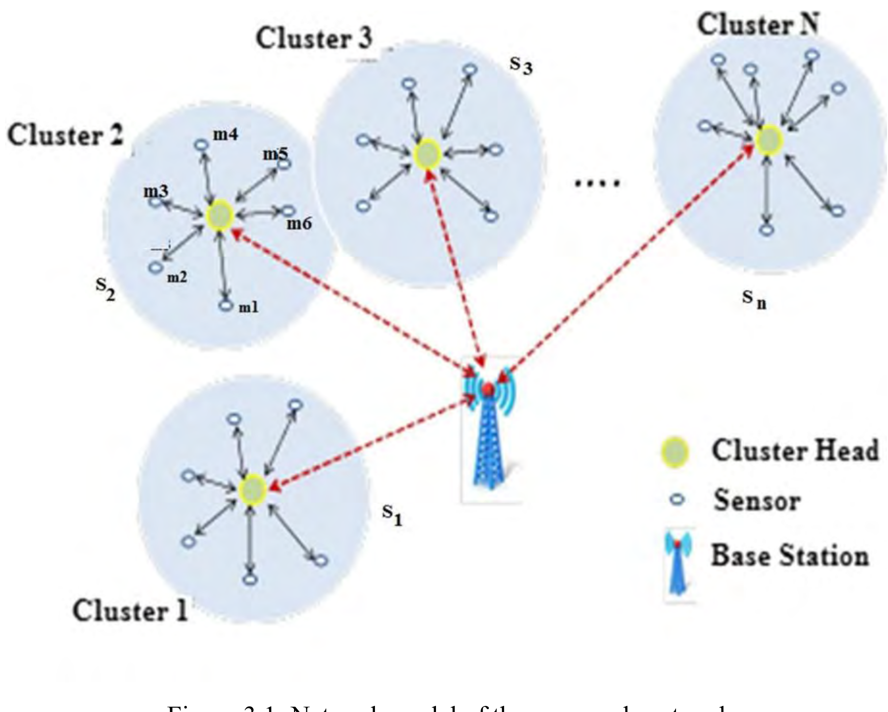


Figure 3.1: Network model of the proposed protocol

3.2 Radio Model

We considered a radio model [23] to analyze energy consumption. This model is first order model which is shown in Fig. 3.2.

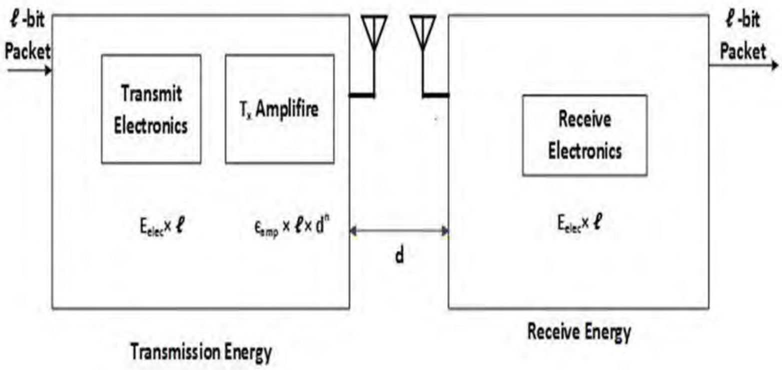


Figure 3.2: Network energy model

Let distance is d, number of bit is l then energy needed:

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

Here, E_{elec} is energy needed for processing one bit, ϵf_s and ϵ_{mp} are energy needed for transmitting one bit to d_0 with an allowable bit error rate in free space and multipath model respectively. d_0 is threshold value where:

$$d_0 = \sqrt{\frac{\epsilon f_s}{\epsilon_{mp}}} \quad (3.2)$$

$$E_{RX} = E_{elec} \times l \quad (3.3)$$

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

3.3 Protocol Description

General overview of our proposed protocol is depicted in Figure 3.3. Designing the proposed protocol is divided mainly into three parts. I) Cluster Head (CH) selection and ii) Cluster formation and iii) Intra-cluster communication technique. We used intra cluster communication technique by capture based intra-cluster mechanism. We used various technique for doing those process. Whole procedure will be done in setup and steady state phase.

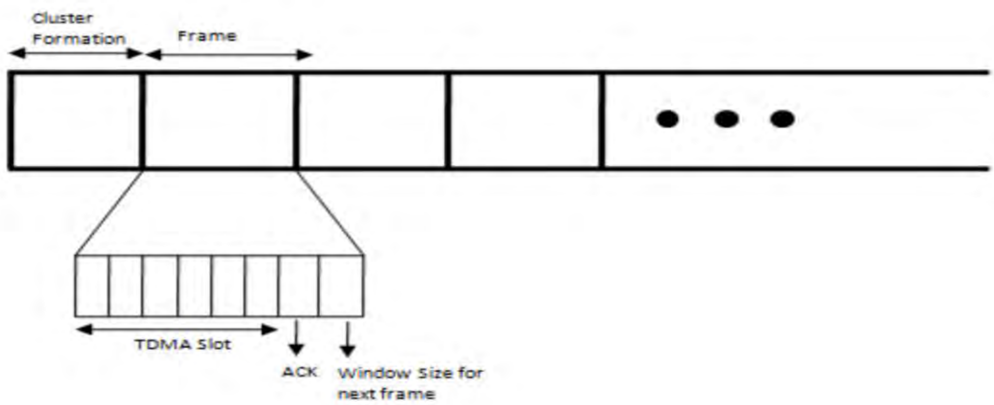


Figure 3.3: Overview of the proposed protocol

3.3.1 Setup Phase

In setup phase, the network is divided into several areas called cluster. Cluster formation can be generally classified into centralized and distributed techniques. In centralized clustering, the control message for clustering computation is received from a central base station (BS) based on information collected from all the sensors in the network [40]. In the distributed approach, the decisions related to formation of the clusters are made by the sensor nodes without the use of a central entity (or BS). In our research work, we used centralized method where BS has enough resources for any computation.

3.3.1.1 Cluster Head Selection

Our proposed protocol is a centralized protocol, where base station is responsible to select cluster head according to the algorithm. When network will be formed by randomly deployed N sensor nodes, at first base station will calculate radius for every node‟s neighbors and optimum number of cluster of network according to the equation 3.4 and 3.5 [19] respectively. BS will also calculate neighbor set for each node within radius R.

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

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

Here, network area is M×M, optimal number of cluster is k, N is total nodes, d^2_{toBS} is average distance from BS to a node.

We have used four parameters for CH selection. First, Residual Energy (RE) has been considered. If more than one consecutive or in communication range candidate nodes (to be CH) have highest or same or nearly same residual energy, then more than one consecutive nodes or in communication range of previously selected CH may be selected as CH and unbalanced situation may occur in network. To overcome this problem, we have considered Number of Neighbor nodes (NN) with an adjustable value of α. We also considered one hop neighbor information that will restrict more than one CH in one cluster. At last, we consider distance from BS to CHs (nodes) in reverse order with another adjustable value of β. It will help to select comparative nearer nodes to BS as CH and save energy to transmit packets to BS as distance from CH to BS is less than others candidate nodes. All nodes will send residual energy as well as position to BS. Then, BS will calculate distance of all nodes from itself, neighbor set and number of neighbor (NN) of each node within radius R. Base station will calculate weighted value (WV) for each node according to equation (3.6).

$$\mathrm{WV}_{v_i} = \mathrm{RE}_{v_i} + \alpha \times \mathrm{NN}_{v_i} + \frac{\beta}{\mathrm{Distance}_{v_i}} \qquad (3.6)$$

Where α and β are adjusting factors that will adjust number of neighbors within radius R and distance of each node with residual energy. In our analysis we have used α=0.1 . This value has been considered because of maximizing lifetime and throughput of the network. We have used cumulative distribution function (CDF) to select α value which has been described in section 4.3.2.1 Since our system is a centralized system, it has enough computational power. It will check all possible combination of β and find out the β value for minimum energy requirement of the network at a particular round. For example, at round 10 for β =75, required energy is 0.33855 nJ where as if we can select β =70, then the required energy is 0.342329 nJ which is greater than of β =70. So, if we will select β value equal to 75 and required energy will be minimum. So, network lifetime will be increased. Then, BS will sort the WV of n nodes in descending order. Highest WV valued node will be selected as CH, then next highest WV valued node will be considered. If this node is in the area of previous selected CHs, then it will be considered as normal node. By this method, when all nodes will be evaluated then we will get a CHs list. Flowchart (Figure. 3.4) and Algorithm of whole task is shown in Algorithm-1.

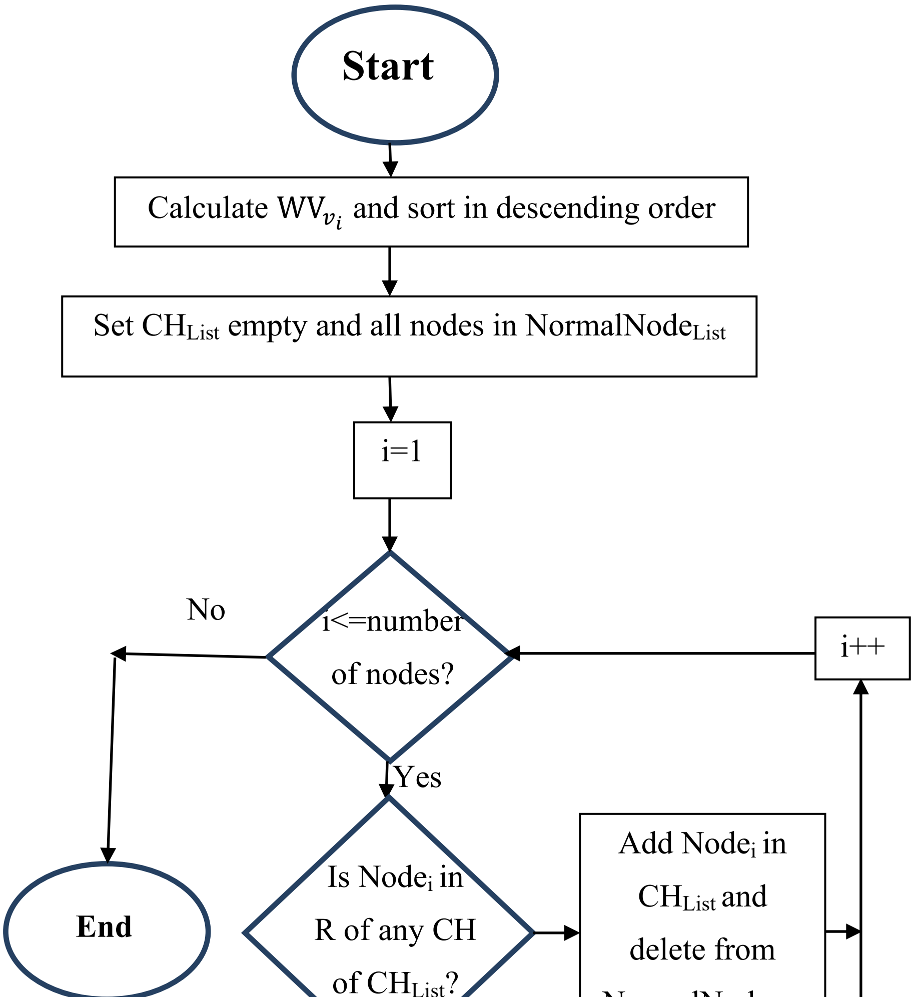


Figure 3.4: Flowchart of CH selection process

Algorithm 1 CH selection algorithm

```
Input:    Residual energy and position of all nodes
Output: CHs is the cluster head set

    NN = 0
    NBR=∅
    for ∀V ∈ G do
        Calculate R by using Equation 3.4
        Update NBR & NN
    end for

    CHs=∅                          //Initially set of CHs is empty
    for (each vertex vi ∈ V) do
        calculate its weight WVvi by using E.q. 3.6
    end for
    SORT (WVvi)           //sort the weighted value in descending order
    for i=1 to N do
        if NBRvi α  CH is ∅ then
            CHs α vi  ∪ CHs
        else
            vi is a normal node
        end if
    end for
```

3.3.1.2 Cluster Formation

The main aim of our clustering method is to create energy balanced network which distributes the energy load balance among the network. Eventually, lifetime of the network will increase. After selecting CH, as describe in section 3.3.1.1, we have a marginal border line for those CH whose position are near the border of the network.

