Efficient Load Balancing and Data Aggregation Multipath Routing in Wireless Sensor Networks

A thesis submitted

in partial fulfillment of the requirements for the award of degree of

Doctor of Philosophy

by

Sukhchandan
(951303002)

under the guidance of

Dr. Sushma Jain
Assistant Professor

Computer Science and Engineering Department
Thapar Institute of Engineering & Technology,
Patiala – 147004

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY, PATIALA – 147004
November 2017

Certificate

I hereby certify that the work which is being presented in the thesis titled, "***Efficient Load Balancing and Data Aggregation Multipath Routing in Wireless Sensor Networks***", in partial fulfilment of the requirements for the award of degree of ***Doctor of Philosophy*** submitted in ***Computer Science and Engineering Department*** of Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Sushma Jain* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Sukhchandan

(Sukhchandan)

951303002

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(Dr. Sushma Jain)

Assistant Professor

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology, Patiala

Acknowledgements

First of all, I am highly indebted and thankful to Almighty for all his blessings and showing me the right path.

At the outset, I would like to express my appreciation to Dr. Sushma Jain for her advice during my doctorate's research endeavour for the past four years. As my supervisor, she has constantly persuaded me to remain focused on achieving my goal. Her observations and comments helped me to establish the overall direction of the research and to move forward with investigation in depth. Her constant pursuit for perfection, guidance and valuable suggestions has guided me at every step of my career. Particularly, I value her respect towards professionalism and the desire to excel higher and higher. Apart from research expertise, she has made me competent in organizing and managing research activities such as International conferences, workshops and software demonstrations.

I am highly thankful to Dr. Maninder Singh, Professor and Head, CSED for his motivation, kind help and unstinted support. I wish to express my gratitude to Dr. Rinkle Rani, Dr. Kulbir Singh and Dr. Ashima Singh for being my Doctoral committee members and advising me at the early stage of my research studies. I would also like to thank Dr. O. P. Pandey, Dean of Research and Sponsored Projects for academic support. My sincere thanks to Dr. Prakash Gopalan, Director, TIET for providing all the facilities, which have been instrumental in creation of a healthy research environment in Institute.

I would like to thank administrative and technical staff members of the Computer Science and Engineering Department who have been kind enough to advise and help in their respective roles. I am grateful to numerous local and global "peers" who have contributed towards shaping this mammoth research work

Last but not the least, I am highly grateful to all my family members for their inspiration and ever encouraging moral support, which helped me to purse my studies.

Sukhchandan

Abstract

Wireless Sensor Networks (WSNs) comprise spatially distributed wireless sensors, small battery powered devices, which are deployed among geographical areas to collect the data continuously. The routing plays an important role in a network for deciding the path for transmitting the data between source and destination based on certain Quality of Service (QoS) parameters. In order to transmit the data in an energy efficient and reliable manner, multipath routing is the most viable routing mechanism. The energy of nodes near to sink drains quickly and thereby giving rise to network hole and hotspot problem and resulting into partitioning of network. If the traffic or load can be distributed over the multiple routes, then network hole problem can be avoided and energy dissipation of a network can be balanced. To ensure reliability and load balancing, the trade-off is needed between the level of redundancy and energy efficiency. As the energy dissipation for communication is significantly higher than the energy required for computation, the energy can be conserved by performing in-network data aggregation at intermediate nodes instead of sending the whole data to the Base Station individually. Therefore, to optimize the QoS parameters such as energy, delay and network lifetime etc., there is a need of multipath routing framework for energy efficient data aggregation and load balancing for WSNs.

Currently, most of the data aggregation techniques maximize the energy efficiency without paying considerable attention to the data accuracy and varying conditions of underlying network topology. The level of data aggregation and frequency of data reporting should be adaptive with the varying density of the network and traffic pattern. The energy level of a node also gets affected due to collisions happening during the transmissions. These collisions should be prevented by applying protocols that are free of contention. While splitting and sending the data over the multiple paths, reliability and security should be ensured. Whenever the data is lost in between the source and the destination due to dead nodes, it should be recovered at the destination and data if captured by the malicious nodes, only a part of it

should be revealed. Consequently, there is a need of multipath load balancing routing technique, which can assure the required level of Quality of Service (QoS) parameters and avoids hotspot problem.

This work is carried out to develop load balancing and data aggregation multipath techniques to efficiently communicate the data based on energy efficient usage of network resources. A comprehensive investigation has been conducted to study various existing multipath routing techniques in WSNs for data aggregation and load balancing techniques. The load balancing through bio-inspired, nature inspired and other optimization techniques have been explored for WSNs. The investigations have been extended for multi-objective optimization in WSNs.

A novel Dynamic Adaptive Hierarchical Data Aggregation (DAHDA) technique has been presented for uniform and non-uniform networks while maintaining the data accuracy. The aim of this proposed technique is to reduce the energy consumption of sensors and therefore, increase the network lifetime without critically affecting the data accuracy. In addition, the algorithm is able to handle sudden bursts in the underlying data by recording the data in the area of interest for the whole event duration. It introduces the concept of weighted sensors and density based clustering to decide nodes to be selected as CHs and nodes responsible for sending data at certain rounds. The assignment of weights to nodes is based on the residual energy and density. It is a newly developed measure to determine the percentage of its neighbor nodes to the total number of nodes. The algorithm includes the adaptivity feature, which can handle any sudden bursts in the underlying data values to continuously make sure that the data in the areas of interest is captured. Three variants namely *DAHDA, Extended DAHDA (EDAHDA)* and *Modified EDAHDA* are presented using the level of underlying functionalities. Proposed algorithms are compared with variants of LEACH.

A Cross-layer Energy-efficient Clustering (CEC) technique is proposed for efficient contention free data aggregation in heterogeneous networks. In this technique, clusters of sensor nodes are formed in hexagonal shape. The cluster head is opted from the members of cluster itself on the basis of the ideal cluster head distance and remaining energy of sensor nodes having value greater than threshold value. In order to make a balance between consumption of energy and the network traffic, the rotation of cluster heads is performed. The energy level of a node gets affected due to collisions happening during the transmissions, which can be prevented by applying protocols that are free of contention. Slots are allocated

by cluster head to all member nodes within a cluster on the basis of their remaining energy so that nodes can switch to sleep mode. In the proposed technique, cluster head selection probability changes dynamically and data aggregation is performed by Cluster Heads based on the cost metric i.e. residual energy of the cluster head. The performance of proposed CEC technique is evaluated and compared with SOEECP, LCM and EEPCA.

A Particle Swarm Optimization Energy Efficient Load Balancing (PSO-EELB) technique is proposed. Based on maximum residual energy, paths are selected and load balancing is performed by splitting and sending the data over the selected paths using network based coding. Deterministic PSO is utilized for faster convergence of the algorithm. The underlying network is based on the clustered architecture in which a Cluster Head is elected on the basis of fitness function within each cluster. The fitness function is defined in terms of distance and residual energy of a node. The proposed technique performs better than existing techniques in terms of various QoS parameters.

In case of WSNs, there are scenarios when conflicting objectives are to be dealt. Based on the type of the application, the network scenario and required input/output of the problem, the type of optimization problem changes. Multi-objective Load Balancing Clustering (MLBC) technique has been proposed. Multi-Objective Particle Swarm Optimization (MOPSO), an evolutionary optimization approach, is utilized for accounting multiple objectives at a time. Two objective functions namely Energy Efficiency and Reliability have been considered simultaneously. Energy efficiency is measured in terms of residual energy of cluster head and reliability is measured in terms of packet delivery ratio. Weight is assigned to each node on the basis of residual energy and distance. The node having highest weight is selected as CH. A healing function is utilized in order to avoid loops in the generated path. The objective functions are evaluated for each individual. After, the completion of the execution, a set of non-dominating solutions called Pareto set is obtained. In order to choose the best compromised solution, fuzzy based approach is applied. The performance of the proposed MLBC technique is compared with techniques namely JPSO, MOPSO-DE and IMOWCA in terms of residual energy, packet delivery ratio and number of active nodes.

Contents

List of Figures

List of Tables

List of Abbreviations

Notation	Description
ABC	Artificial Bee Colony
ACMRA	Ant Colony based Multi-path Routing Algorithm
ACO	Ant Colony Optimization
ACOLBR	ACO based Load Balancing Routing Algorithm
ADA	Adaptive Data Aggregation
AOMDV	Ad-hoc On-demand Multipath Distance Vector
BMR	Braided Multipath Routing
BS	Base Station
CBLB	Cluster Based Load Balancing
CBLBA	Clustering based Load Balancing Algorithm
CCE	Connected Coverage based Energy efficient
CEC	Cross-layer Energy based Clustering
CH	Cluster Head
CoV	Coefficient of Variation
CRS	Cauchy Reed-Solomon
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTP	Collective Tree Protocol
DA	Data Aggregation
DAHDA	Dynamic Adaptive Hierarchical Data Aggregation
DAS	Data Aggregation Scheduling
DASDR	Data Aggregation Supported by Dynamic Routing
DAT	Data Aggregation Technique
DC	Density based Clustering
DCHT	Delay-Constrained High-Throughput

Notation	Description
DD	Directed Diffusion
DESA	Differential Evolution and Simulated Annealing
DUCF	Distributed Unequal Clustering using Fuzzy logic
ECMP	Energy Constrained Multi-Path
EDAHDA	Extended Dynamic Adaptive Hierarchical Data Aggregation
EDR	Energy efficient and Load balanced Distributed Routing
EECA	Energy-Efficient and Collision-Aware
EECS	Energy Efficient Clustering Scheme
EEMR	Energy Efficient-adaptive Multipath Routing
EEPCA	Energy-Efficient Prediction Clustering Algorithm
EEUC	Energy Efficient Unequal Clustering
EGDAM	Evolutionary Game-Based Data Aggregation Model
EQSR	Energy-Efficient and QoS-based
EW	Energy Window
FEC	Forward Error Correction
FLB	Fuzzy based Load Balancing
FoS	Focus of Study
FNNLBO	Fuzzy Neural Network Based Load Balancing Optimization
GA	Genetic Algorithm
GACCTR	Genetic Algorithm inspired Congestion Control using Trust based Routing
GPS	Global Positioning System
GSTEB	General Self-Organized Tree-based Energy-Balance
HDAS	Hierarchical Data Aggregation based Secure
HDMRP	Heterogeneous Disjoint Multipath Routing Protocol
HEED	Hybrid Energy Efficient Distributed
HSC	High-Speed Coding
H-SPREAD	Hybrid Secure Protocol for REliable dAta Delivery
JPSO	Jumping Particle Swarm Optimization
LA	Learning Automata

Notation	Description
LBI	Load Balancing for Infrastructure
LBEA	Load Balanced Energy-Aware
LCA-DG	Load-balancing Clustering Algorithm for Data Gathering
LCM	Link-aware Clustering Mechanism
LEACH	Low Energy Adaptive Clustering Hierarchy
LIEMRO	Low-Interference Energy-efficient Multipath ROuting
LP/NLP	Linear/Nonlinear Programming
MAC	Media Access Control
MCCT	Maximum Connected load-balancing Cover Tree
MCMP	Multi-Constrained QoS Multi-Path
MHC	Meta-Heuristic Computational
MHC-DC	Meta-Heuristic Computational – Density based Clustering
MLBC	Multi-Objective Load Balancing Clustering
M-MPR	Meshed Multi-Path Routing
MMSPEED	Multipath Multi SPEED
MNs	Member Nodes
MOEA	Multi Objective Evolutionary Algorithm
MOEA/D	Multi-Objective Evolutionary Algorithm based on Decomposition
MOP	Multi-objective Optimization Problem
MOPSO	Multi-Objective Particle Swarm Optimization
MOPSO-DE	Multi-Objective Particle Swarm Optimization Differential Evolution
MR2	Maximally Radio-disjoint multipath Routing
MST	Minimum Spanning Tree
NLL	Network Lifetime based Load balancing
PDR	Packet Delivery Ratio
PLBT	Pruning based Load Balancing Technique
PSO	Particle Swarm Optimization
PSO-EELB	PSO based Energy Efficient Load Balancing
PTC	Predicted Transmission Count
QoS	Quality of Service

Notation	Description
REAR	Reliable Energy Aware Routing
RREQ	Route Request
SCT	Semantic Correlation Tree
SI	Swarm Intelligence
SOEECP	Stochastic and Optimized Energy Efficient Clustering Protocol
TDMA	Time Division Multiple Access
TFCA	Time Frame based Congestion Avoidance
TLB	Threshold based Load Balancing
TMAC	Timeout Media Access Control
TS	Tournament Selection
UMAC	Utilized- Media Access Control
VWCA	Variable Weight based Clustering Approach
WDA	Weight based Data Aggregation
WSNs	Wireless Sensor Networks

List of Symbols

Symbol	Meaning
l	Length of packet in bits
N_{PT}	Number of Packets
CoV	Coefficient of Variation
d	Distance between Source Node and Destination Node
E_{mp}	Multipath Energy Loss
E_{fs}	Free Space Energy Loss
E_{TX}	Energy Required to Transmit a Message
d_o	Crossover Distance
E_{RX}	Energy Required to Receive a Message
n	Total Number of Nodes
k	Required Number of CHs
r_c	Current Round
S	Set of Nodes which have not been Chosen as CHs
Q	Network Area
q_n	n^{th} Quadrant
n_q	Number of Quadrants
R_1	Radius of the Cluster
$MinNodes$	Minimum Number of Nodes or Data Points required inside Cluster
A	Set of Objects
$E_{initial}$	Initial Energy of a Node
C_{rate}	Crossover Rate
C	Cluster, a Non-empty Subset of A
$y_i(j)$	j^{th} Variable of the i^{th} Individual
y_j^l	Lower Limit of j^{th} Variable
y_j^u	Upper Limit of j^{th} Variable
j_{rand}	Randomly Selected Index
U_i	Train Vector used in Mutation Operation

$nCluster$	Number of Clusters
S_i	i^{th} Sensor Node
d_i	Density of i^{th} node
e_i	Residual Energy of i^{th} node
w_i	Weight Assigned to i^{th} node
c	Coverage Range of Sensor s_i
d_{thresh}	Density Threshold
c_{thresh}	Number of Sensors that defines the Low Density Areas
$V_i avg(t)$	Accumulated Average of Sensor Reading upto Round t
$V_i(t)$	Current Recorded Value
$\Delta V(t)$	Difference between Current Value and Accumulated Average Value
n	Number of Nodes in a Network
n_s	Number of Samples
ct	Number of Categories
$n_{s,q}^{j}$	Number of Samples in Cluster q that belongs to the Original Class j
$Best\_Sol$	Best Solution based on Fitness Value
Ŋ	Scaling Factor in Mutation Operation
$VarN$	Number of Decision Variables
p	probability of Transmitting the Data
N_p	Population Size
ps	Application Specific Probability
d_{CH}	Density of Cluster Head
E_{amp}	Transmit Amplifier Energy
E_{elec}	Receiver/ Transmitter Electronics Energy
nbr	Number of Nodes in a Locality of a Particular Node
n_o	Number of Objects
n_{var}	Number of Variables
x_{mat}	Data Matrix of n_{var} Variables and n_m Objects
Pop	Population
$k_{dynamic}$	Percentage of Nodes used to Transmit Data in DAHDA
$Detection\_Ratio$	Quality metric to define the quality of detecting an event
$MaxIter$	Maximum number of iterations
E_o	Energy of an Optimal Node
f_1	Percentage of Nodes (Distinguished and Divine nodes) having Energy more than Optimal Nodes

f_2	Distinguished Nodes Percentage
E_{total}	Total Initial Energy of Network
p_t	Power of the Transmitter
G_r	Gain of Receiver
G_t	Gain of Transmitter
ED_{Loss}	Energy Dissipation in the Transmission of the Packet
λ	Wavelength of Carrier
β	Propagation Loss Factor
p_{thresh}	Threshold Power for Data Reception
e_a	Per Bit Energy Utilized in the RF Amplifier of the Transmitter
P_length	Length of packet in bits
e_e	Per bit energy consumption of the transmitter
e_l	Energy spent per second in listening state
d_R	Sending/ Receiving Data Rate of a node
E_t	Energy Consumption in transmission state of node
E_l	Energy Consumption in idle state of node
E_r	Energy Consumption in Data Reception State of Node
N_r	Number of Bits Received
R	Transmission range of node
N_t	Number of Bits Transmitted
L	Distance between Sink and Source
D	Inter-Cluster heads Distance
EW	Energy Window
p_t	Power of Transmitter
p_r	Power of Receiver
r	Length of hexagon's side /optimum distance for cluster head
E_{cap}	Average Energy Holding Ability of Member Nodes
l_a	Count of Levels of Aggregation
T_t	Transmission Time of Traffic between Clusters
T_r	Reception Time of Traffic between Clusters
T_l	Time Required to Sense the Radio Channel
E_i	Total Energy Required for Network Operations for Traffic Burst
α	Weight Assigned to Parameters in p_{set}
E_r	Average Energy of Current Round
$T_{absolute}$	Threshold Value when Energy Level of Distinguished and Divine