We have drawn a connection line among those CHs which is called marginal border [Fig. 3.5 (a)]. We set the minimum node threshold $Node_{min}$ which depicts the lowest number of nodes can be contained by a cluster. Maximum node threshold $Node_{Th}$ which depicts the maximum number of node can join in a cluster. Calculation of $Node_{Th}$ and $Node_{min}$ are shown in equation 3.6 and 3.7 respectively.

$$Node_{Th} = \frac{\text{No.of alive node}}{\text{No.of cluster}} \tag{3.6}$$

$$Node_{mim} = Node_{Th} X q\% \tag{3.7}$$

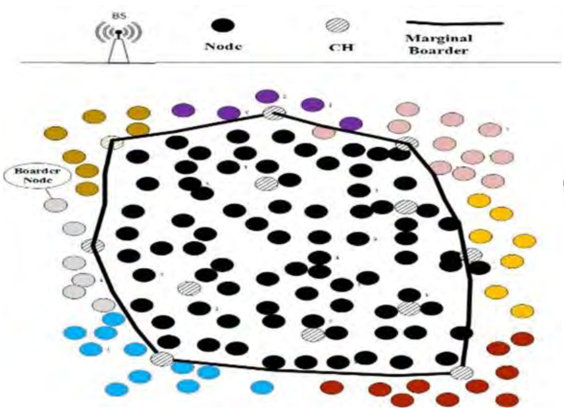


Figure 3.5 (a): Initial network with border line and marginal nodes joining

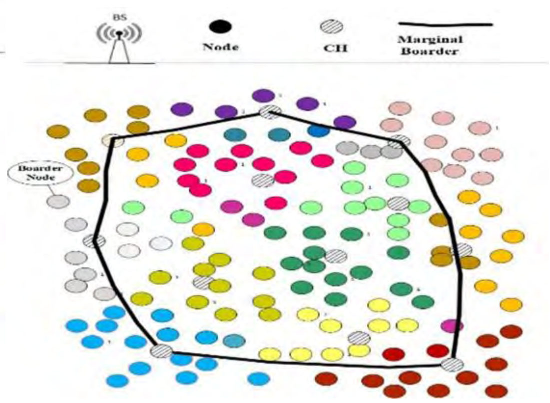


Figure 3.5(b): Final cluster formation

Now, nodes outside of the marginal border are consideres as a Central Border nodes (CBN). We have joined these CBN with their nearest CH. Then, we have considered the nodes inside the marginal border as Internal Nodes (IN); join IN to its nearest CH until the number nodes of that CH is less than $Node_{min}$. Next, we have considered the nodes which have distance almost equal from two or more CH. Actually, these nodes have multiple options for joining with the CH for cluster formation. We have called these nodes as Balancing Nodes (BN). Then for each BN, we choose the nearest CH for which the number of node less than $Node_{Th}$ and the BN joins with that CH for cluster formation. Otherwise, the BN finds the next nearest CH to join for cluster formation. Thus, we create approximately equal size cluster among the network. If any nodes of IN remains free, then we join those nodes to nearest CH. Fig. 3.5 (a) shows initial network with border nodes and joining of border nodes to clusters. Fig. 3.5 (b) shows the final cluster formation where every node joins to a cluster ensuring energy balanced. Each color of Fig 3.5 (a) represents a cluster. After a certain number of rounds, many nodes will be dead. Then clustering approach will reset, otherwise network will become unbalanced. Steps of the protocol is described in algorithm 2 and flowchart in Fig. 3.6.

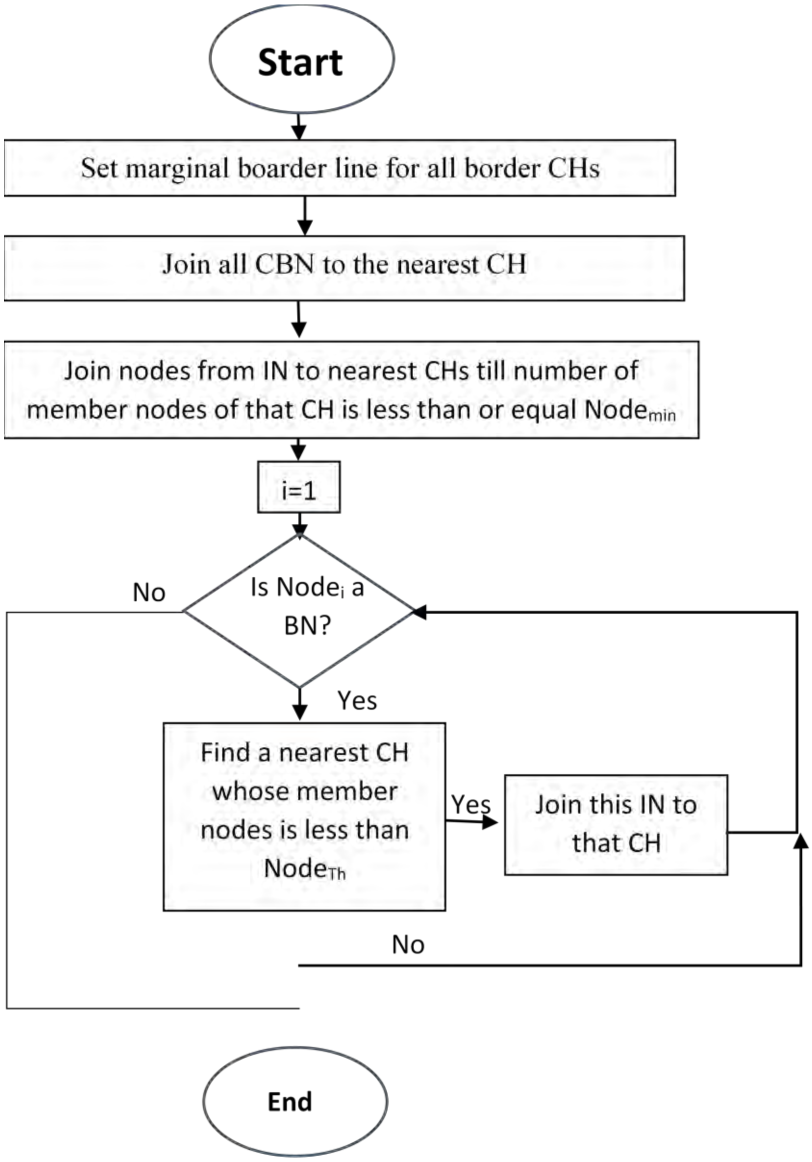


Figure 3.6: Flowchart of cluster formation

Algorithm 2 Cluster Formation

```
Input: CHs //Set of cluster heads
       Nodes // Set of nodes
Output: Each CH contains a cluster

for (each CH_i ∈ CHs) do
    if (CH_i is a border CH)
        Draw line to neighbor border CHs
    end if
end for

for (∀node_i ∈ Nodes) do
    if (node_i is CBN)
        join node_i to its nearest CH
    end if
end for
for (each CH_i ∈ CHs) do
    Join maximum Node_min nearest nodes to CH_i within R
end for
    for (∀node_i ∈ Nodes) do
        if (node_i is IN and not join to any CH)
            Find nearest CH
            If (number of member nodes of this CH< Node_Th
                join node_i to this CH
            else
                Find next nearest CH without overlapping with
                number of member nodes of the CH< Node_Th
                join node_i to this CH
            end if
        end if
    end for

for (∀node_i ∈ Nodes) do
        if (node_i does not join to any CH)
            join node_i to nearest CH
        end if
end for
```

All information will be stored to BS. BS will send a message to all CHs by performing CSMA/CA. The nodes which received this message will treat itself as a CH. This message also contains member nodes information (nodes ID). CH will understand those nodes are member node and will assign TDMA time slot for those member nodes. All CHs will be treated as a member nodes itself for data collection of its own area. BS will send another message to each node other than CHs by performing CSMA/CA. This message contains only a CH ID and respective TDMA time slot. After receiving this message, those nodes will understand that node ID is the CH of itself and TDMA time slot is for message sending to that CH. Moreover, CH will treat itself as a member of the cluster. After performing these tasks, the whole network is divided into several clusters equal to number of CH and completes setup phase. After completing setup phase, network enters into steady state phase.

3.3.2 Steady State Phase

In this section, an intra-cluster communication technique without using extra control and various powered data is introduced which adopts the slot as per traffic condition. This technique will be captcha based. The fundamental concept of capture effect is, while multiple senders send packet concurrently, the receiver can decode the relatively high powered packet. The term capture effect was first introduced in 1976 by Metzener et al. [42]. Capture effect was first exploited in WSN after 29 years of capture based ALOHA network via 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. We used window size as same as FAWAC, then adapted the window as per traffic situation and number of data. We used several stage for proposed method of Intra-cluster Communication Technique as described blew:

3.3.2.1 Power Level Adjustment

Within any cluster, each member node has its own slot for sending the sensed data. In our proposed technique, the owner of each slot always sends data with highest power and non-owner nodes will try to send data with various power. In this system, a non-owner node will adjust power of its data according to amount of data as equation 3.8.

$$Power_{Data} = Power_{High} \times DPP_{NWN} - Power_{Adjust} \tag{3.8}$$

Here $Power_{Data}$ is the power of data, $Power_{High}$ is the normal power to communicate, DPP_{NWN} is the percentage of data packet of any nodes; for owner‟s node always DPP_{NWN}=100% if data is available. $Power_{Adjust}$ is the adjusting power to differ the power of data with owner‟s node and non-owner‟s node. $Power_{Adjust}$=0 for owner‟s node and $Power_{Adjust}$=10 for non-ower‟s node. If a non-owner node DPP_{NWN}=20%, then it will send data with very low power (20-10) 10% of its normal (highest) power.

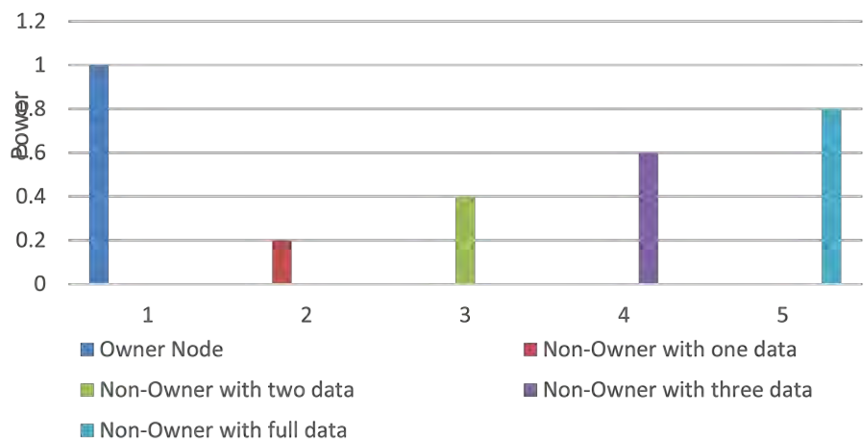


Figure 3.7 (a): various powered data (without idle slot)

Figure 3.7 (b): various powered data (with idle slot)

If this node‟s $Power_{Adjust}$=40%, then it will send data with higher power compare to previous; it will send (40-10) 30% of normal powered data. If the node‟s

$Power_{Adjust}$=60%, then it will produce more powerful data compare to previous and so on. If this node has full data, then it will send the highest powered (90% of its normal powered) data but less than owner node. If there is no traffic from the owner, then the non-owner node with highest powered data have the possibility to send its data successfully. Thus, throughput of the whole network will be increased. All nodes (if has sensed data) of a cluster will send data like this which is illustrated in Fig-3.7. Fig-3.7 (a) shows that owner node will send highest powered data (1) while non-owner node with 30% data is sending less powered (0.2); non-owner node with 50% data is sending with higher powered (0.4) and so on. But, non-owner node with full data is sending much higher (0.9) powered while owner-node sending with highest power (1). In this situation, CH will receive owner-node‟s data. If owner-node has no data but other non-owner nodes have data as like Fig-3.7 (b), then CH will receive data from non-owner node‟s data (with much higher powered). When multiple non-owner nodes will send multiple data and owner node has no data, then CH will receive highest powered data.

3.3.2.2 Time Slot Adjustment

We have also used another technique time slot adjustment, for various traffic of non-owner node. Let us discuss this technique with an example Assume one cluster has 10 nodes and they are trying to send data to their CH and window size is 4. We used $(\lfloor\frac{n}{2}\rfloor - 1)$ as window size. We divided slot into two categories: „left side slot outside window‟ and „right side slot outside window‟ which is shown in Fig 3.7 (a) and (b) respectively. In clustering system, member nodes send data to CH in TDMA schedule and CH is responsible to collect data from members, aggregate them and send to BS. If any node has data less than buffer size and it is in "left slot outside window" condition, then it will not send data; it will wait for its owner slot. This situation is described in Fig 3.8 (a). Here, current slot is 5 and node 10 has data to send; node 10 is non-owner. If node 10 has more than one data into buffer, then it

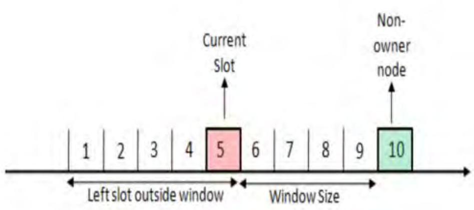


Fig. 3.8 (a): Left side slot outside window

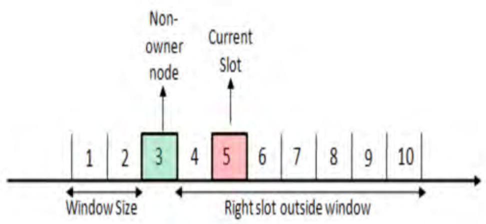


Fig 3.8 (b): Right side slot outside window

will send with adjust power level of that data according to number of data; otherwise it will wait for its own slot. On the other hand, if the current slot is right outside the window as like Fig. 3.8 (b), then it will send its data to CH by adjusting power level of the data. Fig. 3.8 (b) illustrated this situation. Here, current slot is 5 and node 3 has data to transmit; node 3 will transmit data without waiting and adjusting power level of data according to number of data. In our protocol, multiple non-owner nodes can send data to CH by adjusting power level of sending data. If more than one nodes send data to CH, CH will receive comparatively higher powered data. This will not only increase the number of packets sent to BS but also decrease the per packet delay.

3.3.2.3 Preamble Based CSMA

We have used preamble based CSMA technique to avoid collusion of data at CH. For this, we send a preamble for sensing channel. But, we send this preamble using a technique where if non-owner node has data to send, it will wait according to amount of data and then send preamble. If amount of data (D_n) is full or highest, then it will send preamble immediately. If D_n is very small, then it will wait for highest time then send preamble. According to D_n, it will wait and send preamble for channel sensing. The waiting time (W_t) will be according to equation 3.9.

$$W_t = \frac{T_w}{D_n} \tag{3.9}$$

Here T_w is the time slot of TDMA.

For example, let $T_{w=}1$ and a non-owner node D_n=40%, then it will wait for a time period W_t=0.25 unit, if D_n=50% then W_t=0.20 and for D_n=10%, W_t=1 or maximum time. By this technique, non-owner node with highest amount of data will get better

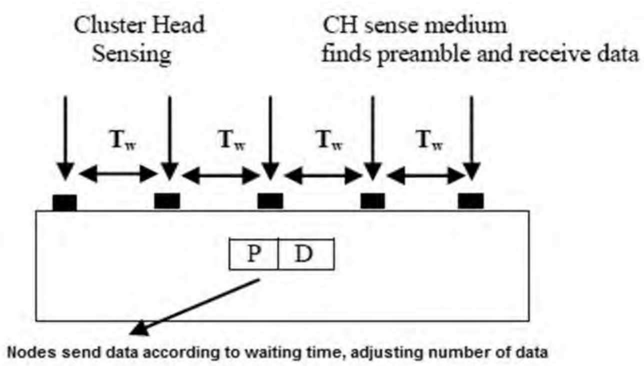


Fig 3.9: Preamble based CSMA (Redrawn from [37])

chance to send. It will increase data sending as well as throughput and reduce collusion of data. This scenario is illustrated in Fig. 3.9. This procedure is summarized into algorithm 3 blew:

Algorithm 3 Slots re-use

Input: C_s =Current Slot, S_{bw} =Slot behind the window, D_{bf} = Number of packet into the queue, Saw= Slot after the window,
Output: Sends data or Waits

When data generate outside the window
  if (C_s is S_{bw} && D_{bf} >= 2) then
    Adjust power of data according to equation 3.8
    Send preamble according equation 3.9 to Sense channel
    if (channel is free) then
      sends data as non-owner node
    end if
else if (C_s is S_{aw}) then
    Adjust power of data according to number of data
    Send preamble according equation 3.9 to Sense channel
If (channel is free) then
      sends data as non-owner node
else
    waits for its own slot
end if

3.4 Summary

In this chapter, a balanced clustering protocol for wireless sensor networks has been proposed. We have divided our procedure into three parts. At first, we have selected a set of cluster heads; then created clusters ensuring energy balanced and at last, adopted a capture based intra-cluster communication technique. We have used three steps (Power level adjustment, Time slot adjustment and Preamble based CSMA) for a better Intra-Cluster Communication. In this protocol, BS takes the decision of cluster head selection based on the residual energy, number of neighbor nodes, one hop CH information and distance between BS and CHs. Along with this, one intra cluster communication mechanism is proposed with adaptive window selection.

CHAPTER 4
Results and Performance Evaluation

4.1 Introduction

In this chapter, we described the performance of the proposed protocol evaluated through simulation study. For performance analysis, we have compared our protocol with a pioneer clustering protocol (LEACH-C) [19], LEACH - Medium Access Control (LEACH-MAC) [28] and An Energy Efficient and Balanced Clustering Approach for Improving Throughput of WSN (EEBCA) [51]. We have used OMNeT++ simulator [47] and considered a network with 50 and 75 sensor nodes which are deployed randomly. Network area is 200X200 meter^2. The Base Station (BS) or Sink Node is placed on the border of the network. We have considered simulation parameters as like Table 4.1.

Table 4.1: Network Parameters

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

According to [48], lifetime of WSN is the total amount of time over which the network remains operation. The lifetime actually depends on each sensor node‟s lifespan. Node lifespan actually depends on two factors: firstly. how much energy it spends over a round and secondly, how much energy needed to receive, aggregate and send data. The lifetime is individually corresponding to the measure of data received from the network. So, the lifetime of WSN is the amount of time that the sensor nodes will stay alive in the network. Lifetime also is measured as First Node Dies (FND), Tenth Node Dies (TND) and Last Node Dies (LND) metrics of the network. FND means the time needed from start of network operation till first node dies, TND is the time from first operation to tenth node dies and LND is the time from first operation to last node dies. For simulating Intra-Cluster Communication technique, we considered parameter as like Table 4.2.

Table 4.2: Parameters for Intra-Cluster Communication

Parameter	Value
Area	200 meter× 200 meter
Nodes	50, 100, 200, 300, 400 and 500
CH	10% of Nodes
Possible idle slot	10%
Rounds	100, 200, 400, 600, 800 and 1000

Number of clusters: In each round, network is divided into several areas called cluster. Network lifetime can be affected based on number of cluster. Too much or too less number of cluster can decrease lifetime of the network. Number of clusters also can be affected by Cluster Head (CH). Generally, number of cluster is equal to number of CH. We used optimal number of cluster according to Equation-3.5 for increasing network lifetime.

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

4.2 Simulation Environment

Fig. 4.1 and 4.2 presents our proposed network structure with 50 and 75 nodes respectively. Here nodes are randomly deployed and base is station placed on the outside/border of the network. Each node contains a unique ID. From this structure, we used various parameters as shown in Table 4.1. Sample code of this simulation is attached in the Appendix.

4.3 Simulation Results with Various Metrics

At first we have simulated Intra-Cluster Communication Technique in terms of sending packets per rounds and improvement using idle slots per rounds. Then, we have compared simulation results of proposed whole techniques in terms of energy consumption vs rounds, rounds required for total energy dissipation of FND, TND and LND, Remaining energy vs Rounds, Number of packets send to BS and CH vs Rounds, Dead nodes vs Rounds, Alive nodes vs Rounds.