	Nodes is Same
ts	Size of Time Slot
n_slot	Number of Slots
k_i	Probability of a Node *i* to become Cluster Head
d_{R_max}	Maximum Data Transmission Rate
p_{set}	Set of Parameters
$G1$	Group of Optimal Nodes
$G2$	Group of Divine Nodes
$G3$	Group of Distinguished Nodes
R_{total}	Total Number of Rounds
E_{avg}	Average Residual Energy of a Network
T	Time Period of a Frame in a TDMA Schedule
n_d	Number of Sub-packets / Number of Disjoint Paths
S_n	A Random Variable representing the Number of Successfully Data Delivering Paths
P_i	Probability of Delivering a Packet to Destination Node Successfully
$E(S_n)$	Expected Number of Successful Data Delivering Paths
β	Upper Bound for Required Probability of Successfully Reconstructing the Sent Message at the Destination
μ	Mean of Normal Distribution
σ	Standard deviation of Normal Distribution
S_n^*	Standard Normal Distribution Variable
L_j	Load at Particular Sensor *j*
L_i	Load for a Group
P_n	Total Number of Packets Transferred in One Time Slot
L_{total}	Total Amount of Load Transferred in Entire Network
rS_n	Packets which are Generated by the Sensors of the Group
G_i	Group of Sensors
L_{pc}	Number of Packets Received from the Child Nodes of the Group (G_i)
P_p	Particle Position
R_v	Random Velocity
P_v	Particle Velocity
$lbestpos$	Local Best Position
$gbestpos$	Global Best Position
$OldVel$	Old velocity of particle
$currbestpos$	Current Best Position of Particle

$OldPosition$	Old Position of Particle
$NewPosition$	New Position of Particle
NL_n^n	Network Lifetime
X	Number of Iterations
n_p	Number of Paths
s	Area of the Cluster
NL_v	Lifetime of Node v
V	Node Set Excluding the Base Station
n_k	Number of Redundant Words
n_m	Number of Data Words
w	Size of the Words
Y	$n_k \times n_m$ Distribution Matrix
B	Matrix of Binary Code Words
D	Original Data Words Matrix
Z	Matrix in which n_mData Words are stored in Upper Part and n_k Code Words are Stored in Lower Part
M	Matrix having $n_m \times n_m$ I Matrix in Upper Part and Y Matrix in Lower Part
M'	Matrix M Received at Destination having $n_m \times n_m$ Elements
Z'	Resulting Matrix Z received at Destination
U	Cauchy Matrix of $n_m \times n_k$ Elements
I	Identity Matrix
$Y(p)$	Binary Representation of Number p
χ	Constriction Factor
X_{gb}	Global Best Position Ever Found among All Particles
$X_{i,pb}$	Personal Best Position Ever Found by the i^{th} Particle
c_1	Social Learning Rate
c_2	Cognitive Learning Rate
r_1, r_2	Random Numbers in the Range [0, 1]
v_i^k	Velocity of the i^{th} Particle at the k^{th} Iteration
X_i^k	Position of the i^{th} Particle at the k^{th} Iteration
w	Inertia Weight
α_1, α_2	Weighting Parameters (normalized values)
n_c	Number of Members Covered within a Cluster
γ	Represents the Presence or Absence of Coverage by a Current Node

Fv	Fitness Value
$FitnessValue_{CH}$	Fitness Value for Cluster Head Selection
$FitnessValue_{DT}$	Fitness Value for Data Transmission
$FitnessValue_{CF}$	Fitness Value for Cluster Formation
ω	Randomized Tuning Parameter
S_N	Sink node or Base Station
n_f	Number of Employed Bees or Food Sources
F_1	Objective Function 1 in Terms of Energy Efficiency
F_2	Objective Function 2 in Terms of Reliability
d_v	Number of Decision Variables
M	Number of Objective Functions
NBR	Neighbour Set of a Node
Cov	Coverage
NS	Number of Solutions in the Non-Dominated Set

Chapter 1

Introduction

Wireless Sensor Networks (WSNs) comprise spatially distributed wireless sensors which are small battery powered devices deployed among geographical areas to collect the data continuously. This captured data is used to detect occurrences of certain events and to take the appropriate actions accordingly. The sensor nodes capture data and then communicate to central unit, known as a Base Station (BS) or Sink, for processing, analysis and decision making. These sensor nodes have limited capabilities, limited lifetime and short communication ranges. Their applications include home automation, environmental monitoring, flood detection and military applications, etc. Designing a reliable WSN requires efficient methods in routing, data aggregation and load balancing.

Routing plays a key role in a network for deciding the path for transmitting the data between source and destination on the basis of certain parameters. To transmit the data in an energy efficient and reliable manner, multipath routing is considered as the most viable routing mechanism. It distributes the traffic load over the multiple routes, avoiding the network hole problem and balancing the energy dissipation of a network. Energy can also be conserved via in-network aggregation of data. Therefore, to optimize the parameters such as energy, delay and network lifetime etc., there is a need of multipath routing technique for data aggregation and load balancing for WSNs.

This Chapter summarizes an overview of Wireless Sensor Networks, architecture and protocol stack of WSNs and applications of WSNs. The chapter summarizes an overview of Multipath routing protocols along with the data aggregation and load balancing techniques. It also enlists the research motivation, research objectives, primary contributions of the research work and the organization of this Thesis.

1.1 Wireless Sensor Networks: An Overview

A Wireless Sensor Network (WSN) consists of a large number of sensor nodes deployed in region of interest for specific applications. These nodes are generally of small size and have computation, communication and sensing capabilities. Sensor nodes communicate via a short range radio signals and collaborate among themselves to accomplish the common tasks [1]. The main task of a sensor node is to sense the target phenomena and then report that data to the host controller or Sink in the form of query response as shown in Figure 1.1. However, the sensor nodes have limited bandwidth, power, memory, processing resources and lifetime [2], [3].

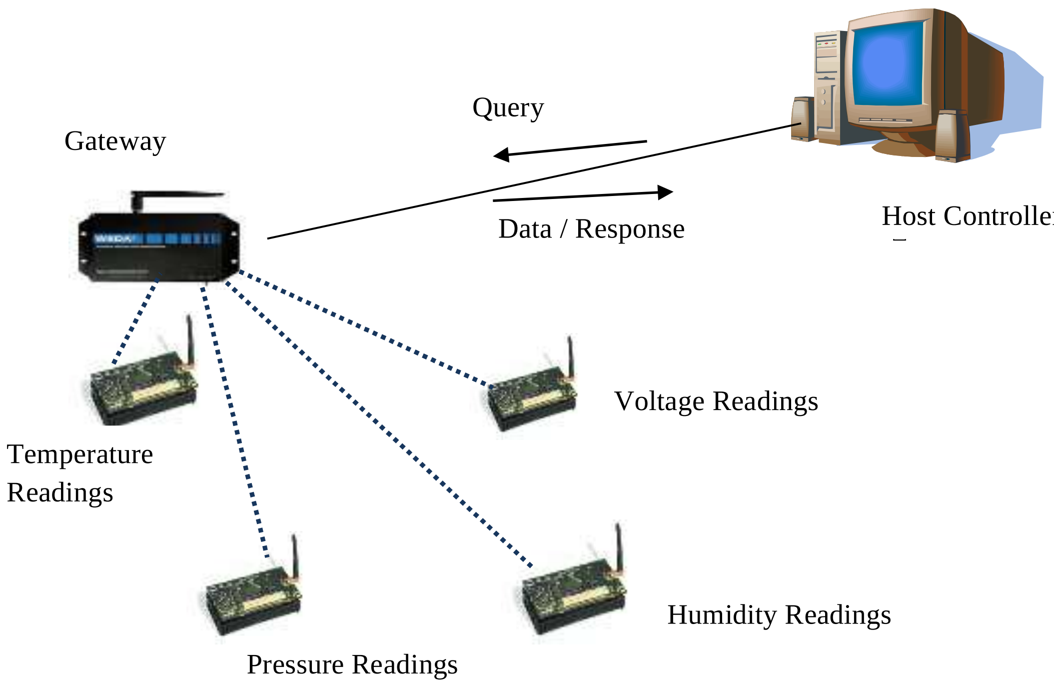


Figure 1.1 Wireless Sensor Network

Sensor networks may consist of different types of sensors such as seismic, magnetic, thermal, visual, infrared, acoustic and radar to list a few. These sensors are able to monitor a wide variety of ambient conditions such as temperature, humidity, vehicular movement, lightning condition, pressure, soil makeup, noise levels, the presence or absence of certain kinds of objects and the characteristics such as speed, direction and size of an object etc. [4]. The important parameters and objectives for an efficient WSN solution are: small size, power efficient, scalability, security, adaptability, self-configurability, reliability, fault-tolerance, bandwidth utilization, low cost and Quality of Service (QoS) support.

1.1.1 Architecture and Protocol Stack

The sensor node structure, as shown in Figure 1.2, usually consists of four modules *namely sensing module, processing module, communication module (transceiver)* and *power module* along with power generator. Depending upon specific application, it can also be equipped with other units. For example in case of location based routing, there can be a Global Positioning System (GPS) unit to provide location information for routing decisions. A sensor module senses the target and sends the sensed data to the processing module for computation. The task of a transceiver is to send and receive the data to/ from the external environment. An external power generator along with energy harvester system can be deployed to provide continuous power supply to the power unit depending upon the environment in which a WSN is deployed. A mobilizer module having an in-built motor may be needed to move sensor nodes in region of interest for some sensing tasks.

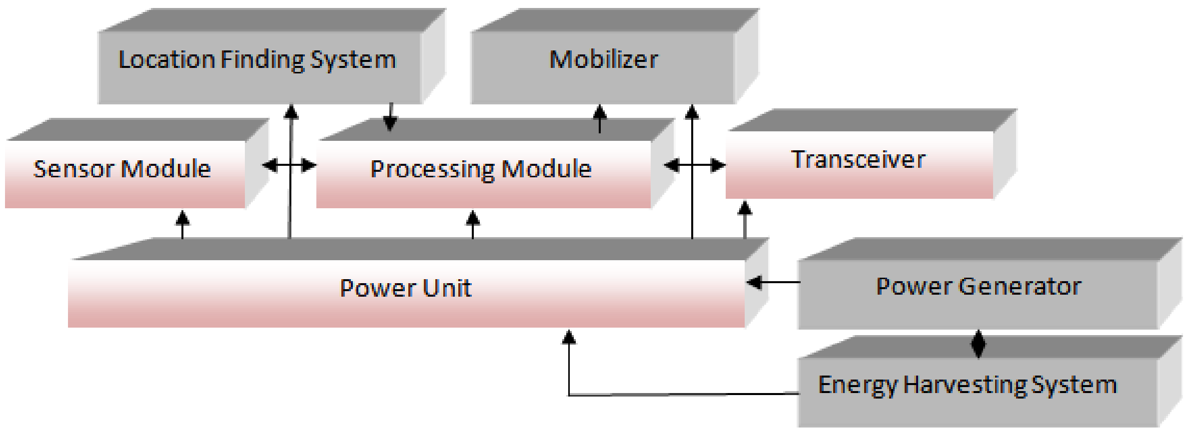


Figure 1.2 Sensor Node Structure

The protocol stack of WSN, as shown in the Figure 1.3, consists of five layers namely *physical layer* (media and interface dependent), *data link layer* (multiplexing, error control and flow control), *network layer* (routing), *transport layer* (upstream and downstream routing) and *application layer* (application oriented functions). The tasks and responsibilities of these layers are described as follows:

Physical Layer: It deals with the issues like conversion of bit stream into signals, frequency selection, data encryption, carrier frequency generation, signal modulation, design of underlying hardware and electrical/mechanical interfaces.

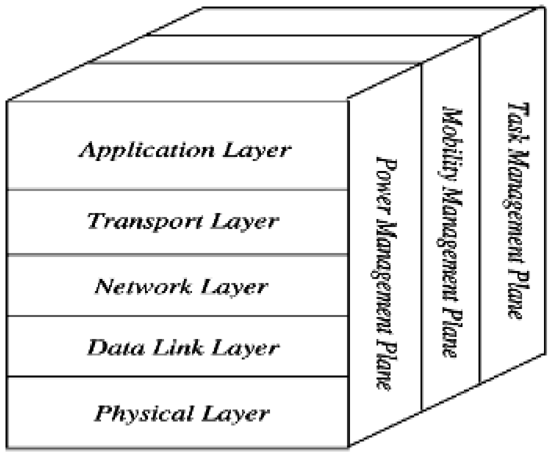


Figure 1.3 Sensor Network Protocol Stack

Data Link Layer: In order to provide reliable point-to-point and point-to-multipoint transmissions, this layer performs error control along with other tasks such as medium access control, data frame creation and data multiplexing. The primary objective of Media Access Control (MAC) is to share the resources among multiple sensor nodes in term of throughput, latency and energy consumption. In case of error control, the Automatic Repeat Request (ARQ) cannot be used. Despite being little bit complex, Forward Error Correction (FEC) is still used in WSNs.

Network Layer: The main task of this layer is to route the data (received from the upper transport layer) to the destination. It is important to take into account the energy constraint as well as unique traffic pattern in the design of network layer and routing protocols (for multi-hop, single-hop).

Transport Layer: It is responsible for reliable end–to-end delivery of data between the nodes and sink. There are two types of data delivery in WSN: *Upstream* and *Downstream*. In *Upstream* data delivery, sensor nodes send data to sink node while in *downstream* data delivery, data is originated at the sink node. The reliability of both upstream and downstream data delivery are different. The data delivery should be fault tolerant in upstream, but this is not the case in downstream data delivery.

Application Layer: It performs various application-oriented functions such as query dissemination, node localization and network security etc. For example: the tasks of Sensor Management Protocol (SMP), an application layer protocol, include moving sensor nodes to different locations, exchanging location related data and querying the sensor nodes for specific data etc.

The protocol stack is also divided into management planes including *power, connection* and *task management* planes. These planes are briefed as follows:

Power Management Plane: A sensor node performs various tasks such as sensing, communication and computation. This plane is responsible for managing the power level of various tasks. For example, sensor can be turned off when there is no data to transmit and receive.

Connection Management Plane**:** It is responsible for configuration or reconfiguration of sensor nodes in order to establish and maintain the connectivity of a network in case of node failure, node movement and deployment of additional nodes etc.

Task Management Plane: It is responsible for task distribution among various nodes to improve energy efficiency and prolong the network lifetime. It is possible in case of densely deployed WSN that many nodes may perform the similar/ redundant task that is of no use and leads to wastage of energy resources, so tasks should be properly assigned.

1.1.2 Applications Areas

WSNs have number of applications for event detection[5], continuous location sensing and local control of actuators. The applications can range from environment, military, health, home to other commercial areas etc. [6], [7]. This classification can be expanded with categories such as chemical processing, space exploration and disaster relief etc. as shown in Figure 1.4.

Military applications: WSNs can be used as an integral part of military intelligence, command, communications, control, computing, reconnaissance, surveillance and targeting systems [8]. The versatile characteristics of WSNs make these networks suitable for self-organization, fault tolerance, rapid deployment and a very promising technology for military services. Since WSNs are based on the dense deployment of disposable and low-cost sensor nodes, destruction of some nodes by hostile actions does not affect a military operation. Some of the military applications of WSNs are monitoring forces, ammunition and equipment; battlefield surveillance; reconnaissance of opposing forces and terrain; targeting; battle damage assessment and Nuclear, Biological and Chemical (NBC) attack detection and reconnaissance etc.

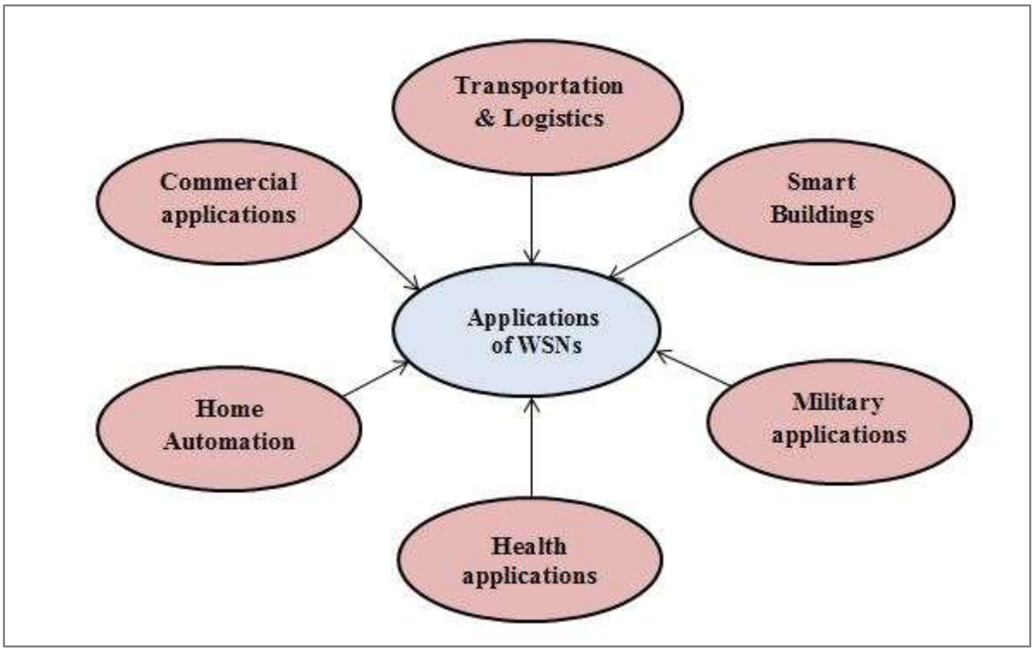


Figure 1.4 Applications of WSNs

***Environmental applications*:** There are several environmental applications of WSNs [9], [10], which include macro instruments for planetary exploration and large-scale earth monitoring; bio-complexity mapping of the environment; forest fire detection; tracking the movements of birds, small animals and insects; monitoring environmental conditions that affect livestock and crops; chemical/ biological detection; biological, earth and environmental monitoring of soil, marine and atmospheric contexts; flood detection; geophysical or meteorological research; irrigation and pollution study.

***Health applications*:** There are number of health applications of WSNs [11], [12] which provide interfaces for the disabled; drug administration in hospitals; integrated patient monitoring; diagnostics; monitoring the movements and internal processes of insects or other small animals; tele-monitoring of human physiological data; monitoring and tracking patients and doctors inside a hospital.

Home automation**:** With the advancement of technology, smart actuators and sensors are deployed in appliances, such as micro-wave ovens, vacuum cleaners, refrigerators and lighting system in Smart Home [13]. These sensor nodes can interact with each other and with the external network via Internet. These nodes allow end users to manage home devices more easily.

Smart environment**:** There are two design perspectives for smart environment, i.e., *technology-centered* and *human-centered* [13-15]. In case of human-centered, a smart environment can adapt itself according to the user's requirements in terms of input/output capabilities. In case of technology-centered, new networking solutions, middleware and hardware technologies have to be developed. Sensors are used to create such a smart environment.

Sensors are usually deployed in appliances where these nodes communicate among themselves and with room server. The room server communicates among themselves to know about the available services, e.g., printing, faxing and scanning. These sensor nodes and room can be deployed into embedded devices to make such environment adaptive, self-organizing, and self-regulating [16], [17]. The computing and sensing in this environment has to be reliable, persistent and transparent.

Other commercial applications **:** There are several other commercial applications of WSNs [18], which include factory process control and automation; managing inventory; monitoring product quality; material fatigue; building virtual keyboards; constructing smart office spaces; environmental control in office buildings; robot guidance and control in automating the environment for manufacturing; monitoring disaster area; transportation; vehicle tracking and detection; factory instrumentation; detecting and monitoring car thefts; local control of actuators and instrumentation of semiconductor processing chambers, rotating wind tunnels, machinery and anechoic chambers.

Apart from the above-discussed applications, the popularity of WSNs is growing day by day and WSNs are being used in number of real life scenarios.

1.2 Routing Protocols

The routing protocols are used to decide the best route to send data from the sensor node to host controller. The routing protocols, as illustrated in Figure 1.5, can be classified on the basis of *network structure* and *protocol operation* point of view. From network structure point of view, protocols can be classified into three categories namely *flat routing, location based routing* and *hierarchical routing*. In flat routing, a network usually consists of homogenous nodes that have same processing power, energy level and same data transmission capability. In hierarchical routing, a network is divided into number of clusters and within each cluster, a cluster head is elected to gather the data from other nodes in its cluster, aggregates the data and reports that data to the BS. In location based routing, the exact position of the nodes is gathered via GPS and then routing decision is made.

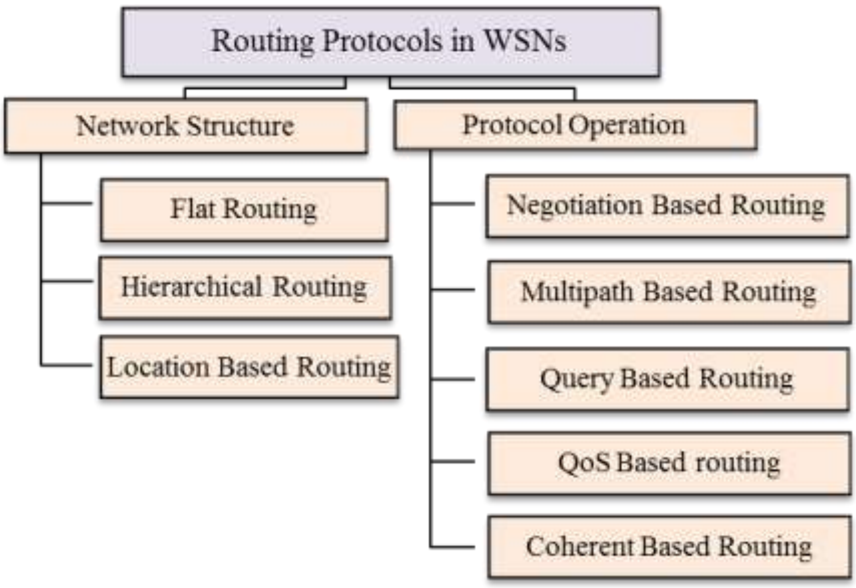


Figure 1.5 Classification of Routing Protocols

From protocol operation point of view, protocols are classified into five categories namely *negotiation based routing, multi-path based routing, query based routing, QoS based routing* and *coherent based routing* [19-21]. In negotiation based routing, data redundancy is reduced by providing higher-level description to the gathered data and by doing negotiations. In multipath routing, each node constructs multiple paths towards the destination, so that fault tolerance and reliability can be increased which is the major drawback of single path routing. In query based routing, the BS propagates the query for desired information. The node having the requested data sends back it to the BS. In QoS based routing, the desired QoS level such as throughput, delay and jitter is maintained depending upon the applications. In coherent based routing, data is aggregated first to reduce the data redundancy and enhance the energy efficiency. Otherwise, same event may be reported by multiple nodes and which will result into wastage of energy. The data can be sent via *single-hop* and *multi-hop* routing. In single-hop routing, there is direct data transmission between the source and destination. In multi-hop routing, there are number of intermediate nodes which forward the data between source and destination.

1.3 Multipath Routing

The multipath routing is carried out to provide high resilience and load balancing to the network. In single path routing, a shortest path between the source and destination is constructed based on metric such as number of hops, delay and distance etc. Then, the data is transferred through this single path which drains out its energy and results in the network partitioning [22]. The reliability is always an issue with single path routing because the data will not reach to the destination in case of any failure of path due to coverage problem or energy depletion. To cope up with such limitations of single-path routing, multipath routing has become a promising technique nowadays, which improves network lifetime, load balancing and data reliability. This technique identifies multiple routes to transfer the data from source to destination, which improves throughput and robustness of network transfer. The energy can also be utilized efficiently by balancing load across WSN, which results in improved network lifetime.

The multipath routing protocols are more secure, reliable and energy efficient. These protocols are classified into three categories as shown in Figure 1.6 based on the methodology and design for constructing paths and data transmission [23]. These categories are *infrastructure based protocols, non-infrastructure based protocols* and *coding based protocols*.

In infrastructure based routing, multiple paths are constructed between source-destination pair before the actual data transmission. In non-infrastructure based routing, there is no need to construct and maintain the multiple paths in the table; rather data is forwarded to the multiple next neighbours based on the local information. In coding based routing, data is divided into multiple fragments and then these fragments are transmitted individually on the multiple paths constructed.

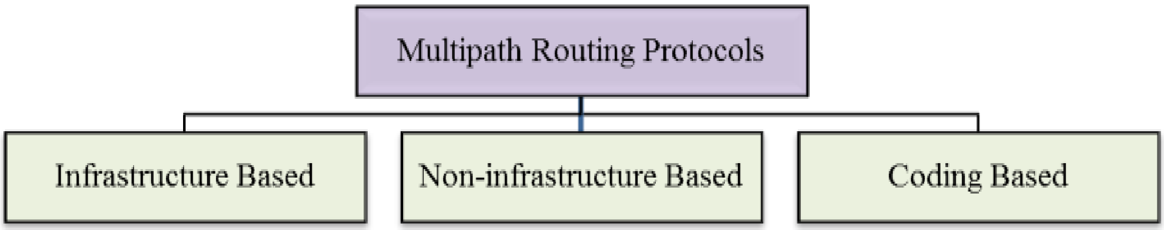


Figure 1.6 Classification of Multipath Routing Protocols

The discovered paths, as shown in Figure 1.7, can be further classified into three categories based on the amount of disjointedness as *link disjoint path*, *node disjoint paths* and *partially disjoint paths* [24][25]. In node-disjoint paths, the discovered paths do not have any node or path common among themselves. For instance in Figure 1.7(a), there are two paths from node A to F and no node is common in these two paths. Thus, the failure of one path will not affect the performance of the other paths. The link disjoint paths may consist of some common nodes thereby resulting into failure of multiple paths during the failure of any node as shown in Figure 1.7 (b). Partially disjoint paths may have multiple nodes and links common among the multiple paths discovered, but these paths are easy to discover as compared to other type of paths. In Figure 1.7 (c), there are multiple paths discovered between node A and node E having multiple nodes and links in common.

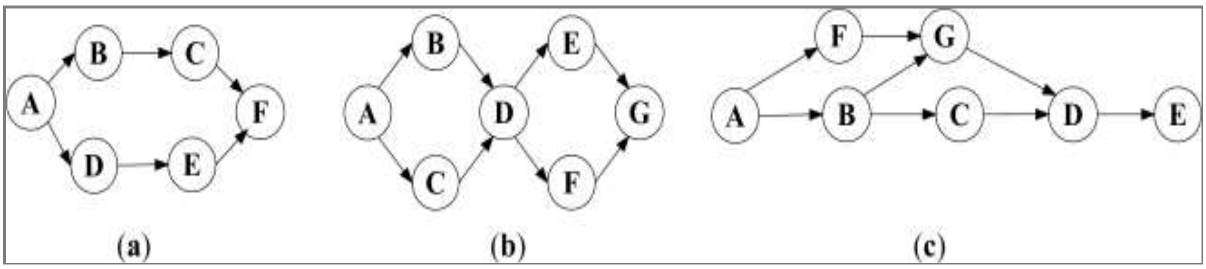


Figure 1.7 Types of Discovered Paths (a) Node Disjoint Path (b) Link Disjoint Path (c) Partial Disjoint path

The key benefits of multipath routing over single path routing are as follows:

- Multipath routing offers flexibility in case of node or link failure via two approaches: i) transmitting multiple copies of the similar data over the multiple routes to guarantee data recovery in case of route failure, and ii) By using network coding, in which data packets are divided into multiple sub-packets and some extra information is added. The sub-packets are transferred over different routes. At sink node, original packets are reconstructed based on extra information.

- Fault tolerance is an important aspect of routing protocol and this can be achieved through multipath routing by transferring redundant information from source to destination through alternative paths that leads to minimum disruption of communication.

- In high-data rate applications, network congestion occurs due to high traffic load which effects performance of WSNs. The network traffic is divided over multiple routes using multipath routing to minimize the chances of network congestion.

- Data is divided into different streams and is transferred to destination through multiple paths by aggregating bandwidth of the channels (multiplexing). This technique can be useful specially when there are multiple low bandwidth channels and the required bandwidth is more than individual bandwidth of a channel.

- The data transmitted over the network is not secure in case of single path routing as a node may get compromised due to malicious activity or cyber-attacks. Since, the whole traffic is transmitted over the single path, it can be easily altered or captured via malicious node. Another type of attack is selective forwarding, in which the packet that is supposed to be forwarded is dropped by malicious node. Multipath routing can minimize the effects of such type of attacks. In case of multipath routing, as the data is transmitted over the multiple paths, the robustness and confidentiality of data can be assured. In network coding based multipath technique, data is transmitted through encoding and decoding process to prevent eavesdropping of the sensed data during transmission. Therefore, this method is effective in resource-oriented environment where much less power is needed for encryption and decryption during communication.

- The usage of same path for data transmission in single path routing can cause energy depletion very quickly which leads to network partition. By using multipath routing, load can be distributed in a balanced way and network partitioning can be prevented. Rather than sending the multiple copies of the same data, it can be divided into sub-packets and transmitted over multiple paths along with minimal redundancy. In order to combine the data coming from multiple sources, data aggregation technique is required in multipath routing.

The design of multipath routing protocol comprises components to construct multiple routes between source and destination and to transfer data over identified routes. The important components of efficient multipath routing protocol [26] are:

- **Path Discovery:** Generally, data transmission is performed in WSNs using multi-hop data forwarding approaches, in which a set of intermediate nodes should be identified to make several routes from source to destination. Different routes are identified from source to destination using different parameters.

- **Path Selection and Traffic Distribution:** It is also important to select the paths from total discovered paths based on the performance demands of the intended application. Further, multipath routing protocol distributes traffic over selected paths using different traffic management methods.

- **Path Maintenance:** To improve network performance and its reliability, there is need to maintain the multiple paths from source to destination periodically. Discovery of path after the failure of active node affects network performance and increases overhead. Multipath routing protocol selects different optimal path in case of path failure. The discovery of path is initiated in three scenarios: i) if all the active paths have failed, ii) if certain number of active paths have failed and iii) if a single active path has failed, then an alternate discovered path should be utilized.