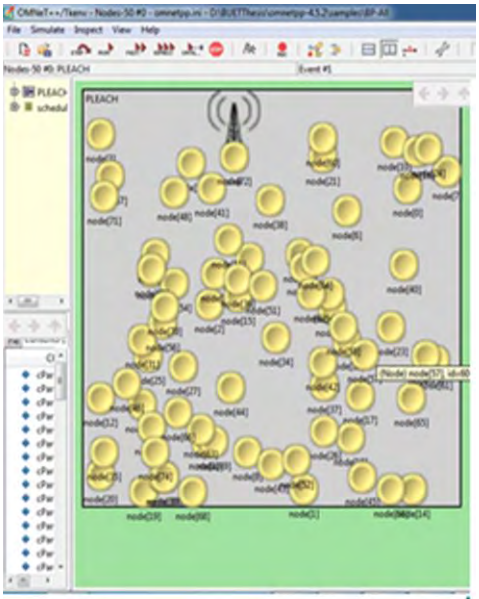


Figure 4.1 Simulated Network structure in OMNeT++ (for 50 nodes)

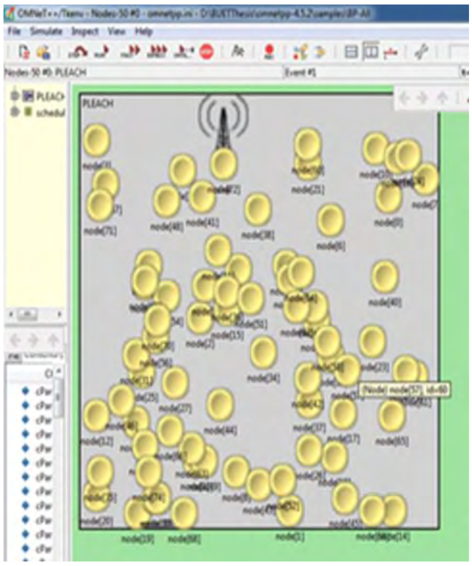


Figure 4.2 Simulated Network structure in OMNeT++ (for 75 nodes)

4.3.1 Simulation of Intra-Cluster Communication Technique

For simulating proposed Intra-Cluster Communication Technique, we used parameters as like Table 4.2. We have compared proposed technique in term of packets sending per round and improvement using idle slots per round with traditional TMDA (LEACH-C) and A CSMA Based Intra Cluster Communication Technique (CICT).

Fig 4.3 shows packet sending vs Rounds. Almost 70 percent of total energy is consumed for communication purpose. So, it is very much essential to design a proper communication system. Proper Intra-Cluster Communication technique can save much energy. We have found that CICT is better than traditional TDMA (LEACH-C) method from packet sending and proposed protocol is much better than that of traditional TDMA and CICT system. Proposed protocol sends number of frame always more than that of two systems. From Fig. 4.3, we found that LEACH-C sends packets to CH and BS on an average of 4279, where CICT sends 4535 and our proposed technique sends 4721 packets per rounds. From this analysis, we found our technique improves throughput 10% and 4% than that of LEACH-C and CICT respectively.

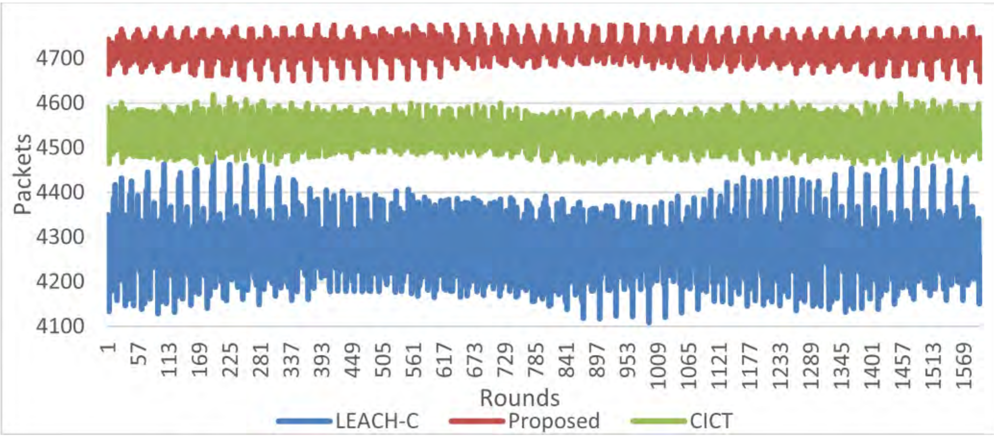


Figure 4.3: Packets per round

We have analyzed idle slots remaining of each round in Fig. 4.4. By analyzing this figure, we found that our technique remains vacant for 5.00% of available slots, where LEACH-C and CICT remain vacant for 13.90% and 8.75% respectively. So, our technique uses 8.9% and 3.75% more idle slots compare to LEACH-C and CICT respectively.

We have also analyzed proposed protocol by increasing number of nodes. We have found that total number of data is increased than that of Traditional TDMA and CICT respectively. Respective analysis is illustrated in Fig. 4.5.

At last we have analyzed proposed protocol by increasing number of rounds with a fixed number of nodes. We have used 100 nodes and increased number of rounds as

200, 400, 600,800 and 1000. From this analysis, it is seen that we are able to re-use

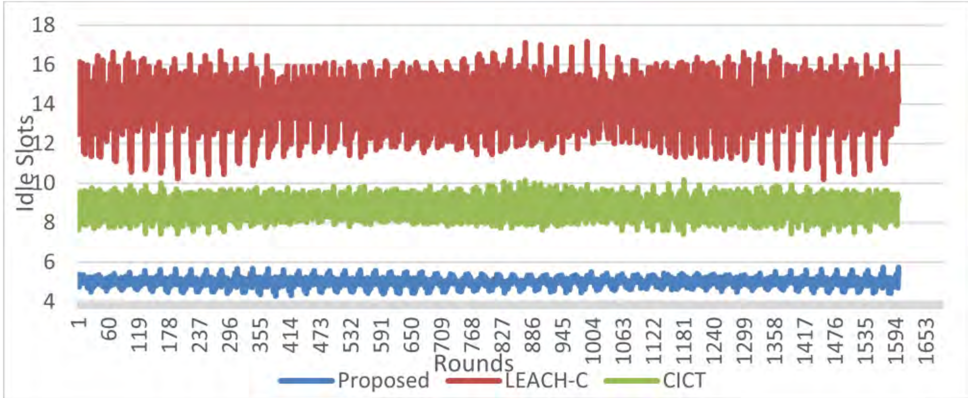


Figure 4.4: Idle slots per round

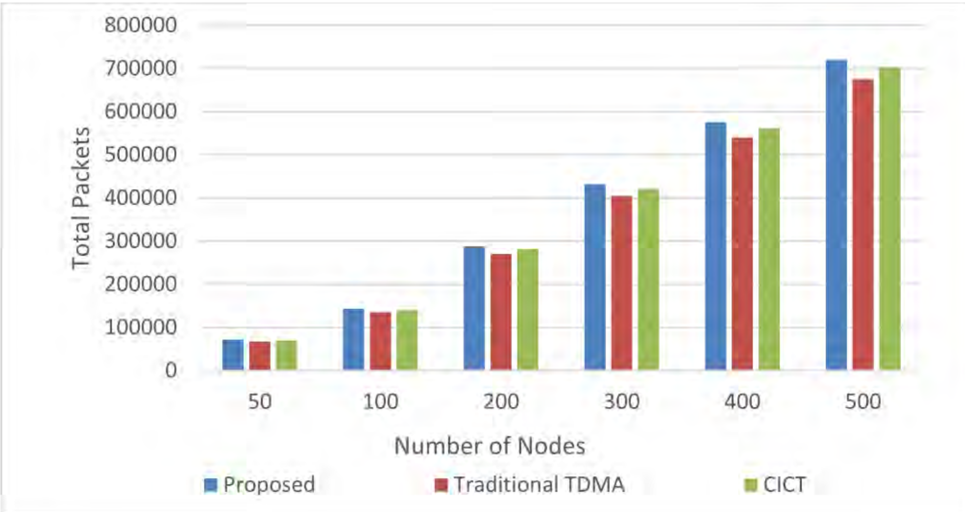


Figure 4.5: Nodes vs Total Packets

67% more idle slots than traditional TDMA and 24% than CICT shown in fig 4.6.

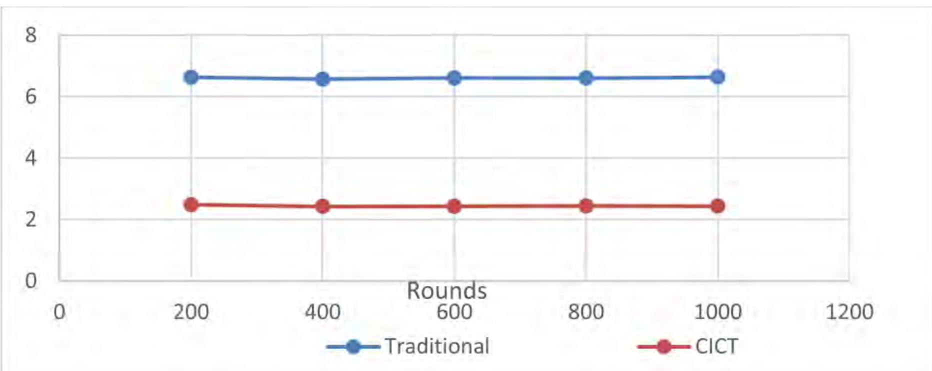


Figure 4.6: Rounds vs improvement reuse of idle slots

4.3.2 Combined Simulation

In this section, we have simulated the whole combination of proposed method, i.e., combination of CH selection, Cluster formation and Intra-cluster communication technique.

4.3.2.1 Selection of α Value

To select α, cumulative distribution function (CDF) has been used. To create CDF, we have considered α value from 0.1 to 1.0. We have run our simulation by considering α equal to 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0; and observed lifetime as well as throughput of the network. Then we have created probability mass function (PMF) and CDF. Value of α , lifetime, PMF and CDF are listed in Table 4.3. We have also generated consume energy vs. rounds charts. Then we have created PMF and CDF charts from Table 4.3. which is shown in Fig.1. From Fig-1 it is seen that, at alpha=0.4, lifetime is maximum (1700 rounds) where at alpha=0.1, lifetime is 1632 rounds. Comparing consume energy vs. rounds chart, we have seen that, if we select alpha=0.4, the network consumes huge energy compare to alpha=0.1. For all value of alpha rather than 0.1, the network consumes more energy. It is also clear from CDF chart. From this CDF chart (Fig. 1), it is seen that for alpha=0.4, increasing step is better than other alpha values. From consume energy vs. rounds charts, it is seen that if we select alpha=0.1, then the network consumed less energy compare to other alpha values. So it seems that alpha=0.1 is a suitable value to increase lifetime of the network.

Table 4.3: Various factor to select α

α	**Lifetime (Rounds)**	**PMF**	**CDF**
0.1	1632	0.11124	0.11124
0.2	1504	0.102515	0.213755
0.3	1407	0.095903	0.309659
0.4	1700	0.115875	0.425533
0.5	1370	0.093382	0.518915
0.6	1424	0.097062	0.615977
0.7	1410	0.096108	0.712085
0.8	1428	0.097335	0.80942
0.9	1282	0.087383	0.896803
1	1514	0.103197	1

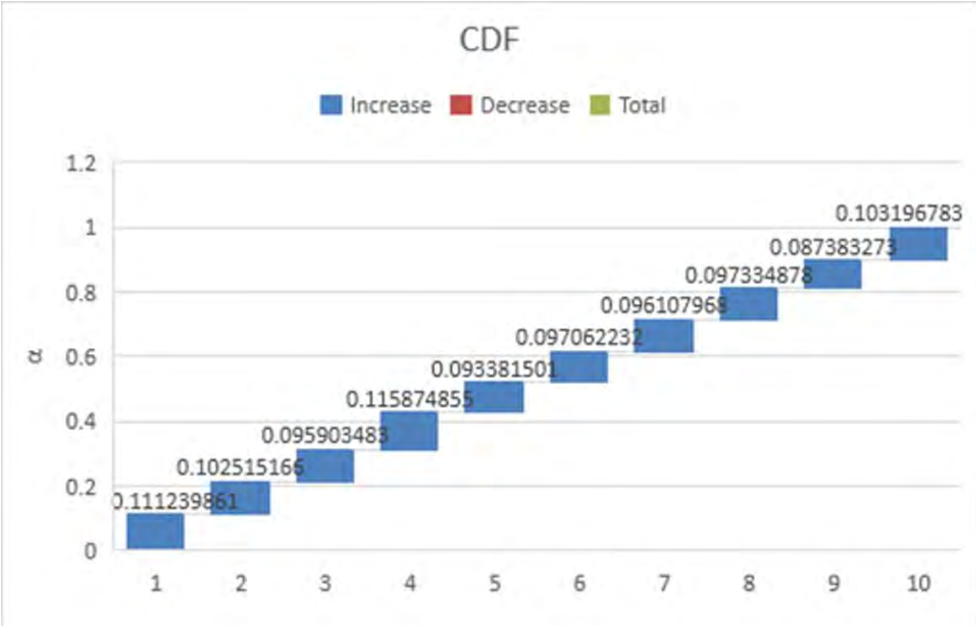


Figure 4.7: CDF chart with various α

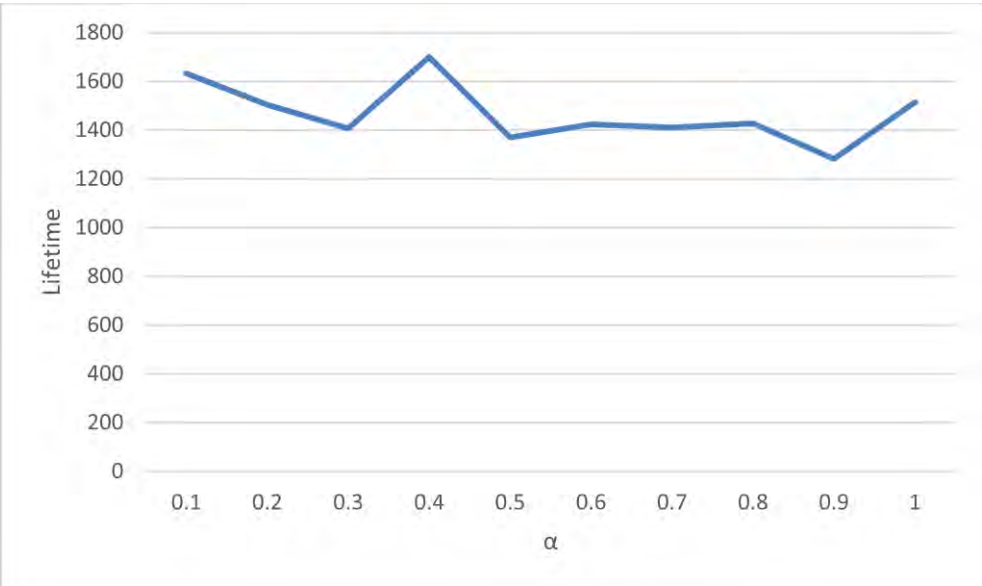
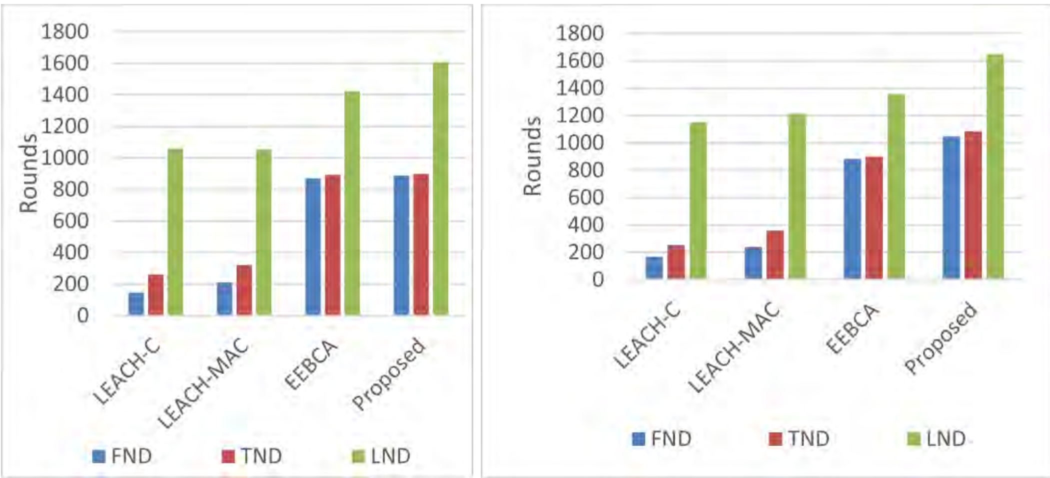


Figure 4.8: Lifetime vs. α

4.3.2.2 FND, TND and LND with Regard to Number of Round

FND, TND and LND terms were first introduced by Handy et al. [20]. Fig. 4.9 illustrated FND, TND and LND analysis and comparison with LEACH-C, LEACH-MAC and EECBA. From this analysis, we found that for 50 nodes FND, LND and TND required were 168, 254 and 1153 rounds for LEACH-C; 239, 360 and 1213 for LEACH-MAC; 883, 901 and 1356 for EECBA; 1048, 1084 and 1650 for proposed method. From this analysis, we have seen that our proposed method required more rounds in terms of FND, TND and LND compared to those methods, i.e., lifetime increased much more than that of the compared methods.

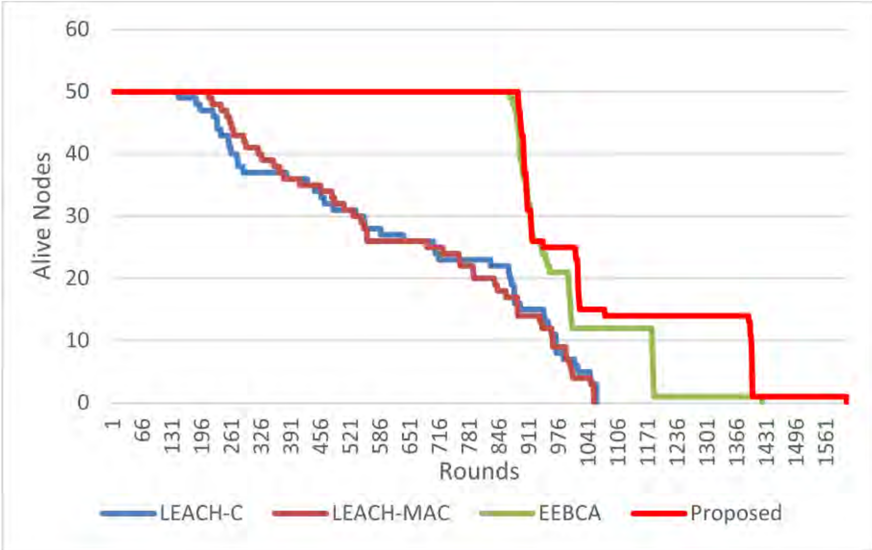


(a) (b)

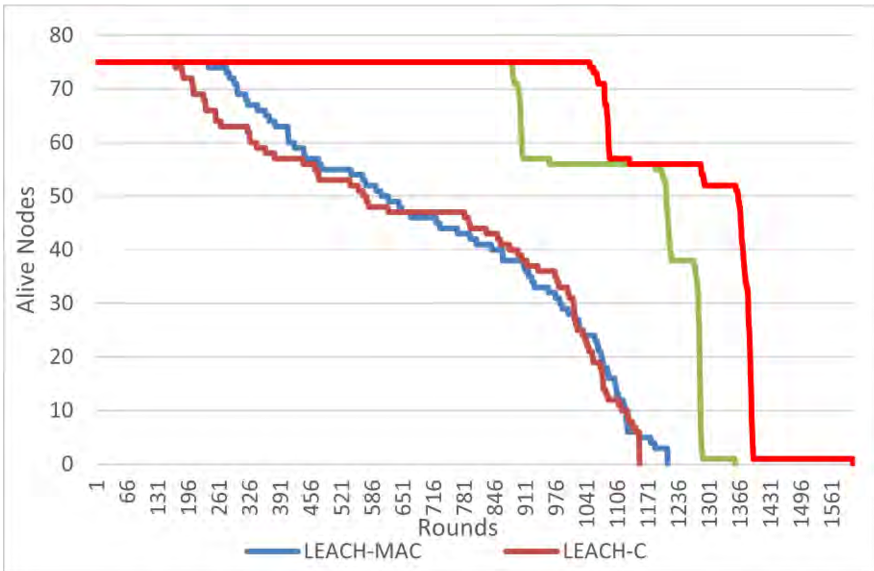
Figure 4.9: FND, TND and LND (a)50 and (b)75 nodes

4.3.2.3 Number of Alive Nodes with Respect to Rounds

Fig. 4.10 shows Rounds vs Alive nodes analysis. From this analysis, it is seen network with 75 [Fig. 4.10(b)] nodes is more stable than that of 50[fig-4.10(a)] nodes. It is also seen that every round has more nodes alive compared to LEACH-MAC, LEACH-C and EECBA. So, network remains much stable than those protocols and enhanced lifetime of the network.