1.4 Data Aggregation in Wireless Sensor Networks

Data aggregation is defined as the process of aggregating the data from multiple sensors to eliminate redundant transmission and provide consolidated information to the BS. The energy consumption is less for computation as compared to data transmission in WSNs [27]. Rather than sending the sensed data each time to the sink node individually, if the data is first collected and aggregated using aggregate functions such as sum(), avg() etc. and then forwarded to the sink, a lot of energy will be saved. The effectiveness of the communication among nodes depends on the data aggregation technique being used. Data aggregation can be considered as a fundamental processing procedure to reduce energy consumption and to save the limited resources. An effective data aggregation technique can enhance network lifetime. Data aggregation can also be useful when multiple nodes sense the same phenomena due to high node density (also known as *data overlapping problem*) [28]. In addition, the amount of data generated in large WSNs is usually enormous for the BS to process. Use of an effective data aggregation technique can help in processing speed. The working of data aggregation technique (DAT) in WSN is shown in the Figure 1.8. The DAT can be used to aggregate the sensor readings.

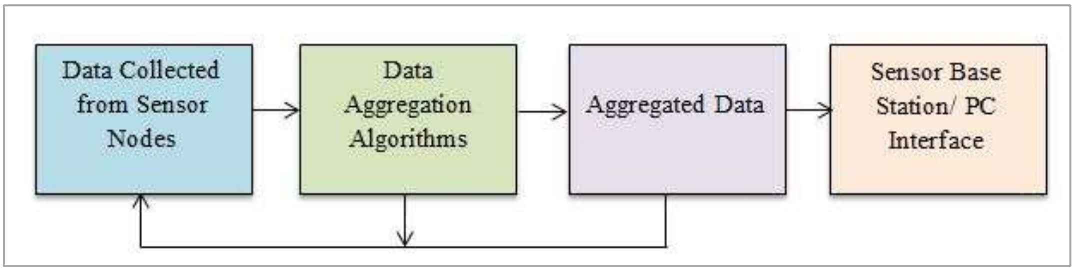


Figure 1.8 Data Aggregation Process

First of all, data is collected from various nodes. The various algorithms like Low Energy Adaptive Clustering Hierarchy (LEACH), Centralized approach, Tiny AGgregation (TAG) etc. are used to aggregate the sensor data coming from the sensor nodes. The input for an aggregation algorithm is the sensor readings taken from various nodes and output is the aggregated data. After aggregation, an efficient path is selected to transfer the data to the sink node. The idea of data aggregation is to combine the sensed data coming from multiple sensor nodes by applying various statistical or algebraic operations such as multiplication, addition, median, maximum, minimum and mean etc. The data aggregation architecture model of WSNs is illustrated in Figure 1.9.

In beginning, nodes are selected and distributed among different clusters. Parameters like energy consumption, memory and bandwidth etc. are considered to calculate the number of nodes that will participate in a cluster. After this, one Cluster Head (CH) is nominated from each cluster [29]. The responsibility of CH includes supervision of all nodes inside cluster and gathering the data from every node in a cluster and transfers this data to adjacent CH for update operation.

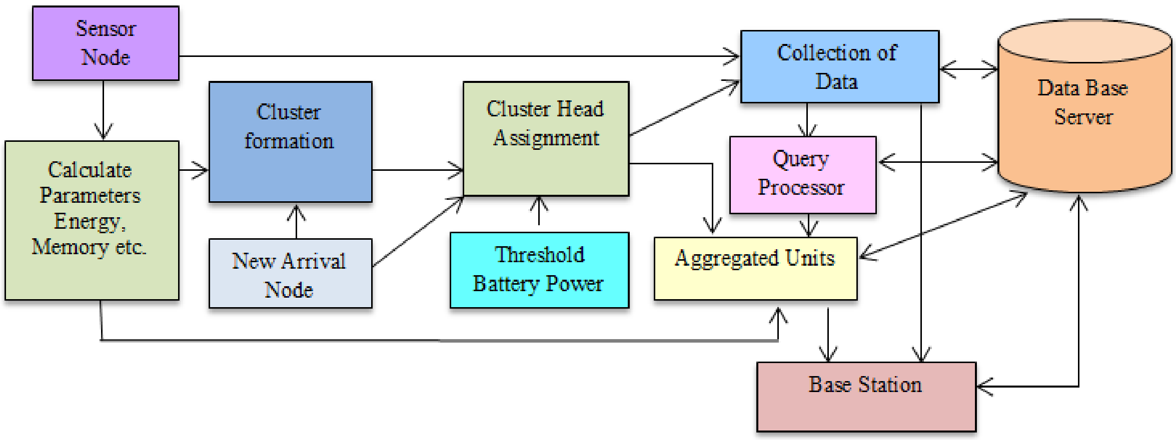


Figure 1.9 Data Gathering and Aggregation Architecture in WSN

Cluster will be allotted to newly arrived nodes. The collection of data and various user queries are processed by data aggregation technique and then query processor transforms these queries into low level format [30]. Database server is used to store the gathered and aggregated data.

1.4.1 Approaches of Data Aggregation

Data aggregation can be defined as a process of collecting and aggregating the data which is performed by a cluster head or master node so that network lifetime can be extended. The aggregated data is sent to the BS via underlying routing mechanism. There are various approaches to perform data aggregation in WSN [31], [32]. The classification of these DAT techniques is shown in Figure 1.10.

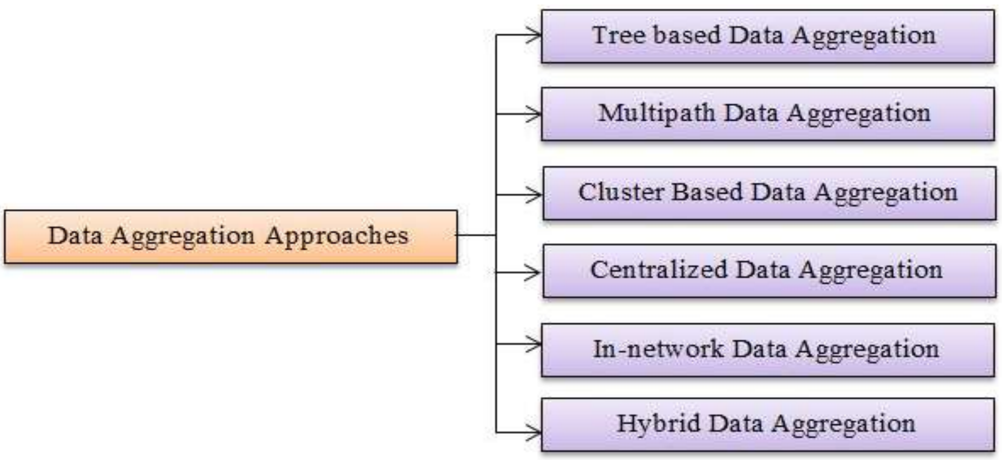


Figure 1.10 Data Aggregation Approaches in WSNs

Tree based Data Aggregation Approach: In Tree based approach; data aggregation is performed by constructing aggregation tree, which can be a Minimum Spanning Tree (MST). In this tree, root node acts as BS, leaf nodes act as source nodes and intermediate nodes act as parent nodes. Leaf node send its sensed data to its parent node in a path discovered between leaf node and BS as shown in Figure 1.11. The aggregation is performed at the parent node.

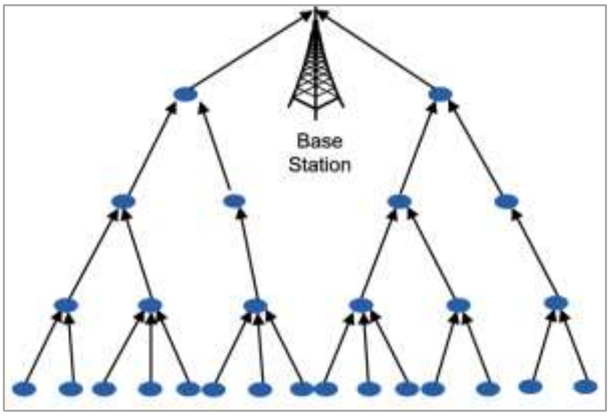


Figure 1.11 Tree based Data Aggregation Approach

Multipath Data Aggregation Approach: WSNs are prone to frequent route and node failures. In such scenarios, tree based approach is not suitable as the parent/leaf node or the path can be easily damaged or compromised due to external attacks or energy depletion. In case of multipath data aggregation approach, each node sends the data packets to its neighbouring nodes via multiple discovered paths. The task of data aggregation is performed at every intermediate node between sources and BS as shown in Figure 1.12.

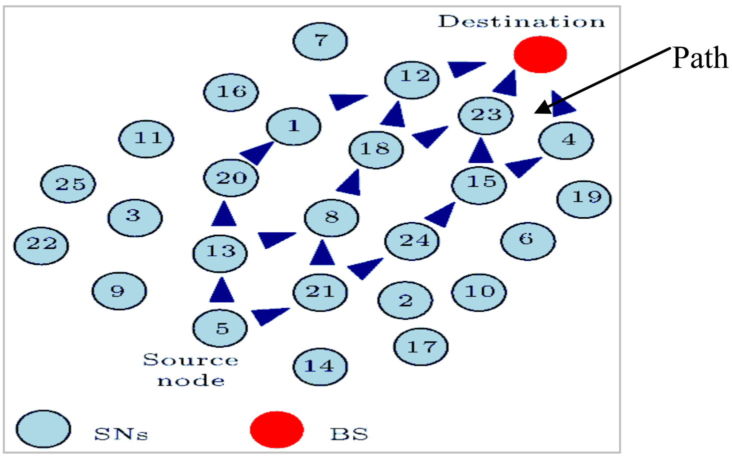


Figure 1.12 Multipath Data Aggregation Approach

This approach makes the network more robust however the communication overhead ratio will be higher than the tree based approach. In any event of link or node failures, this approach discovers alternative paths to ensure the delivery of as many packets as possible within the time constraints.

Cluster Based Data Aggregation Approach: In cluster based approach, the whole network is divided into several clusters. The sensor nodes themselves form a cluster and elect a node as cluster head as shown in Figure 1.13. The data sensed by the sensor nodes is passed to the cluster head. Cluster head performs data aggregation and forwards the data to the sink. Communication cost is reduced since only aggregated results reach the BS. Each node reports the data to its cluster head rather than sending the data directly to the BS. Hence, it saves a lot of energy in a network.

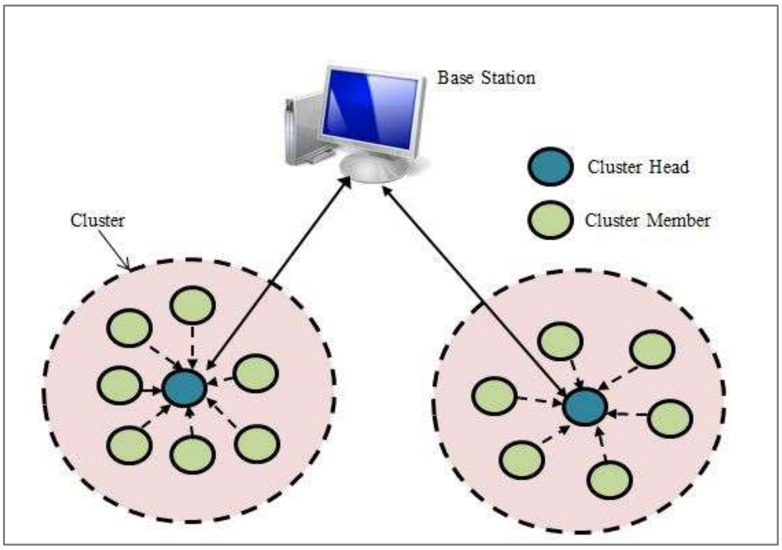


Figure 1.13 Cluster based Data Aggregation Approach

Centralized Data Aggregation Approach: In this approach, each and every node transmits its sensed data to the central node, which is usually the most powerful node (in terms of resources such as energy, lifetime and bandwidth etc.) using the shortest path as shown in Figure 1.14. Address centric routing is used along with multi hop algorithm by considering cost metric at every intermediate node. The central node's task is to aggregate the data received from other nodes and report the same data to the BS. This approach suffers from the problem of high traffic due to large number of messages being transmitted to the central node. The various techniques like vector quantization clustering are implemented for efficient centralized data aggregation.

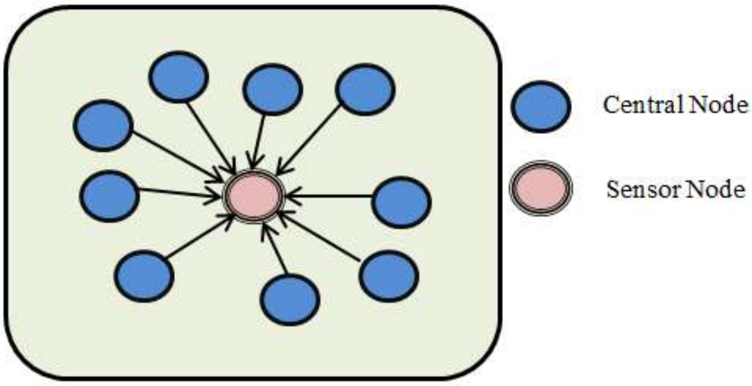


Figure 1.14 Centralized Data Aggregation Approach

In-Network Data Aggregation Approach: In this methodology, data is processed at intermediate nodes in order to reduce consumption of critical resources such as energy, memory and computation time etc. This approach increases the network lifetime as it tries to reduce the energy consumption at every node as shown in Figure 1.15.

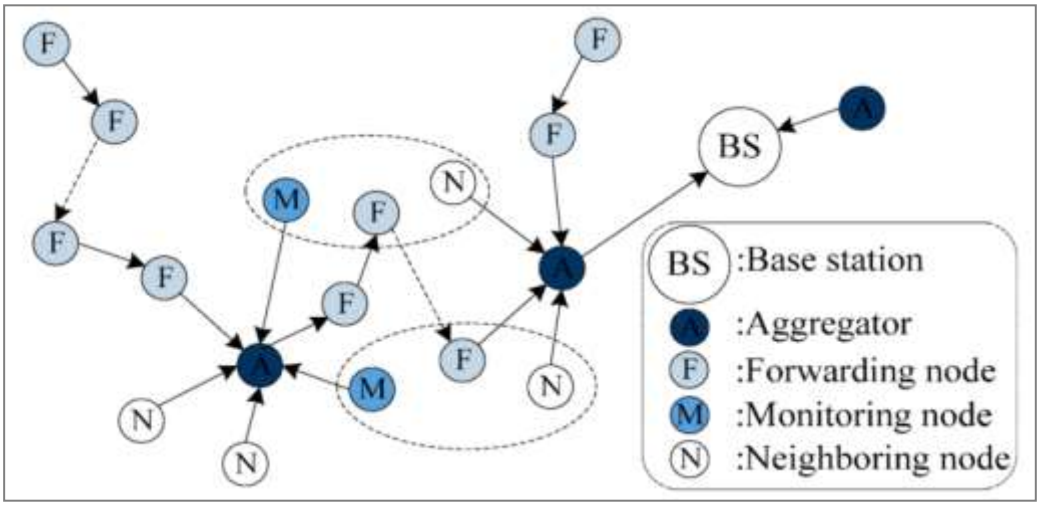


Figure 1.15 In-network Data Aggregation Approach

There are two approaches for in-network aggregation: *lossy aggregation with packet size reduction* and *lossless data aggregation without packet size reduction*. In lossy aggregation, data is gathered from various source nodes and then a group function is applied over the gathered data such as sum(),count(),maximum() and minimum() etc. In this approach, size of the packet is reduced, as only calculated value of aggregate function is inserted into the packet after compression, rather than sending the whole packet of every node. For applications such as forest fire monitoring system, average or maximum temperature reading is required in a timely manner. Hence, a lossy aggregation is required, as it responds in a timely manner to the BS. In lossless aggregation, every packet is merged into single packet without compression.