(a)

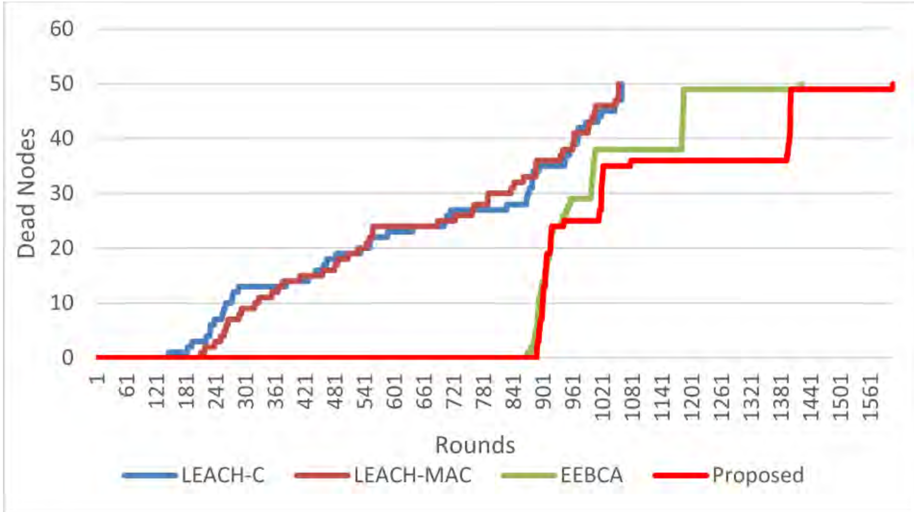


(b)

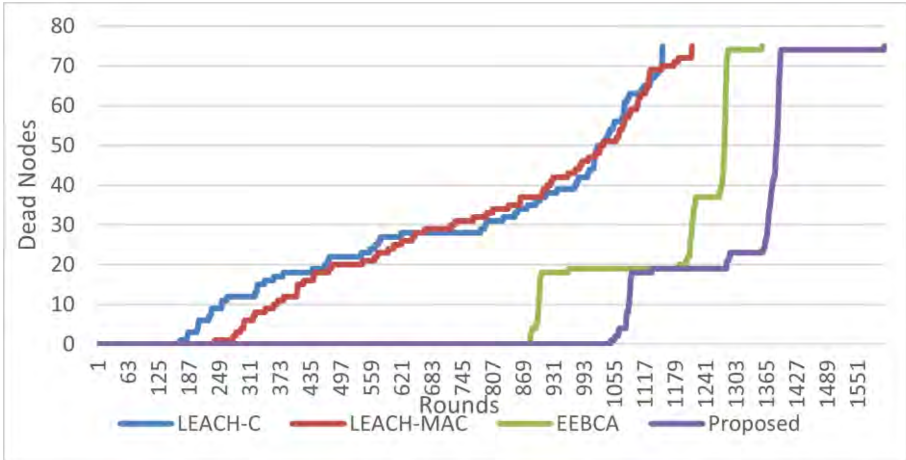
Figure 4.10: Number of alive nodes in the network with respect to round (a) for 50 nodes (b) for 75 nodes

4.3.2.4 Number of Dead Nodes with Respect to Rounds

Fig. 4.11 illustrate Rounds vs Dead Nodes analysis. For network stability, dead nodes should be as minimum as possible of each round. From this analysis, we found that our proposed protocol remains more stable than LEACH-C, LEACH-MAC and EECBA as number of dead nodes are less than those protocol. In 50 (Fig. 4.11(a)) and 75 (fig-4.11(b)) nodes network, nodes death started at round no 891 and 1084 respectively in proposed system, where nodes death started at round 34, 54; 36, 51; and 6, 19 in LEACH-C, LEACH-MAC and EECBA respectively. So, network remained more stable and network lifetime increased more than those protocols.



(a)

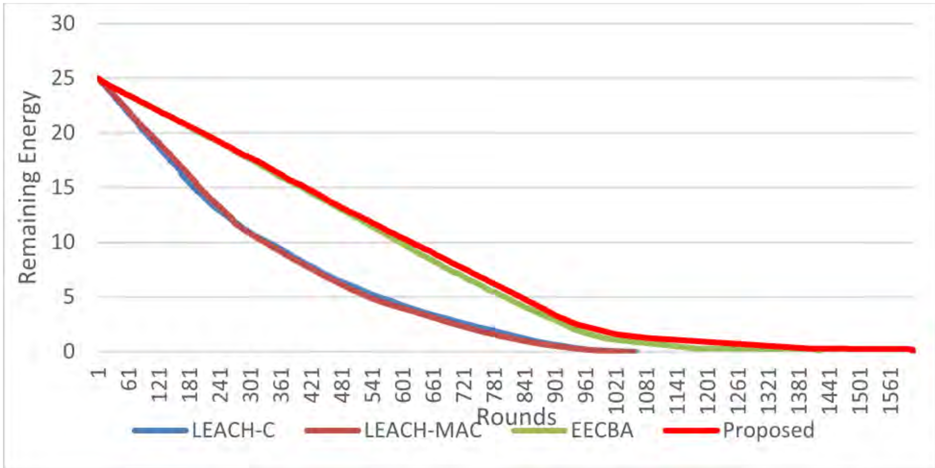


(b)

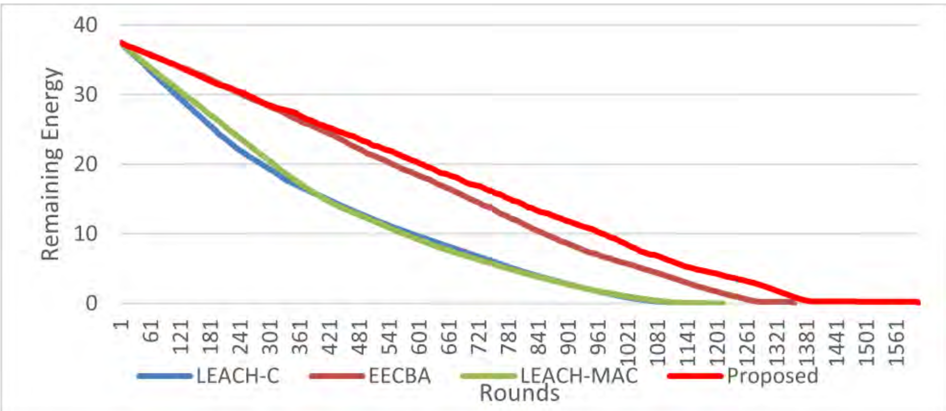
Figure 4.11: Number of dead nodes in the network with respect to round (a) for 50 nodes (b) for 75 nodes

4.3.2.5 Remaining Energy of the Network Per Rounds

Fig. 4.12 demonstrates the performance comparison in terms of overall remaining energy of the network with respect to rounds. From this Fig. 4.12, we found that more energy remains available in our scheme compare to LEACH-C, LEACH-MAC and EECBA. From Fig. 4.12 (b) it is seen in 75 nodes network, that 0.23001pf energy remains at round no 1606 in our proposed system, where 0.001881, 0.000029 and 0.230017 pf remains at round no 1151, 1213 and 1356 in LEACH-C, LEACH-MAC and EECBA respectively. From this analysis, we see that our protocol is more energy saving and increase lifetime compared to those protocols.



(a)

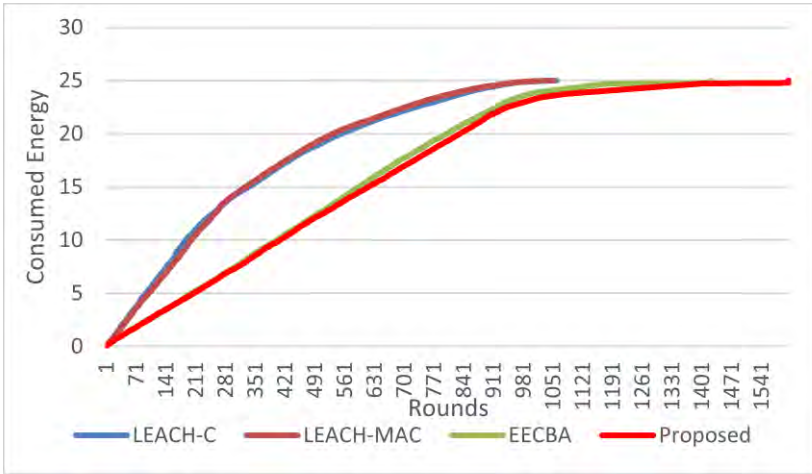


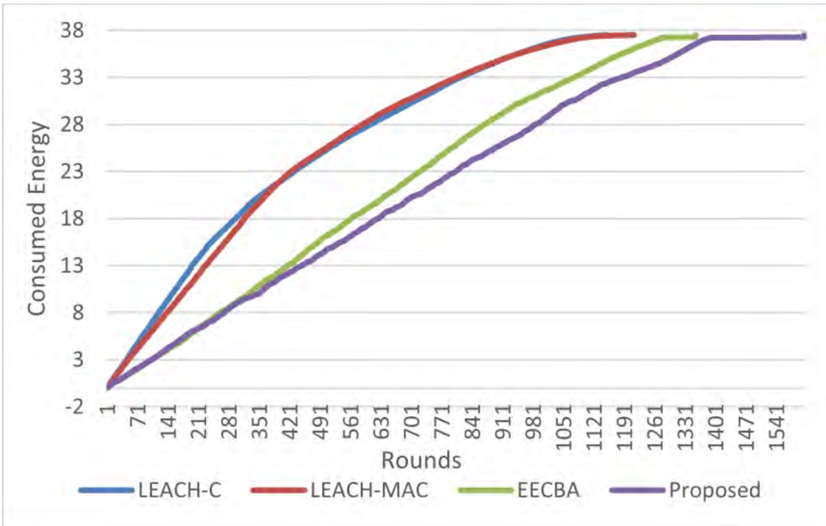
(b)

Figure 4.12: Remaining energy of the network with respect to round (a) for 50 nodes (b) for 75 nodes

4.3.2.6 Energy Consumption of the Network Per Round

Energy consumption of network per round is shown in Figure 4.13. It is observed that the proposed scheme consumes less energy than previous scheme as residual energy per round. So, it is clear that proposed technique consumed less energy than that of LEACH-C, LEACH-MAC and EECBA and increased lifetime of the network.



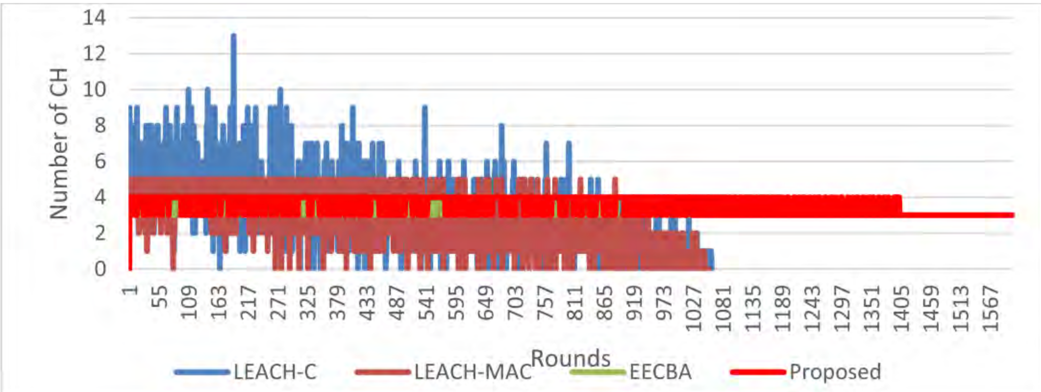


(b)

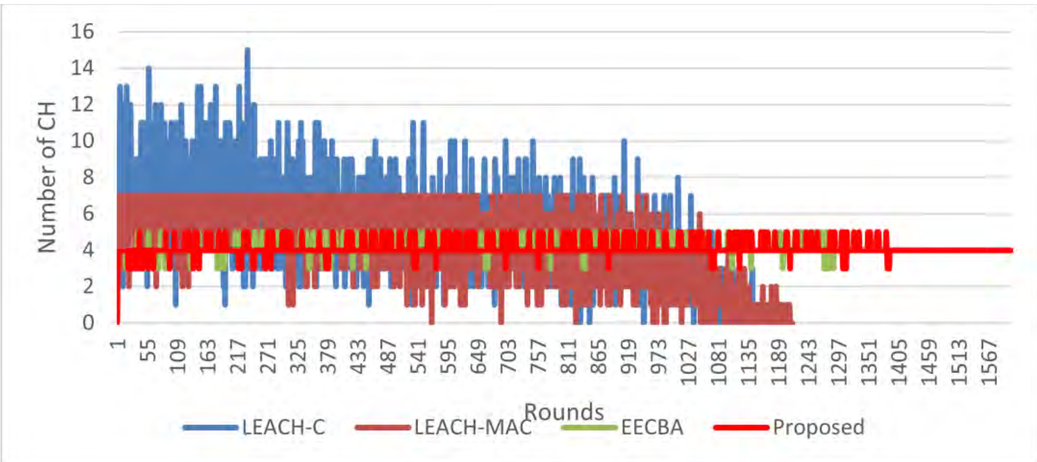
Figure 4.13: Energy consumption of the network with respect to round (a) for 50 nodes (b) for 75 nodes

4.3.2.7 Number of Selected CH Per Round

The number of CH selection should be in a balanced manner. Too many or too less number of cluster can degrade lifetime of the network. If number of CH become too high, then communication cost between CHs and BS will be high. So, more energy will be consumed. On the other hand, if number of CH become very low, then intra-cluster communication cost become very high. In both cases, huge energy will be consumed and lifetime of the network will be degraded. So, number of CH selection should be in a balanced manner. Fig. 4.14 illustrate number of CH comparison. Both Fig. 4.14 (a) and Fi. 4.14 (b) shows that, in the proposed protocol, the number of CH varies from 4 to 5. On the other hand, there is no restriction of CH selection for distributed approach. In LEACH-C, for the very first round, the number of CH is between 10 to 14 nodes that creates extra energy consumption.



(a)

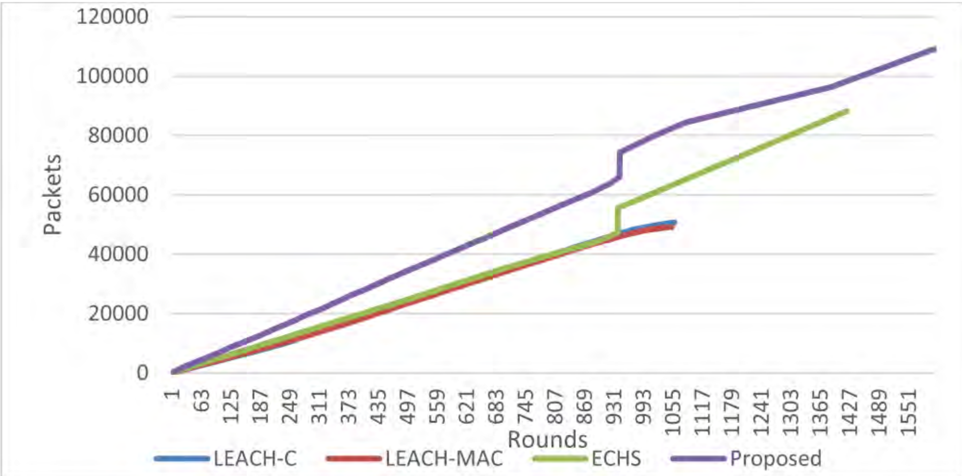


(b)

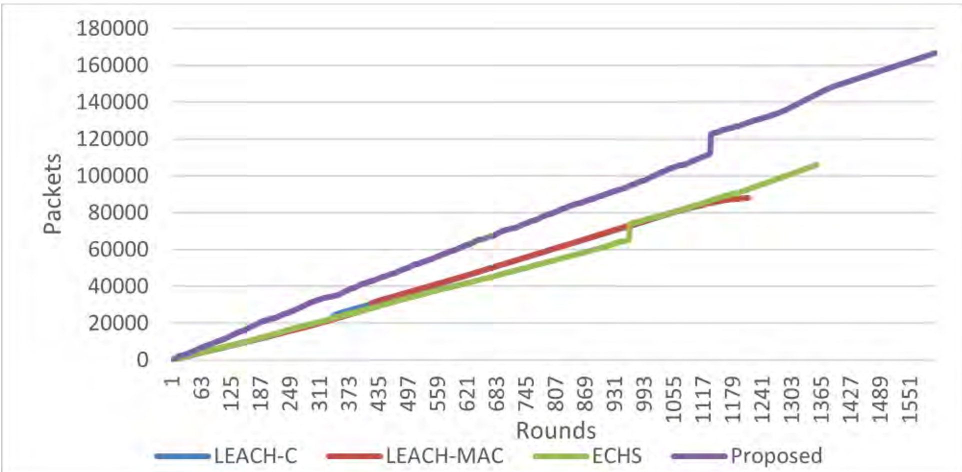
Figure 4.14: Number of selected CH with respect to round (a) for 50 nodes (b) for 75 nodes

4.3.2.8 Number of Packets Sent to Base Station

Fig. 4.15 shows the total number of packets sent to BS for both [50 nodes in Fig 4.15 (a) and 75 nodes in Fig. 4.15(b)] 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 4.4 and 4.5 shows the comparative performance analysis of the competitive protocols in terms of stability period, instability period, network lifetime and CH fluctuation.



(a)



(b)

Figure 4.15: Number of Transmitted packet to BS (a) for 50 nodes (b) for 75 nodes

Table 4.4 Performance comparison of competitive protocols for 50 nodes

Clustering Protocol	Stability Period	Instability Period	Network Lifetime Round	CH Fluctuation
LEACH-C	1-186	187-1060	1060	0-13
LEACH-MAC	1-220	221-1054	1054	0-5
EECBA	1-871	872-1422	1422	2-5
Proposed	1-885	886-1607	1607	3-4

Table 4.5 Performance comparison of competitive protocols for 75 nodes

Clustering Protocol	Stability Period	Instability Period	Network Lifetime Round	CH Fluctuation
LEACH-C	1-254	255-1153	1153	0-15
LEACH-MAC	1-318	319-1213	1213	0-7
EECBA	1-897	898-1356	1356	1-6
Proposed	1-1090	1091-1650	1650	3-5

4.3.2.9 Conclusion

In this chapter, we have simulated and compared our proposed protocol in terms of various matric with some existing protocols. We have analyzed and shown a great improvement of lifetime and throughput.

CHAPTER 5
Conclusion and Future Works

5.1 Conclusion

In this research work, we proposed a balanced clustering approach to enhance lifetime and throughput of WSN. We observed a lot of protocol for cluster head selection, cluster formation and intra-cluster communication technique in central and distributed approach. Usually, many researchers used residual energy, distance from nodes/ CHs to BS, positioning, etc., for cluster head selection. But, most of them cannot create a fully balanced cluster. In this research works we actually done three things: (a) CH selection using Residual Energy, Number of Neighbors, one hop neighbor information and distance from BS to CHs, (b) Cluster Formation using balancing nodes freely join to appropriate cluster, and (c) a new technique for Intra-Communication.

a) In this research work, we have used four parameters (Residual Energy, Number of Neighbors, one hop neighbor information and distance from BS to CHs) for CH selection which made our protocol more balanced and increased lifetime, and throughput.
b) We have used balancing nodes for energy balancing in cluster formation that enhanced balancing of the method.
c) In WSN, about most of its energy is spent for communication purpose. Intra-Cluster Communication Technique is one of them. Most of the energy is spent for this type of communication. We have offered a new Intra-Cluster Communication Technique. We used three steps in Intra-Cluster Communication Technique for Reducing energy consumption. At first, we used “Power Level Adjustment” for sending data to CH by exploiting capture effect. Secondly, we used “Time Slot Adjustment” by adjusting window size. Lastly, we used “Preamble based CSMA” for sensing channel. In this three steps, more packet sending is possible to CH and BS, hence, throughput is increased more and more compare to previous works.

We used OMNeT++ simulator for simulating proposed protocol and compared with LEACH-C, LEACH-MAC and another Balanced Clustering (EECBA) with various terms and shown a great improvement in terms of lifetime and throughput. 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, Cluster Formation and capture based intra-cluster communication technique using various adjustment.

5.2 Future Work

We have not considered mobile nodes in this protocol. Security aspect also have not been considered. So, a future extension of this protocol may be to address mobile environment and enhance security aspect. Moreover, a further research may be incorporated as collusion detection and prevention may be addressed.

CHAPTER 6

Bibliography

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

114 (2002).

[2] Dong M., Ota K., and Liu A., “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] Ben-Othman J. and Yahya B., “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] Wang J., Zhang Z., Xia F., Yuan W., and Lee S., “An energy efficient stable election-based routing algorithm for wireless sensor networks," Sensors, vol. 13, no. 11, pp. 14301-14320 (2013).

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

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

[7] Rout R. R. and Ghosh S. K., “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] Akkaya K. and Younis M., “A survey on routing protocols for wireless sensor networks," Ad hoc networks, vol. 3, no. 3, pp. 325-349 (2005).

[9] Zanjireh M. M. and Larijani H., “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] Zanjireh M. M., Shahrabi A., and Larijani H., “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] Soua R. and Minet P., “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] Pantazis N. A., Nikolidakis S. A., and Vergados D. D., “Energy-e_cient routing protocols in wireless sensor networks: A survey," IEEE Communications surveys & tutorials, vol. 15, no. 2, pp. 551-591 (2013).

[13] Patra A. and Chouhan S., “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] Arjunan S. and Pothula S., “A survey on unequal clustering protocols in wireless sensor networks," Journal of King Saud University-Computer and Information Sciences (2017).

[15] Aldeer M., Howard R., and Al-Hilli A., “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] Corn J. and Bruce J., “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] Afsar M. M. and Tayarani-N M.-H., “Clustering in sensor networks: A literature survey," Journal of Network and Computer Applications, vol. 46,

pp. 198-226 (2014).

[18] Heinzelman W. B., Chandrakasan R., A., and Balakrishnan H., “Energy-efficient communication protocol for wireless microsensor networks," in System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on, pp. 10-20 IEEE(2000).

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

[20] Handy M., Haase M., and Timmermann D., “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] Ali M. S., T. Dey, and Biswas R., “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] Thein M. C. M. and Thein T., “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] Beiranvand Z., Patooghy A., and Fazeli M., “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] Ye M., Li C., Chen G., and Wu J., “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] Nguyen T. G., So-In C., and Nguyen N. G., “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] Zhou W., “Energy efficient clustering algorithm based on neighbors for wireless sensor networks," Journal of Shanghai University (English Edition), vol. 15, pp. 150-153, (2011).

[27] Abdulsalam H. M. and Kamel L. K., “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] Batra P. K. and Kant K., “LEACH-MAC: a new cluster head selection algorithm for wireless sensor networks," in Wireless Networks, vol. 22, no. 1, pp. 49-60, Springer (2016).

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

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

[31] Wang J., Cao Y., Cao J., Ji H., and Yu X., “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] Essa A., A. Al-Dubai Romdhani Y., I., and Eshaftri M. A., “A New DynamicWeight-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] Sharma S. and Jena S. K., “Cluster based multipath routing protocol for wireless sensor networks," ACM SIGCOMM Computer Communication Review, vol. 45, no. 2, pp. 14-20 (2015).

[34] Rani S., Ahmed S. H., Talwar R., and Malhotra J., “Can Sensors Collect Big Data? An Energy Efficient Big Data Gathering Algorithm for WSN," IEEE Transactions on Industrial Informatics, vol. 13, issue 4, pp. 1961-1968 (2017).