Hybrid Data Aggregation Approach: The hybrid data aggregation approach is a blend of various approaches. Figure 1.16 depicts the structure of Hybrid data aggregation in which multipath approach, tree based approach and cluster based approach are utilized. In this approach, data aggregation will be performed based on specific network situation, performance statistics and network structure. The sensed information will follow various approaches based on the underlying network topology.

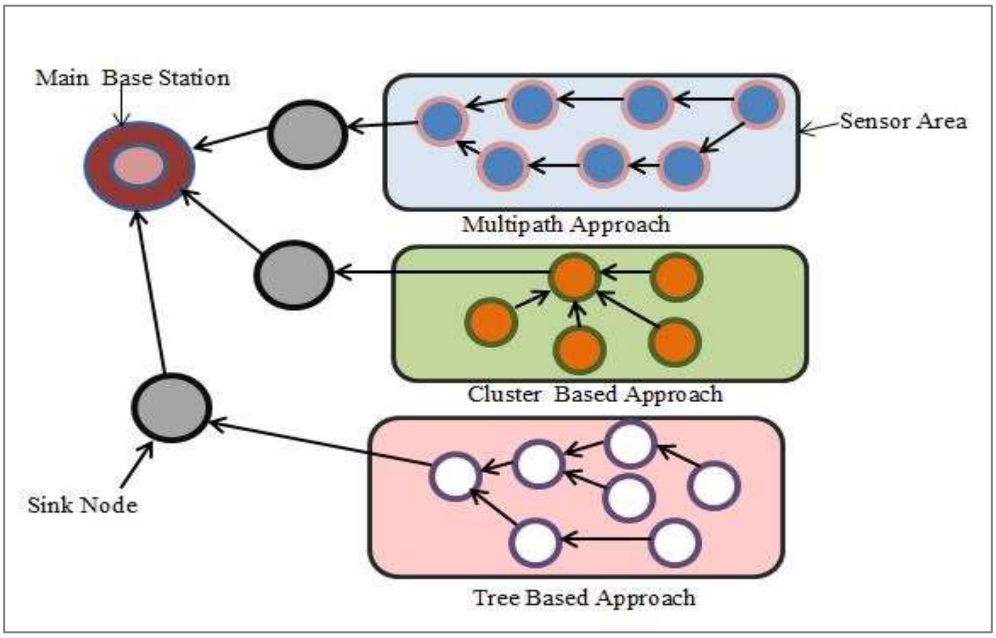


Figure 1.16 Hybrid based Data Aggregation Approach

1.4.2 Queries in Data Aggregation

In data aggregation, a query is generated at BS or cluster head and is sent towards source nodes to sense the data required to answer the query and to revert back with a reply [33], [34]. There are generally three types of queries, which are generated during data aggregation in WSNs namely *simple query, complex query* and *event based query*.

- **Simple query:** In these queries, a predicate clause is used for filtering the sensed data. Aggregate functions or aggregators are not used. Example: SELECT humidity FROM sensors WHERE node=1;

- **Complex query:** In this type of query, aggregate functions and sub-queries are used. Example: SELECT humidity FROM sensor WHERE loc= (SELECT roomno WHERE floorno = 7);

- **Event Driven Query:** In this type of query, data is returned or reported from source nodes at periodic time intervals for event reporting.

1.4.3 Parameters for Designing Data Aggregation Technique

The four aspects namely time bound, energy efficiency, applicability and QoS support are important and must be considered while implementing data aggregation approaches. These are briefed herewith as:

- **Time Bound:** While performing data aggregation, time constraint should be considered along with energy efficiency. A considerable time is involved in aggregating the data and then reporting the aggregated data to the BS. But, in some real life applications, delay is the main constraint. So an aggregation approach should be applied in such a manner that it reduces delay along with energy consumption.

- **Energy Efficiency:** Every sensor should consume same amount of energy in every data-gathering round in an ideal situation but in real world situations, sensor nodes consume different amount of energy for data transmission. Data aggregation technique is energy efficient if it provides the maximum functionality with minimum energy consumption in WSNs. Energy Efficiency (EE) is a ratio of amount of data successfully transferred in a sensor network to total energy consumed to transfer that data. Equation (1.1) is used to calculate energy efficiency.

$$EE = \sum_{i=1}^{n} \left(\frac{Amount\ of\ data\ successfully\ transferred\ in\ a\ sensor\ network}{Total\ energy\ consumed\ to\ transfer\ that\ data} \right) \tag{1.1}$$

where n is the number of sensors nodes deployed in a sensor network.

- **Application Oriented:** Data aggregation is applied as per the application requirements. In some cases, delay is the main constraint, thus lossy aggregation may be applied in this case as data accuracy is not the main concern. In other applications, data accuracy is the main concern, so there is a necessity of lossless aggregation to grasp the minute details of every event.

- **QoS Support**: Energy is one of the main constraints in WSNs. There are several other parameters also known as Quality of Service (QoS) parameters, which are considered while designing and implementing data aggregation approach. Some QoS parameters are delay, data accuracy, bandwidth and throughput etc. The required level of QoS parameters varies according to the application requirement.

1.5 Load Balancing in Wireless Sensor Networks

Load balancing is a method to balance the energy consumption of all nodes to avoid network hole problem. In load balancing, the lifetime of the network does not depend only on the lifetime of weak node but depends on the lifetime of all nodes in the network which helps to increase the network lifetime [35], [36]. Load balancing can be used to extend the lifetime of a WSN by reducing energy consumption. Load balancing using clustering can also increase network scalability. WSN with the different energy levels nodes can prolong the lifetime and reliability of the network.

The sensor nodes if not properly assigned to their Cluster Heads (CHs) for cluster formation, then some CHs may be overloaded with high number of sensor nodes and traffic load. Such overload may increase latency in communication and degrade the overall performance of the WSN [37]. Therefore, load balancing is the most important issue for clustering of sensor nodes particularly, when the sensor nodes are not uniformly distributed. In order to balance the load of the CHs, some sensor nodes are assigned to a CH that may be farther from it. As a result the energy of these nodes may drain out due to long haul transmission to the CH and may quickly die. Therefore, while designing clustering algorithm, one should take care not only the load balancing of the CH but also energy consumption of the sensor nodes to increase the network lifetime.

1.6 Performance Metrics

The metrics have been designed to measure the performance of multipath routing technique in WSNs. These are the measures of:

- Energy efficiency
- Network lifetime
- Packet delivery ratio
- Data accuracy
- Route setup time
- End to end delay

Energy Efficiency: Energy efficiency of multipath routing can be evaluated based on network coverage, connectivity, computation and communication cost and energy consumption. It can also be evaluated based on the number of data packets successfully transferred from source to destination. Average energy consumption for various network operations such as transmitting and receiving data are calculated based on some underlying radio energy model.

Network Lifetime: Network Lifetime in WSN can be defined as 'the time span from the deployment of the network to the instant when the network is considered as non-functional'. Network lifetime can be defined as the time until the first sensor node or group of sensor nodes in the network runs out of energy (battery power) or the time (number of rounds) of network disconnection due to the failure of one or more sensor [38]. It is expressed as Equation (1.2):

$$NL_n^n = min_{v \epsilon V} NL_v \tag{1.2}$$

where the network lifetime NL_n^n ends as soon as the first node fails, with NL_v being the lifetime of node v and V is the node set excluding the sink node.

Packet Delivery Ratio (PDR): It is the ratio of the number of packets received successfully and the total number of packets being transmitted as given by Equation (1.3):

$$PDR = \frac{Amount\ of\ data\ transferred\ successfully}{Total\ amount\ of\ data\ sent} \tag{1.3}$$

Data accuracy: It is evaluated by estimating the difference between the actual data sent and the actual data received at the receiver end. It is defined in different contexts, based on the application for which sensor network is designed. For example, the close estimate of target location at the sink defines the data accuracy in target localization problem.

Data Aggregation Rate (DAR): It is defined as the ratio of amount of data aggregated successfully to total amount of data sensed as given by Equation (1.4).

$$DAR = \frac{Amount\ of\ data\ aggregated\ successfully}{Total\ amount\ of\ data\ sensed} \times 100 \tag{1.4}$$

Route Setup Time: To evaluate the overhead of multipath routing, route setup time can be used. It is defined as the amount of time taken by source node to identify the paths from sender to receiver.

End-to-end Delay: Multipath routing could take a longer path than single path routing due to consideration of reliability and load balancing factors. Delay is the amount of time between the data packets produced at the source node and received at the sink. In other words, it can be a defined as a time difference between sending the data and receiving the data by a sensor node. It includes the delay involved in data transmission, routing and data aggregation. Network congestion can result into delay in transmission and degrades the performance.

Average end-to-end delay is evaluated by taking the average of the time required for the delivery of all surviving data packets (N_{PT}) from the source node to destination node as given by Equation (1.5).

$$AverageDelay = \frac{\sum_{i=0}^{N_{PT}} ReceivingTime_i - SendingTime_i}{N_{PT}} \tag{1.5}$$

1.7 Research Motivation

The reliable WSN requires efficient methods in routing, data aggregation and load balancing. The multipath routing is important in a network for transmitting the data between source and destination in an energy efficient and reliable manner. To avoid network hole problem and to balance the energy dissipation of network, the traffic load needs to be distributed over the multiple routes. The efficient data aggregation and load balancing multipath routing should be based on optimizing the parameters such as energy, delay and network lifetime etc.

Mostly the data aggregation techniques maximize the energy efficiency without paying considerable attention to the data accuracy. The level of data aggregation and reporting frequency of data should be adaptively changed with the varying density of the network and traffic pattern. While splitting and sending the data over the multiple paths in a network using load balancing multipath technique, reliability and energy efficiency should be assured. Whenever the data is lost due to dead nodes or is captured by malicious nodes, then only a part of the data should be revealed. The lost data should be recoverable at destination end. Consequently, there is a need of multipath load balancing routing technique that can assure the required level of QoS parameters and avoids hotspot problem.

In case of WSNs, there are scenarios which need to be modelled as multi-objective optimization formulations, in which objectives can be conflicting and consequently there is a need to choose one of the trade-off solutions. The optimization problem changes with the nature of application, network scenario and input/output parameters. Therefore, there is a need of an appropriate Multi-Objective Optimization technique to resolve such issues. The required technique should be able to manage the multiple conflicting objectives simultaneously and to provide an overall optimal solution depending upon the underlying application requirements.

In this thesis, an efficient multi-objective load balancing and data aggregation multipath routing technique is proposed that will efficiently communicate the data and would be based on energy efficient usage of network resources. A comprehensive investigation has been conducted to study various existing multipath routing techniques in WSNs accomplished by in-depth learning of data aggregation and load balancing techniques. The various load balancing techniques based on bio-inspired, nature inspired and other optimization techniques have been explored for WSNs. After much exploration, a multi-objective load balancing and data aggregation multipath routing technique for WSNs based on underlying requirements has been proposed and designed. The proposed technique automatically manages multiple objectives that may or may not be conflicting in nature in a more efficient way.

1.8 Research Objectives

On the basis of identified research gaps and research motivation, research work entitled "*Efficient Load Balancing and Data Aggregation Multipath Routing in Wireless Sensor Networks*" has been accomplished through various research objectives. The broad objectives of this research work are:

- Study and analyse the multipath routing problem for Wireless Sensor Networks.
- Develop a multi-objective multipath technique for an energy efficient data aggregation and load balancing routing in Wireless Sensor Networks.
- Validation and comparison of proposed technique with existing approaches.

1.9 Thesis Contributions

The attainment of research objectives, summarized above, is resulting into this Thesis. The contributions are summarized as follows:

- A comprehensive investigation has been conducted to study various existing multipath routing techniques in Wireless Sensor Networks by in-depth learning of data aggregation techniques. Along with that, load balancing techniques based on heuristic and meta-heuristic approaches such as bio-inspired, nature inspired and other optimization techniques have been explored for energy efficiency.
- A novel Dynamic Adaptive Hierarchical Data Aggregation (DAHDA) has been proposed with its three variants with significant improvements to perform the data aggregation efficiently for uniform as well as non-uniform networks.

- Cross-layer Energy based Clustering (CEC) technique is proposed to form clusters of sensor nodes in heterogeneous WSNs. It provides a contention free energy efficient data aggregation technique along with the required level of accuracy. The level of data aggregation changes dynamically in accordance to the residual energy.

- An intelligent PSO based Energy Efficient Load Balancing technique (PSO-EELB) has been proposed in which deterministic PSO based clustering approach is utilized. The proposed technique works in three phases: Cluster formation, cluster head selection and data transmission. Data is sent from source to sink via multiple paths using erasure coding.

- A Multi-Objective Load Balancing Clustering (MLBC) technique for load balancing and data aggregation is proposed. Reliability and energy efficiency are considered as objective functions. To demonstrate the validation of the proposed work, the simulation environment has been used. The experimental results demonstrate the superiority of the proposed technique in terms of several QoS parameters.

- Statistical analysis of simulation output has been performed in order to assess the accuracy of the estimated performance indices by using Coefficient of Variation (CoV). The CoV has been calculated for execution time achieved by existing and proposed load balancing (PSO-EELB) and data aggregation (CEC) techniques.

1.10 Thesis Outline

After the Introduction to the research work in this chapter, the rest of the thesis is organized into following chapters:

Chapter 2 Literature Survey

This chapter provides literature review on multipath routing techniques in Wireless Sensor Networks. Moreover, load balancing and data aggregation techniques in WSN have been identified, discussed and analysed. It also presents the summary of research challenges derived from the literature review.

Chapter 3: Dynamic Adaptive Hierarchical Data Aggregation for WSNs

A novel Dynamic Adaptive Hierarchical Data Aggregation (DAHDA) algorithm has been presented in this chapter for evolving, uniform and non-uniform networks while maintaining the data accuracy. The functionality of the proposed technique is divided into four phases: Information Propagation phase, Network Partitioning phase, Cluster Formation phase and Weight based Data Aggregation (WDA) phase (Section 3.2).

. Three variants namely *DAHDA*, *Extended DAHDA (EDAHDA)* and *Modified EDAHDA* (Section 3.3 to 3.5) are presented based on the level of underlying functionalities. DAHDA introduces the concept of weighted sensors. The weights are assigned to nodes on the basis of residual energy and density to decide which nodes will be selected as CHs and which nodes will send their data at certain rounds. Each variant works for different type of application requirement. NS-2.34 simulator has been set up to test and validate the proposed work (Section 3.6). Proposed algorithms are compared with variants of LEACH (Section 3.7).

Chapter 4: Cross-layer Energy based Clustering for Heterogeneous WSNs

In this chapter, Cross-layer Energy based Clustering (CEC) technique is proposed to form clusters of sensor nodes in hexagonal shape. The cluster head is opted from the members of cluster on the basis of the ideal cluster head distance and remaining energy of sensor nodes. In order to make a balance between consumption of energy and the network traffic, the rotation of cluster heads is performed (Section 4.2). The time slots are allocated by cluster head to all member nodes within a cluster on the basis of their remaining energy so that nodes can switch to sleep mode. To reduce the consumption of energy, data aggregation is implemented on the basis of the remaining energy of the cluster head (Section 4.3). In the proposed technique, cluster head selection probability changes dynamically. The performance of proposed CEC technique is evaluated and compared with existing SOEECP, LCM and EEPCA for three-level heterogeneous WSNs (Section 4.4).

Chapter 5: PSO based Energy Efficient Load Balancing in WSNs

This chapter presents PSO based Energy Efficient Load Balancing (PSO-EELB) technique for WSNs. The proposed technique is executed in three phases: *Cluster formation*, *cluster head selection* and *multi-hop intra-cluster and inter-cluster data transmission* (Section 5.2). A deterministic PSO variant is utilized for faster convergence rate. The required number of routing paths and energy consumption of different nodes and paths are calculated. Utilizing maximum value of residual energy and distance, paths are selected and load balancing is performed. Erasure coding is utilized for splitting the data over the multiple disjoint paths. The discovered multiple paths result in balanced energy dissipation in the network and avoids network hole problem. The performance of the proposed PSO-EELB is compared with existing load balancing techniques in the simulated environment (Section 5.4).

Chapter 6:Multi-objective Load Balancing Clustering Technique for WSNs

In this chapter Multi Objective Particle Swarm Optimization (MOPSO) is utilized and novel Multi-objective Load Balancing Clustering (MLBC) technique is proposed (Section 6.2). Two objective functions are defined: Energy efficiency and Reliability. Energy efficiency is based on the average residual energy of CHs. Reliability is based on the transmission cost of inter-cluster routing. Weights are assigned to each edge based on link quality, which is calculated on the basis of PDR. For nodes, weights are assigned based on residual energy and distance. For each CH, sets of neighbours are stored along with their count. Then an encoding scheme is applied which results in cluster formation and cluster head selection. After applying sorting, the node having highest weight is selected as next hop for data transmission. The role of nodes, depending on weights, keeps on changing and network hole problem is avoided. Nodes transmit their information to Cluster Head or Base Station in single-hop or multi-hop manner. A healing function is utilized in order to avoid loops in the generated path. The objective functions are evaluated for each individual. The best compromise solution is selected based on the fuzzy based approach (Section 6.2.4). The experimental results conclude that the proposed technique is better as compared to the existing techniques in terms of various QoS parameters (Section 6.3).

Chapter 7 Conclusions and Future Scope

This chapter summarizes the conclusions drawn in the thesis along with future research directions.

Chapter 2

Literature Survey

Data aggregation and load balancing are important design factors in multipath routing. Data aggregation improves energy efficiency in WSNs by eliminating redundancy in routing packets from source nodes to base station, which otherwise has been present in sensor networks with high node density. The load balancing also plays an important role for energy efficient operation of the underlying network. It distributes the load evenly in a network in order to avoid network hole problem. Therefore, a comprehensive review has been conducted on various multipath routing techniques with emphasis on data aggregation and load balancing techniques in WSNs.

2.1 Multipath Routing: State of the Art

The sensor nodes in WSNs have limited bandwidth, power, memory, processing resources and limited lifetime; which impose challenges in design of efficient communication protocol [2]. Some of the routing techniques have been designed on the basis of single path routing protocol without handling network congestion [39]. Single path routing is prone to failure due to low battery power constraint, which thereby effects network lifetime and reduces throughput in critical situations. Under heavy traffic conditions, multipath routing is treated as an effective way of data transmission to increase resource utilization and network lifetime.

The review on multipath routing appeared from time to time [23], [26], [40]. Gopi [40] and Masdari *et. al* [26] presented a comprehensive taxonomy on the existing multipath routing protocols. The design and operation of different multipath protocols for *Alternate Path Routing* and *Concurrent Multipath Routing* have been compared on the basis of parameters like path dis-jointness, energy efficiency, delay, fault tolerance and reliability. Sha *et al.* [23] categorized various multipath routing protocols on the basis of three parameters: Based on building multiple paths and delivering sensing data, non-infrastructure based and infrastructure based and summarized the design of protocols and evaluation metrics for each category. The comprehensive classification of the multipath routing protocols is shown in Figure 2.1. The methods are classified on the basis of *energy efficient, data transmsion reliabilty, load balancing* and *alternate path routing* [26].

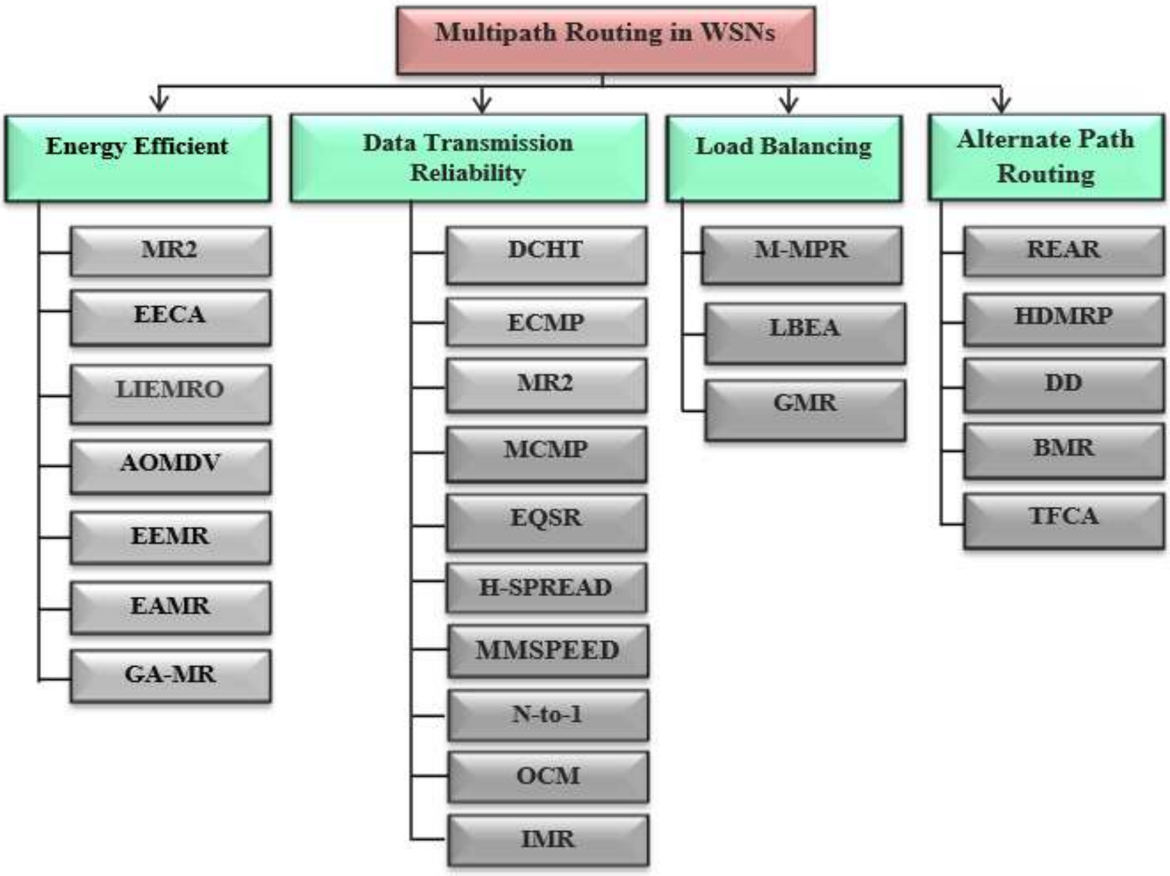


Figure 2.1 Classification of Multipath Routing Protocols

2.1.1 Energy Efficient Multipath Routing Protocols

The energy efficient multipath routing protocols are designed to choose the path yielding minimum energy consumption. The path selection is difficult because nodes with minimum energy drain out quickly i.e. early death of nodes [41]. On the other hand, nodes having higher energy increase throughtput of the network. The objective of these routing protocols is to improve network lifetime, in which network operations is performed in an energy-efficient way. These protocols [42-48] are described as follows:

Maximally Radio-Disjoint Multipath Routing: The Maximally Radio-Disjoint Multipath Routing (MR2) [42] offers required bandwidth to multimedia applications via non-interfering paths. Only one route is created for a particular session using incremental approach. During network congestion, an additional path is created. Only selected nodes will be in active state while other nodes will be switched into sleep mode that leads to energy savings and improved network lifetime.

Energy-Efficient and Collision-Aware Multipath Routing: The Energy-Efficient and Collision-Aware Multipath Routing (EECA) [43] utilizes broadcast nature of wireless communication to avoid collisions among discovered routes. This protocol regulates transmitting power of a node with the assistance of information of node position which reduces energy consumption, improves average packet delivery ratio and reduces average end-to-end delay.

Low-Interference Energy-Efficient Multipath Routing: Low-Interference Energy-Efficient Multipath Routing (LIEMRO) [44] identifies the multiple interference-minimized node disjoint routes between source to destination to improve latency, network lifetime and packet delivery ratio. Further, based on quality of every path LIEMRO distributes traffic over multiple paths through load balancing and reduces route coupling effect.

Energy-Efficient Multipath Routing: This protocol is used to identify multiple node-disjoint paths between source and destination. Adhoc On-demand Distance Vector Multipath (AOMDV) [45] approach is used in this protocol to construct the route using overhearing neighbouring nodes' transmissions, in which receiver constantly remains in the receive state that leads to energy efficiency.

Energy Efficient-adaptive Multipath Routing (EEMR): This protocol [46] uses minimum energy routes in which energy of nodes is drained out quickly and finds alternate paths to save energy consumption.

Energy-aware Multipath Routing: An Energy-Aware Multipath Routing (EAMR) protocol [47] is proposed to improve the energy efficiency of sensor network. The sink initiated route discovery process is used to find the optimal paths for data transmission, which has maximum value of network lifetime.

Genetic Algorithm based Multipath Routing Protocol: A Genetic Algorithm based Multipath Routing(GA-MR) protocol [48] is proposed, in which base station prepares the routing schedule for all the sensor nodes participating for transfer of data from source to destination. Further, a fitness function is proposed find out the minimum distance required to transfer data. This protocol gives better results in terms of network lifetime and energy efficiency.

2.1.2 Data Transmission Reliability Multipath Routing Protocols

To improve the reliability of network, multiple copies of same data are transmitted through different paths. The protocols in this category [41], [49-56] are described as follows:

Delay-Constrained High-Throughput Routing: For multipath video streaming over WSN, Delay-Constrained High-Throughput Routing (DCHT) [49] is used in which multiple disjoint paths are utilized to improve throughput and reliability of network.

Multi-Constrained QoS Multipath Routing: The Multi-Constrained QoS Multipath Routing (MCMP) [50] considers QoS requirements such as reliability and delay while transferring data from source to destination through braided routes. This protocol selects path with minimum number of hops to save energy consumption. Further, MCMP reduces overhead for resource limited sensor nodes and satisfies required level of QoS.

Energy Constrained Multipath Routing: The Energy Constrained Multipath Routing (ECMP) [51] is an extension of MCMP, which uses geo-spatial path selection constraints to optimize energy consumption. ECMP chooses route that has maximum energy efficiency and minimum number of hops and satisfies QoS requirements.

Energy-Efficient and QoS-based Multipath Routing: The Energy-Efficient and QoS-based Multipath Routing (EQSR) [41] protocol uses the concept of service differentiation to transfer high traffic over the network by balancing energy consumption and maximizes network lifetime and reduces end-to-end delay. EQSR predict the best next hop through the paths construction phase using Signal-to-Noise Ratio, node available buffer size and residual energy.

H-SPREAD: Hybrid Secure Protocol for REliable dAta Delivery (H-SPREAD) protocol [52] identifies the extra paths and divides the message into small messages using “one message per node” property. Every time, neighbourhood is informed about identification of new alternate path and increased number of disjoint paths per node thereby improving message delivery ratio.

Multipath Multispeed Routing: SPEED based protocol creates more than one path from source to destination and offers multi-speed transmission. To avoid congestion and reduce the packet loss rate, Multipath Multispeed Routing (MMSPEED) [53] sends packets by considering delay parameter. Further, this protocol offers multiple delivery speed options for timely delivery of packets.

N-to-1 Multipath Routing: This protocol [54] identifies node-disjoint multi paths between source and destination and tree is traversed to transmit data, which improves reliability of network without focusing on energy consumption. This protocol uses one route discovery process to identify multiple paths from every sensor node. Base Station (BS) broadcasts updated route message periodically for discovery process and the node which receives the updated message first time, is considered as a parent node.

Opportunistic Communication Multipath Routing: An Opportunistic Communication Multipath (OCM) routing [55] is proposed based on distributed width-controllable braided, which are working based on the information of neighbour nodes. Further, the information about adjacent nodes is attached with data packet for reliable transmission of data from source to destination. Moreover, less cooperative topology is used to reduce the delay of data transmission.

Immune Based Multipath Routing Algorithm: An algorithm [56] (IMR) is proposed for reliable transmission from source to destination, which improves the fault tolerance of sensor network. This algorithm makes the multipath quickly to improve the network stability in terms of transmission reliability as compared to single path multipath routing.

2.1.3 Load Balancing Based Multipath Routing

The main goal of load balancing based multipath routing protocols is to optimize network traffic and divert traffic through alternate paths when link becomes over-utilized to avoid congestion and bottlenecks. Such protocols [57-59] are described as follows:

Meshed Multipath Routing: This protocol distributes load efficiently and uses route with minimum overhead with respect to packet forwarding technique. The Meshed Multipath Routing (M-MPR) [57] improves throughput and reduce end-to-end delay by distributing traffic more evenly along the meshed route.

Load Balanced Energy-Aware (LBEA): This protocol [58] uses poll-reply communication model to transfer data and balance load efficiently. Minimal hop distance based multiple paths are constructed from source to destination while poll messages are flooded and receiver sends the acknowledgement. Further, mesh structure is used for multi path routing for data reply to reduce network congestion.

Geographic Multipath Routing (GMR) Protocol: A geospatial division (GMR) [59] in duty-cycled underwater based is proposed to reduce the delay of data transmission. In this protocol, three-dimensional underwater network is divided into small space cubes and which are used to transfer data packets by finding the node in target data cube using Geographic Forwarding based on Geospatial Division approach.

2.1.4 Alternative Path Routing Protocols

To increase resiliency to route failures and provide route failure protection, alternative multipath routing can be used. It discovers and maintains multiple paths between the source and destination. The protocols in this category are described [60-64] as follows:

Reliable Energy Aware Routing Protocol: In Reliable Energy Aware Routing (REAR) protocol [60], Routing paths are created using residual energy capacity of every node to improve reliability of data transfer. After succesful transfer of data, REAR provides acknowledgement.

Heterogeneous Disjoint Multipath Routing Protocol: In Heterogeneous Disjoint Multipath Routing Protocol (HDMRP) [61], root neighbors are called sub-roots while sink neighbors are called root nodes. To construct multiple energy-node-disjoint paths, Route Request (RREQ) message propagation is used. Routing table is also maintained by every non-root node to store record of every discovered path. At master nodes, HDMRP uses controlled intersection by permitting these nodes to forward numerous RREQ messages to their neighbors.

Directed Diffusion (DD): This protocol [62] offers support for source node to flood a query towards the intermediate nodes for data transfer. In this, every node is aware of data transfer and interpet data of their intrest without forwarding directly to next nodes.

Braided Multipath Routing (BMR): This protocol [63] transfers data along routes and constructs small aletrnate paths in case of failure. Alternate path does not involve any node from main path for data transmission from source to destination.

Time Frame based Congestion Avoidance (TFCA) Multipath Routing Protocol: A time frame based congestion avoidance multipath routing protocol [64] is proposed for Low power and Lossy Networks. Further, direct acyclic graph is used to find the path with minimum average delay during data transmission. It improves the performance by reducing packer loss ratio.