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

[36] Wang N., Zhou Y., and Xiang W., “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] Karmaker A. and Hasan M. M., “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] Zeb A., Islam A. M., Zareei M., Al Mamoon I., Mansoor N., Baharun S., Katayama Y., and Komaki S., “Clustering Analysis in Wireless Sensor

Networks: The Ambit of Performance Metrics and Schemes Taxonomy," International Journal of Distributed Sensor Networks, vol. 12, no. 7 (2016).

[39] Srikanth B., Harish M., and Bhattacharjee R., "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] Hamza K. S. and Amir F., "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] Tripathi R. K., Base station positioning, nodes' localization and clustering algorithms for wireless sensor networks, PhD thesis, INDIAN INSTITUTE OF TECHNOLOGY KANPUR, 2012.

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

[43] Whitehouse K., Woo A., Jiang F., Polastre J., and Culler D., "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] Shimamoto S., Kudo H., and Onozato Y., "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] Murphy K. P., "Naive bayes classifiers," University of British Columbia, 2006.

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

[47] Varga A.," Modeling and tools for network simulation,” pp. 35-59, 2010.

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

[49] Abushiba W., Johnson P., Alharthi S. and Wright C., “An Energy Efficient and Adaptive Clustering for Wireless Sensor Network (CH-leach) using Leach Protocol,” in Computer Engineering, 13th International Conference (ICENCO), pp. 5-54, IEEE (2017).

[50] Karmaker A., Hasan M. M., Moni S. S., and M. Alam S., “An Efficient Cluster Head Selection Strategy for Provisioning Fairness in Wireless Sensor Networks,” in Electrical and Computer Engineering (WIECON-ECE), IEEE International WIE Conference on, pp. 217-220, (2016).

[51] Karmaker A., An Energy Efficient and Balanced Clustering Approach for Improving Throughput of Wireless Sensor Networks, M. Sc. Engg. Thesis, Institute of Information and Communication Technology (IICT), Bangladesh University of Engineering and Technology, 2017

Appendix A

Sample Simulation Code

Node.cc

```
#include <Node.h>
#include <Sink.h>
#include<RANDOM.H>
#include<custMsg_m.h>
#include<algorithm>
#include<string>
#include<stdlib.h>
#include<cUtility.h>
#include<cmath>

Define_Module(Node);
int setInf=0;

int numberOfCluster=0;
int clusterHeadListShortest[5000];
int l=4000;
int sortedNodesListFromBS[5000];
int spetialZoneNodes[1000];
int proCHIndex;
Node::Node() {
    // TODO Auto-generated constructor stub
    this->X = 0;
    this->Y = 0;
    batteryPower = 0.5;
    this->G = 0.0;
    this->type = 'N'; //Normal Node

    //Sleep time
    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;
    //efactor =  0.0000580;

    //dToBS = 109.06; //average distance to BS , calculated =
109.06//Average do = 87.70 // for 50 nodes
    dToBS = 109.06;
        //82.06; //average distance to BS , calculated = 109.06//Average do =
87.70 , 1370 80 = 1443

    thresholdEnergy = 0.23; //for 50 nodes
    //thresholdEnergy = 0.0; //for 100 nodes
        //0.0;

    noDataSentToCH = 0;
    noDataSentToSink = 0;
}


void Node::initialize() {

    //EV<<"PI: " <<PI<<endl;
    //endSimulation();

    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();
    //CalculateAvgDistanceToBS();
    //Caculate average distance to BS



    this->K = (sqrt(noOfWirelessNode) / sqrt(2 * PI)) * sqrt(Efs / Emp)
            * ((double) netSizeX / (dToBS * dToBS));

    //EV << "K: " << K << endl;
    //* sqrt((Efs * netSizeX) / Emp * dToBS * dToBS);
    this->R = sqrt((double) (netSizeX * netSizeX) / (PI * K)); //==55.3215
calculated
  }
ev<<"   K="<<this->K ;
    //Set Initial Cluster head and cluster
    if (simTime().dbl() == 0.0 && getIndex() == (noOfWirelessNode - 1)) {
        CalculateAvgDistanceToBS();

        //endSimulation();

        CalculateNeighborNode();
        SetEnergyMarker();
        CalculateWGTV();
        //endSimulation();
        ClusterHeadSelection(0);
        ClusterFormation(0);
    }


/*************************************************************
**********
     * Initial schedule for each node
     *
*************************************************************
********/
    scheduleAt(
            simTime().dbl()
                    + ((double) noOfWirelessNode - getIndex())
                            / (double) noOfWirelessNode, wakeup);
}
void Node::sendSelfData()
{
    int noOfNodeDied = getParentModule()->par("noOfNodeDied");
      int destNodeIndex = this->CHIndex;
      tempDestModule = getParentModule()->getSubmodule("node",
```

```
destNodeIndex);

    int queueLen = dataQueue->length();

    if (queueLen > 0) {
      //Added to calculate throughput
      noDataSentToCH = getParentModule()->par("noDataSentToCH");
      getParentModule()->par("noDataSentToCH") = noDataSentToCH
+ 1;

    }

}
//Calculate node distance from Sink

void Node::SendDataToCH() {

  int noOfNodeDied = getParentModule()->par("noOfNodeDied");
  int destNodeIndex = this->CHIndex;
  tempDestModule = getParentModule()->getSubmodule("node",
destNodeIndex);

  int queueLen = dataQueue->length();

  if (queueLen > 0) {
    cObject *cObj = dataQueue->pop();
    custMsg *qMsg = check_and_cast<custMsg *>(cObj);

    if (this->batteryPower > 0) {
      this->batteryPower = this->batteryPower - this->NETX;

      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;
        }
```

```
            deadTime = simTime().dbl();
        }
    }

    //int sendPacketToCH = getParentModule()->par("sendPacketToCH");
    //getParentModule()->par("sendPacketToCH") = sendPacketToCH++;

    //Added to calculate throughput
    noDataSentToCH = getParentModule()->par("noDataSentToCH");
    getParentModule()->par("noDataSentToCH") = noDataSentToCH +
1;

    sendDirect(qMsg, tempDestModule, "radioIn");
  }

}

void Node::SendDataToSink() {

  int noOfNodeDied = getParentModule()->par("noOfNodeDied");
  sinkModule = getParentModule()->getSubmodule("sink");

  int queueLen = chDataQueue->length();

  int noPacketSentToSink = getParentModule()-
>par("noPacketSentToSink");
  getParentModule()->par("noPacketSentToSink") = noPacketSentToSink
      + queueLen;

  while (queueLen > 0) {
    cObject *cObj = chDataQueue->pop();
    custMsg *qMsg = check_and_cast<custMsg *>(cObj);
    sendDirect(qMsg, sinkModule, "radioIn");
    queueLen--;
  }

  //Count number of data sent to sink
  //this->noDataSentToSink == this->noDataSentToCH;

  //Added to calculate throughput
  noDataSentToSink = getParentModule()->par("noDataSentToCH");
  getParentModule()->par("noDataSentToSink") = noDataSentToSink;


  if (this->batteryPower > 0) {
    this->batteryPower = this->batteryPower - this->CHETx;

    if (this->batteryPower <= 0) {
```

```
            noOfNodeDied++;
            getParentModule()->par("noOfNodeDied") = noOfNodeDied;
            deadTime = simTime().dbl();

            if (noOfNodeDied == 1) {
               getParentModule()->par("fstNodeDieRound") = roundNumber;
            }

            if (noOfNodeDied == 10) {
               getParentModule()->par("tenNodeDieRound") = roundNumber;
            }

            if (noOfNodeDied == noOfWirelessNode) {
               getParentModule()->par("allNodeDieRound") = roundNumber;
            }
         }
      }

}
```

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 cornerNodeEffect;

   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 PI; //const value 3.14
double K; //optimum cluster
double dToBS;
int energyMarker;
double alpha;
double efactor;
int spType;

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 ClusterHeadSelectionOld(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();

    void WriteDeadNodeHistory(int noOfDeadNode, int roundNumber);
    void WriteOneTenAllNodeDeadHistory();
    void WriteNetworkEnergyHistory(int roundNumber);
    void CountSinkPacket(int roundNumber);
    void CountThroughput(int roundNumber);


    void TempDataSendToCH();
    void TempDataSendToSink();
    void OptimalClusterFormation();
    void ThresholdCheck();
    void CountCH(int roundNumber, int noOfCH); //Cound CH vs Round

    void ResetParam();
```

```
int getMinDitanceClusterId(int nodeID);
double getMinDitanceOfaNode(int nodeID);

void centralCornerNodesTunningNew();

   void clusterEngeryCalculation() ;
   void clusterHeadListOfCurrentRound();
   void showAllClusterInfo();
   void calculateAllClusterEnergy();
void sortClusterListAccordingToEnergyCurrentRound();
void getIndividualClusterInfo(int clusterNo) ;
void generateAllNodesDistanceTable();
void clusterBalancing();
void centralBorderTunning();
void shortCHFromZeroZero();
void showNecessaryInfo();
void clusterBalance();


void centralBorderNodesTunningNew1();
void centralBorderNodesTunningNew();
void balanceCentralBorderNodesTunning();
void newProposedInfo();
int isABorderNode(int nodeID,int minDistClusterNo);
void otherNodesTunningNew();
void chkValue();
void engergyBalanceingNew();
int getNearestCH(int CH);
void sendEnergyToCluster(int FromclstrNo,int toClstrNo,double
avgEnergy);
int getSecondMinDitanceClusterId(int nodeID);
void centralBorderNodesTunning1();
void showSortingNodesFromBS();
void sortedNodeFromBS();
void sendSelfData();
//void clusterEngeryCalculation();
int getBeta();
protected:
   virtual void initialize();
   virtual void handleMessage(cMessage *msg);
   void finish();
   void adjustNext2Engr(int ClsNo,int Rno);
   void adjustNext1Engr(int ClsNo);
};

#endif /* NODE_H_ */
```

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);

      //Added for throughput
      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); //1=Setup, 2=Data send to CH,
3=Data Send to Sink
      int netSizeX = default(200);
      int netSizeY = default(200);

      int noPacketSentToSink = default(0);
      int sendPacketToCH = default(0);
      double dToBS = default(0);


      int noOfWirelessNode=50;// @prompt("Number of Nodes") =
default(2);
      //@display("bgb=netSizeX,netSizeY");
      //@display("bgb=800,600");
      @display("bgb=200,200");

   submodules:
      sink: Sink;
      node[noOfWirelessNode]: Node {
         @display("i=misc/node_vs,gold;p=230,140");
      }
```

```
}
```

Omnetpp.int

```
[General]
network = PLEACHPckg.PLEACH
#record-eventlog = true        #For generating event log
#sim-time-limit = 150s


[Config Nodes-50]
description = "50 Wireless nodes"
#CoAsymMac.noOfWirelessNode = 27 #Previous node = 100
#CoAsymMac.nodePerLayer = 10
#CoAsymMac.sinkX = 430
#CoAsymMac.sinkY = 79
#CoAsymMac.nodeRange = 70
#asymPercentage should be 5, 10, 15, 20, 25, 30, 35, 40
PLEACH.noOfWirelessNode = 50
PLEACH.clusterHeadPercentage = 0.1
```