Apart from the above classification, the multipath routing techniques can also be classified as *infrastructure based, non-infrastructure based and coding based.*

2.1.5 Infrastructure Based Multipath Routing Protocols

This protocol constructs and maintains the multiple paths from source to destination as per topology. Such as, in MST, tree is traversed to discover multiple paths. Message broadcasting protocols are used to design topology independent infrastructure. In this protocol, every sensor node stores list of capable neighbours to transmit data efficiently. Further, node-disjoint multiple paths are identified based on position, location and capability of a node to avoid collision. This type of protocols offers fast and reliable transmission of data as the next hop is selected in advance. This protocol also provides alternative route that can reduce failure recovery time. Infrastructure based multipath routing protocols are categorized into three categories: *energy-aware, hierarchy-based* and *ant-based*. These protocols are described as follows:

Energy-aware Multipath Routing Protocols: This protocol selects the next hop based on remaining energy of neighbouring nodes. The nodes having higher energy efficiency are selected. The load is balanced based on the remaining energy of sensor nodes and thus network performance is improved. To make a neighbouring table, this protocol uses message broadcasting. Every node has a neighbouring table that contains details about signal strength, hop distance and residual energy. This table is used to find out the best next hop. This is reactive type of routing in which path is created only when required to decrease communication overhead. The inter-arrival delay for each data packet is monitored by destination node to check the performance of path. If the value of delay is more than threshold, then new route is created considering that current path is broken.

The energy-aware multipath routing protocol [65] distributes the traffic based on the signal strength and remaining energy of a particular node. To improve load balancing and to save energy, larger amount of load is assigned to under-utilized paths and puts the idle nodes into passive state. The protocol [66] identifies the node-disjoint multiple paths to transfer data. Further, hop distance and energy based link cost function is used to construct multiple paths. In [67], reliable and energy efficient protocol is developed to construct multiple paths using message broadcasting considering energy and node reliability as QoS parameters.

Hierarchy-Based Multipath Routing Protocols: This protocol mainly focuses on discovering the effective multiple paths by creating hierarchical relationship. Further, it reduces communication overhead and improves network lifetime by allocating heavy load to nodes with high residual energy. The update message is broadcasted to construct the hierarchy and the next hop is selected based on hierarchal relationship generated during transfer of data. The protocol [68] uses set of relay nodes to improve reliability of data. Based on the position of node in the hierarchy, relay nodes are selected. Further, the bottleneck nodes are avoided which have minimum energy efficiency to improve throughput of network. In [69], hierarchy-based multipath routing protocol is proposed to improve scalability, network lifetime and reduce energy consumption. Further, message is broadcasted to neighbouring nodes to construct multiple paths. Every node which is receiving the data further transfers the same to neighbouring nodes and treated them as child node iteratively until the every node identifies its child node. Each sender receives an acknowledgement from receiver, which improves reliability of the network.

Ant-Based Multipath Routing Protocols: In this protocol, population-based meta-heuristic is used to construct multiple paths by inspiring from the behaviour of ants. This protocol utilizes the capability of ants to identify the shortest path between then nest and food source. The AntNet algorithm based multipath routing protocol [70] identifies the multiple paths from source to destination. Ants search the path parallelly and *Forward Ants* utilize the meta-heuristic approach on middle nodes to construct the paths. Ant ignores the middle node that has been visited by some other node and search for other nodes. Otherwise ant checks the closeness of sensor node to the sink and pheromone is updated accordingly. On the other hand, based on the information received from forward nodes, local pheromone is updated by backward ants. Yang *et al.* [71] proposed an ant colony optimization and dynamic clustering based multipath routing protocol to identify the multiple path. Based on signal strength and residual energy, Cluster Head (CH) is selected, then the path between sink node and CH is identified. The CH then selects the dynamic route for data transmission.

2.1.6 Non-Infrastructure Based Multipath Routing Protocols

These protocols construct multiple paths to transmit data from source to destination without considering any infrastructure. Every intermediate node uses local information to transmit data instead of finding prior information of next hop. The packet is forwarded towards sink to save energy and to reduce delay. This routing has many benefits: i) no need of path maintenance as path is created while forwarding the packet ii) improved load balancing as data is transmitted to the destination using randomized routing mechanism and iii) utilizes dynamic packet state for data transfer where data packet contains an important information about conditions of network. Such non-infrastructure based protocols are described as:

Geography-based Multipath Routing Protocols: In such protocols, the positions of destination node, neighboring node and source node are used to make routing decision. When data packet is received by intermediate node, it is forwarded to the neighbor node that is closest to the destination. The benefits of routing is: i) no need to store neighboring table, ii) no need of path maintenance as path is constructed using the location of neighbor and destination, and iii) the query is distributed only to limited area instead of complete network. In [43], multipath routing algorithm (EECA) is proposed to design collision free multiple paths. A route discovery message is broadcasted to the neighbors of the node in order to transmit the data packet towards BS. With the help of information of node's location, every node sends the route discovery message. As the information is disseminated to the limited region of sensors, this technique saves more energy. When the source has data packet to transfer then it selects two groups from neighboring nodes fulfilling these conditions: i) all the nodes should be closed to the destination, ii) both the group of nodes should lie opposite to each other and iii) the distance between nodes of two different groups should be more than $R/2$ to construct paths without interference and collisions.

In [57], meshed multipath routing based protocol is proposed to provide many options to transfer the data packet. Network considered for this type routing is collection of static nodes, while some mobile nodes are used to broadcast the information about the location of node. Further, query message is used to identify the meshed routes. A receiver sends a confirmation message after receiving the query. Every receiver node can receive a maximum of two queries to avoid loops in meshed routes. Selective forwarding approach is used to distribute the data packets into multiple paths, which improves load balancing and network lifetime.

2.1.7 Coding Based Multipath Routing Protocols

Coding based routing solves the problems of infrastructure and non-infrastructure based multipath routing. Path construction and path maintenance are two important overhead in infrastructure based protocols. The non-infrastructure based multipath routing protocols has solved the path construction and maintenance overheads while it has a problem of security and energy consumption. Coding based routing uses encryption and decryption techniques to provide the security to the data packets transferred over the network. It saves energy as it does not require transmitting same copy of data packet through multiple paths. At source node, the data packets are split into fragments and redundancy is added to the fragments before transferring through the discovered paths. After receiving the required number of fragments, the original data will be reconstructed at the destination using decoding process. The coding schemes are: *XOR based coding*, *network coding* and *erasure coding,* which possess with different efficiency of compression and decompression.

The erasure coding based multipath routing protocol [43] utilizes on-demand routing algorithm to create a path from source to destination for energy savings. The message is broadcasted over the network to create multiple paths. Destination node sends a route reply message when it receives the request message.

In [72], network coding based multipath routing protocol is proposed to improve data security and reliability. Based on coding opportunity and path reliability, paths are switched dynamically to improve the network throughput. In [73], Robust and Energy Efficient multipath Routing (REER) protocol is proposed for traffic allocation using single optimal path and through multiple paths with XOR based error correction codes. HELLO message is broadcasted over the network to construct multiple paths and neighbouring table is used to select next best hop. Another protocol improves network lifetime using two batteries. The battery recovers during rest and improves energy savings. The comparison of multipath routing protocols on the basis of various QoS parameters are summarised in Table 2.1.

2.2 Load Balancing in Wireless Sensor Networks

Network lifetime of WSNs is an important performance criteria and it depends on energy consumption of sensors [74]. Load balancing is vital as scarce energy is consumed rapidly if the whole traffic is redirected towards a single path. The energy consumption in a network should be minimized and balanced in order to increase the network lifetime. Energy consumption, reliability and network lifetime are important QoS parameters to be accounted during load balancing. The load balancing techniques are classified as *heuristic* and *meta-heuristic* load balancing techniques.

2.2.1 Heuristic Load Balancing Techniques

Gupta and Younis [75] proposed Cluster Based Load Balancing Mechanism (CBLBM), which improves the transmission power to become a member of cluster. The cluster membership depends on the communication cost. Zhang *et al.* [76] presented technique in which comprehensive weight value composed of distance between the member and CH is used. It also provides the selection of cluster member that improves residual energy. This technique does not focus on network lifetime.

Kim *et al.* [77] presented Cluster Based Load Balancing (CBLB) to provides an effective data aggregation. By distributing the load equally among CHs, the reconfiguration is performed to increase the network lifetime. Endre [78] proposed Optimal Scheduling Algorithm (OSA) for load balancing. The time slots are allocated to sensor nodes for transferring the data packets. This technique ensures uniform packet loss probability. Ozdemir [79] presented Hierarchical Data Aggregation based Secure (HDAS) load balancing for heterogeneous WSNs. It considers pseudo sink to increase bandwidth utilization and data accuracy to improve network lifetime.

Table 2.1 QoS based Comparison of Multipath Routing Protocols

Protocols	**Path Disjointedness**	**Energy-Consumption**	**Delay**	**Fault Tolerance**	**Reliability**
H-SPREAD	Node-disjoint	✓		✓	✓
MR2	Node-disjoint	✓			✓
EEMR	Node-disjoint	✓	✓		
AOMDV Inspired	Link-Disjoint	✓	✓		
N-to-1	Node-disjoint				✓
MMSPEED	Partially disjoint		✓		✓
MCMP	Partially disjoint		✓		✓
ECMP	Partially disjoint	✓	✓		✓
DCHT	Node-disjoint		✓		✓
EQSR	Node-disjoint	✓	✓	✓	
M-MPR	Node-disjoint		✓		
LBEA	Partially disjoint	✓			✓
REAR	Node-disjoint	✓			✓
DD	Partially-disjoint	✓		✓	
BMR	Partially-disjoint	✓			✓
LIEMRO	Node-disjoint	✓	✓		✓
EECA	Node-disjoint	✓			✓
HDMRP	Node-disjoint			✓	

Zhang *et al.* [80] proposed Pruning mechanism based Load Balancing Technique (PLBT). It handles the problem of hot spot in WSNs and leads to effective load balancing. The function cost evaluation is based on count of cluster nodes, residual energy, node location and pruning mechanism, Mahdavi *et al.* [81] presented Connected Coverage based Energy efficient (CCE) load balancing that organizes sensor nodes into small subsets. The network connectivity is ensured by adding extra nodes in every subset. Canci *et al.* [82] proposed Threshold based Load Balancing (TLB) to improve energy efficiency through adaptive distributed control strategy and yields energy consumption lower than its threshold value.

Kuila *et al.* [83] presented Clustering Based Load Balancing Algorithm (CBLBA) in which CHs manage the network traffic by considering traffic load as QoS parameter. It is assumed that all sensor nodes produce same traffic load. Sivasanskar and RamaKrishnan[84] proposed Vitality Aware Cluster Head Election method to prolong the network lifetime. The proposed clustering technique chooses a CH based on residual energy and distance upto BS. The node which does not fulfil the threshold value of the parameters considered, is not allowed to participate in the CH selection process.

Deng *et al.* [85] presented Load-balancing Clustering Algorithm for Data Gathering (LCA-DG) to improve energy efficiency in heterogeneous WSN. Energy distribution based dynamic route calculation is used to balance the battery power and to improve network lifetime. In order to balance the load, receiving threshold and optimization threshold are defined. Zeynali *et al.* [86] proposed Fuzzy based Load Balancing (FLB) for distribution of database to improve the network lifetime. Further, vertical partitioning algorithm is used to balance the load efficiently, in which clusters are formed to distribute the traffic in different clusters.

Perillo *et al.* [87] proposed Energy based Load balancing (ELB) technique and investigated the problem of transmission range distribution. It has been identified that hot-spot problem cannot be solved by changing the transmission power of individual nodes. Kim *et al.* [88] developed a Load Balancing technique for Infrastructure (LBI) based WSN that focuses on heterogeneous traffic distributions. The optimal user association consists of load-equalizing, delay-optimal, throughput-optimal and rate-optimal policies to formulate the cost function. The minimum level of connectivity to all spatial locations is desired.

He *et al.* [89] proposed Network Lifetime based Load balancing (NLL) and considered the load balancing factor and size of WSN while creating the Virtual Backbone. The NP-Hard problems for backbone allocation have been investigated. The technique performs effectively in terms of network lifetime and energy. Petrioli *et al.* [90] proposed CONtention based load balancing method (CON) for converge casting in which connectivity holes are detected for data routing and solved the problem of routing around a dead end. It performs well in terms of energy, PDR and end-to-end latency.

Chen *et al.* [91] presented Maximum Connected load-balancing Cover Tree (MCCT) mechanism to balance the load by achieving full coverage. The load of nodes in transmitting and sensing is shared to improve connectivity maintenance and energy efficiency. Baranidharan and Santhi [92] proposed Distributed Unequal Clustering (DUCF) based load-balancing that chooses CH using fuzzy logic. Residual energy, node degree and distance to BS are considered as input variables and size of network as output variable. It performs effectively in terms of network lifetime, energy consumption and load balancing. Nguyen *et al.* [93] presented Energy efficient and Load balanced Distributed Routing (EDR) scheme for WSNs with holes. The approximate polygon of a specific hole is determined and packets are forwarded using the hole covering parallelogram. It improves the energy consumption, network lifetime and reduces the average length of routing paths.

Zuo *et al.* [94] analysed the problem of load balancing in context-aware and proposed a Fuzzy Neural Network Based Load Balancing Optimization(FNNLBO) technique for context-aware WSNs. Authors used greenhouse environment to test the proposed technique and experimental results show that it performs effectively and captures stable state quickly while distributing the load on the network. Liu *et al.* [95] proposed a Variable Weight based Clustering Approach (VWCA) for balancing the load effectively in WSNs to improve energy efficiency. In this technique, weight is assigned to every node based on their residual energy and based on residual energy and distance between nodes, the load is allocated. The proposed technique performs effectively in energy hole avoidance, which improves the network lifetime of network.

Kamal *et al.* [96] proposed a Unbalanced Routing Load Detection and Mitigation algorithm (URLDM) technique for low data rate WSNs based on supervisory routing control to improve packet delivery ratio. Authors used path tagging to visualize the network traffic load and find the overloaded routes to manage traffic effectively using mitigation algorithms. This technique gives better results in terms of network lifetime and PDR. Kim [97] proposed Sub-Network Management based Load Balancing (SNMLB) and an analytical model based an energy-efficient load balancing technique to increase lifetime of network. This technique uses sub-network management to balance the load, which utilizes the nodes simultaneously to improve the energy efficient and network lifetime.

Gherbi *et al*. [98] proposed an adaptive clustering based load balancing technique for effective management of residual energy of sensor nodes. This technique distributes the network data traffic among cluster members to reduce overflow of traffic and transferred to the destination successfully. The experimental results show that it performs effectively in terms of energy efficiency, network lifetime and average data transmission delay. Baroutis *et al.* [99] proposed a load balancing technique to preserve the location privacy of base station and manages the traffic density across the sensor nodes, which makes base station undistinguishable. Further, a trade-off between network lifetime and location privacy is highlighted and experimental results demonstrate that it balances the load effectively while preserving the privacy of base station.

2.2.2 Meta-heuristic Load Balancing Techniques

Swarm Intelligence (SI) based routing protocols and their applications have been explored [100]. The important features of routing protocols are identified as minimal computational and memory requirements, autonomy, energy efficiency and in-network data aggregation.

The min-heap based Energy Efficient Load-Balanced Clustering Algorithm (EELBCA) [101] focuses on load balancing and energy efficiency based on clusters' cardinality. The number of nodes allotted to the CH is used to construct a min-heap. In Parameter-Based Clustering Algorithm (PBCA), communication load of the CHs is incorporated with respect to the BS. The Energy Efficient Fault Tolerant Clustering and Routing (EEFTCR) algorithm [102] uses concept of distributed run time recovery of the nodes to handle sudden failure of the CHs. CH forms Time Division Multiple Access (TDMA) schedule after cluster formation for member nodes. Further, fault-tolerant clustering algorithm is proposed which balances the energy consumption of the CHs.

Differential Evolution based Clustering Algorithm (DECA) [30] improves the network lifetime by avoiding quicker failure of highly loaded CHs. An efficient vector-encoding scheme is used to derive the fitness function. The generation of initial population is restricted by considering the connectivity between nodes and their CHs. Only those children chromosomes are generated which balance the load and hence the proposed algorithm converges faster as compared to basic GA.

PSO based Linear/Nonlinear Programming (LP/NLP) formulations have been presented for energy-efficient clustering and routing by using multi-objective fitness function and efficient particle encoding [103]. A trade-off between number of relays and transmission distance is presented for particle encoding. It balances the load efficiently to save energy and performs efficiently for delivery of total data packets to the BS and network lifetime.

The categorization of routing protocols in WSNs is done based on path establishment, energy efficiency, network structure and computational complexity [104][105]. SI based and classical routing protocols are compared and various performance metrics have been presented. Multi-Objective Particle Swarm Optimization (MOPSO) based clustering [106] is presented to reduce energy consumption, improve network lifetime and optimize the number of clusters. A GA based Clustering Algorithm (CAGA) [83] is used for efficient load balancing that uses communication between CH and nodes to generate initial population and checks its effectiveness through measures such as the rate of convergence, number of active CHs and nodes, energy consumption and execution time.

The Energy-efficient Delay-aware Lifetime-balancing (EDL) data collection technique [107] is designed to decrease its computational overhead and make the large-scale network operations scalable. The technique is integrated with compressive sensing to decrease total traffic cost for gathering sensor readings under loose delay bounds. A load balancing clustering algorithm [74] performs efficiently while nodes having equal load and runs in $O(n \log n)$ time for n nodes. The algorithm is tested in terms of execution time and network lifetime with variable load.

The study on consumption of energy while balancing the traffic suggests that the multiple paths based traffic generation is effective as compared to single path [108]. The Analytical Model for Load Balancing (AMLB) suggests that efficient management of traffic increases the network lifetime.

Hybrid Differential Evolution and Simulated Annealing (DESA) [109] approach performs clustering to select CH and prevents the prior death of CHs. DESA includes a fitness function accounting the residual energy and distance between the CH and the nodes. The Modified Artificial Bee Colony (MABC) based clustering [110] is used for load balancing of clusters resulting in quick convergence and improved search in choice of CHs.

Yinggao *et al.* [111] proposed Artificial Bee Colony (ABC) based data collection technique for WSNs. It permits a small data latency to identify the mobile sink balance: network reliability optimization, mobile path length optimization and data collection maximization. A GA inspired protocol for Congestion Control in WSNs using Trust based Routing (GACCTR) [112] is proposed for distributing the load among different routes between sink and source node. This protocol is efficient in selecting reliable or trustworthy paths regularly. Life Time Aware Routing Algorithm [113] based on Ant Colony Optimization is used to reduce energy consumption. The energy consumption and hop count are integrated with routing choice by designing new pheromone update operator.

ACO based Load Balancing Routing Algorithm (ACOLBR) [114] uses MST for intra-cluster routing and inter-cluster routing by finding optimal and sub-optimal paths. The message's positive feedback is utilized to consider transmission delay, residual energy and propagation distance as the heuristic factor. Ant Colony based Multi-path Routing Algorithm (ACMRA) [115] identifies disjoint multiple paths between nodes and CH. ACMRA is an on demand multipath algorithm where the traffic is distributed over identified multiple paths. ACO based Ad-Hoc On-Demand Distance-Vector (ACO-AODV) [116] protocol is proposed for transmitting data using identified paths simultaneously, which reduces energy consumption, end-to-end delay, buffer overflow and routing overhead.

GA based Construction (GAC) of load-balanced connected dominating set reduces number of participant nodes in communication to improve network lifetime [117]. The workloads of all the dominators are balanced to improve the network lifetime. Load Balanced Clustering Algorithm (LBCA) [118] balances the load among clusters by utilizing gateway to control the network. The CH was used to control different cluster of nodes of dissimilar nature. If a CH gets failed, then Gateway node takes its place.

Multi-Objective Fractional Artificial Bee Colony (MFABC) algorithm [119] is proposed to select the CH optimally using delay, distance and energy consumption based fitness function to control the convergence rate. The GA inspired protocol for Congestion Control in WSNs using Trust based Routing (GACCTR) [112] balanced the traffic among different nodes for different route trust values. General Self-Organized Tree-based Energy-Balance (GSTEB) Routing Protocol [120] utilize ABC to find the shortest path between source and sink based on clustering. Improved Meta-heuristic (ABC) based Energy-efficient Clustering (IMEC) protocol [121] maintains a balance between exploration and exploitation search abilities with least memory requirements. It performs well in terms of PDR, throughput, energy consumption, network lifetime and latency.

The QoS comparison of load balancing techniques is summarized in Table 2.2.

Table 2.2 QoS Comparison of Load Balancing Techniques

Load Balancing Technique	Year	Energy Consumption	Packet Delivery Ratio	Data Accuracy	Throughput	Delay	Network Lifetime
Cluster Based Load Balancing Mechanism (CBLBM)	2003	✓					
Energy Based Load balancing (EBL) technique	2004						✓
Threshold based Load Balancing (TLB) technique	2005	✓					
Clustering Based Load Balancing Algorithm (CBLBA)							✓
Cluster Based Load Balancing (CBLB) technique	2008						✓
Hierarchical Data Aggregation based Secure (HDAS) load balancing	2009			✓	✓		
Pruning mechanism based Load Balancing Technique (PLBT)		✓					
Connected Coverage based Energy efficient (CCE) load balancing technique		✓					
Fuzzy based Load Balancing (FLB) technique							✓
Load-balancing Clustering Algorithm for Data Gathering (LCA-DG)	2010						✓
Optimal Scheduling Algorithm (OSA) based load balancing technique			✓				

Load Balancing Technique	Year	Energy Consumption	Packet Delivery Ratio	Data Accuracy	Throughput	Delay	Network Lifetime
Load Balancing technique for Infrastructure (LBI)	**2012**					✓	
Network Lifetime based Load balancing (NLL) technique	**2013**	✓					✓
CONtention based load balancing method (CON)	**2014**		✓			✓	
Maximum Connected load-balancing Cover Tree (MCCT) mechanism	**2015**	✓					
Distributed Unequal Clustering using Fuzzy logic (DUCF) based load-balancing technique	**2016**	✓					✓
Energy efficient and load balanced Distributed Routing (EDR) scheme	**2017**	✓					✓
Analytical Model for Load Balancing (AMLB)	**2008**						
Ant Colony based Multi-path Routing Algorithm (ACMRA)	**2009**						✓
ACO based Load Balancing Routing Algorithm (ACOLBR)	**2010**	✓				✓	
Swarm Intelligence (SI) based routing protocols	**2011**						✓
GA based Construction (GAC)							✓
Energy Efficient Load-Balanced Clustering Algorithm (EELBCA)	**2012**	✓					
Clustering Algorithm based on GA (CAGA)	**2013**	✓					✓
GA inspired protocol for Congestion Control in WSNs using Trust based Routing (GACCTR)		✓					
Differential Evolution based Clustering Algorithm (DECA)	**2014**						✓
PSO based Linear/Nonlinear Programming (LP/NLP)			✓				✓
Energy Efficient Fault Tolerant Clustering and Routing (EEFTCR) algorithm	**2015**	✓					
Energy-efficient Delay-aware Lifetime-balancing (EDL)		✓					
Hybrid Differential Evolution and Simulated Annealing (DESA)	**2016**						✓

Load Balancing Technique	Year	Energy Consumption	Packet Delivery Ratio	Data Accuracy	Throughput	Delay	Network Lifetime
Variable Weight based Clustering Approach (VWCA)		✓					✓
Sub-Network Management based Load Balancing (SNMLB)							
Artificial Bee Colony (ABC)						✓	
Hierarchical Energy-Balancing Multipath (HEBM)		✓				✓	✓
Preserve Location Anonymity through Uniform Distribution of Traffic volume (PLAUDIT)		✓	✓		✓		✓
Improved Meta-Heuristic based Energy-efficient Clustering (IMEC) protocol		✓					✓
Fuzzy Neural Network Based Load Balancing Optimization (FNNLBO)	**2017**	✓		✓	✓		
Unbalanced Routing Load Detection and Mitigation algorithm (URLDM)		✓	✓	✓			✓

2.3 Data Aggregation in Wireless Sensor Networks

In a sensor network having high node density, the redundancy is caused as many nodes sense same data. This redundancy can be eliminated by using data aggregation approach and thereby improving energy efficiency. The papers [122] [123] summarize the literature and the state of art in the field of data aggregation techniques (DATs) in WSNs. Still the research is persistently growing and many techniques have been reported in the field of data aggregation. The evolution of DATs from 2002 to 2017 is shown in Figure 2.2, which also depicts the focus of study (FoS).

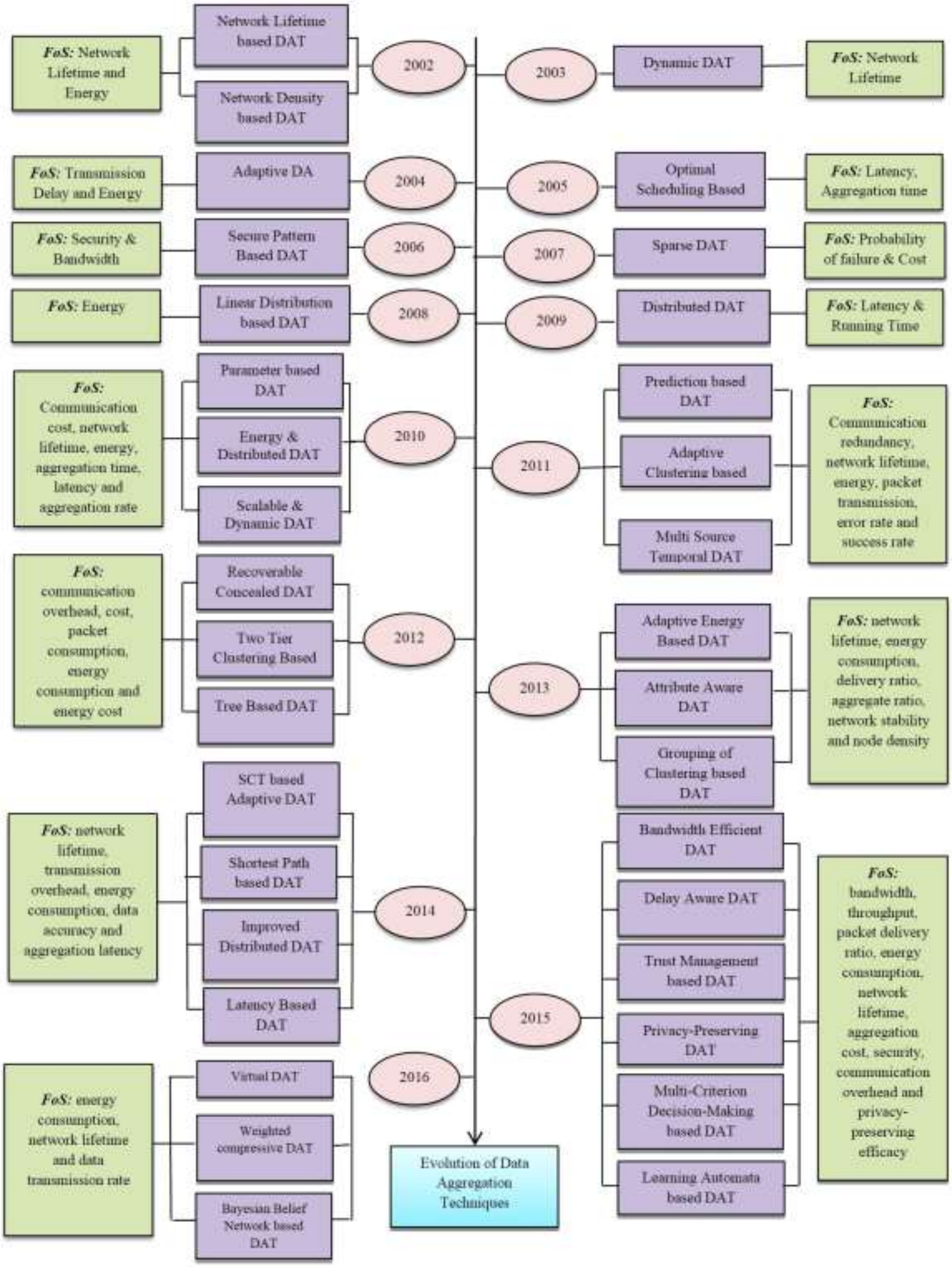


Figure 2.2 Evolution of Data Aggregation Techniques

The classification of various Data Aggregation Techniques (DATs) is presented in Fig. 2.3. The various DATs are described as follows:

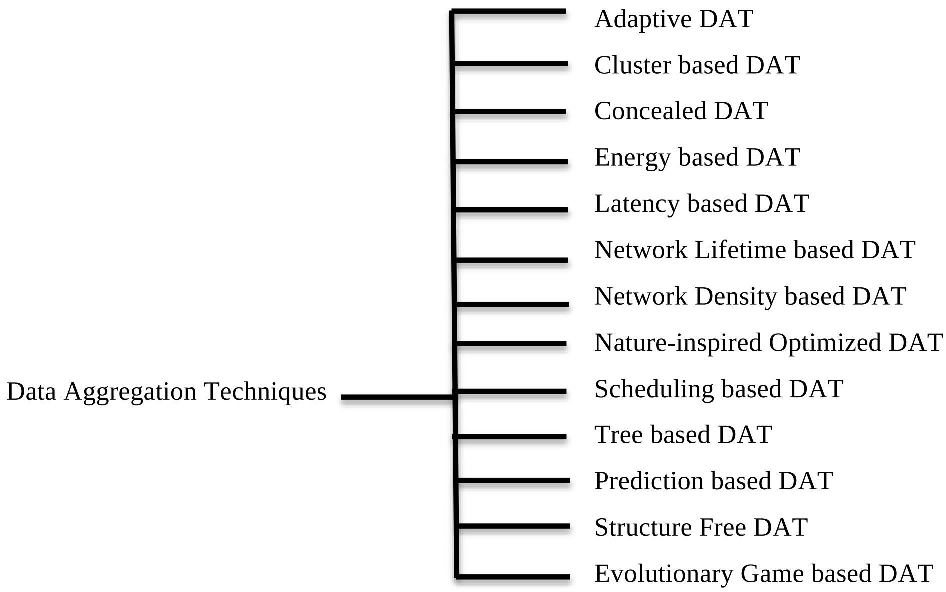


Figure 2.3 Data Aggregation Techniques

2.3.1 Adaptive Data Aggregation Techniques

The latency based energy efficient adaptive DAT [124] that follows staggered active/sleep schedule to forward data in continuous flow in a multi-hop fashion. It uses data prediction along with *More-To-Send* packet, by which a parent node knows whether its child node has more data to send or not.

In the adaptive Application Independent DAT [125], aggregations are decided independent of the application by an intermediate layer between data link layer (MAC) and network layer. It adjusts itself according to the varying network traffic conditions in a timely manner. The cluster based Adaptive DAT [126] mainly focuses on to shift the load of sensor nodes and CHs to the resource enriched sink side and to enhance reliability factor of network. Sink node determines the reporting frequency of nodes and aggregation frequency of CHs. The reliability of event features is represented by Spatial Reliability and Temporal Reliability which are determined by aggregation ratio and reporting frequency respectively.

The node with maximum energy is elected as an aggregator in Adaptive Energy aware Data aggregation Tree [127]. Only the sink and aggregator nodes remain in active mode. Beyond the threshold value for traffic, packets are selected as per parent capacity. Whenever the estimated load exceeds the communication capacity of parent node, OVERLOAD message is transmitted to the network. Shortest path and energy are the factors considered for routing. In the Semantic Correlation Tree (SCT) based DAT [128], whole sensor field is divided into ring like structures. Each ring is further partitioned in terms of sectors. A sector head is selected in each sector, which works as an aggregation node whose task is to report the aggregated data from its respective sector to the BS.

The Adaptive Data Aggregation using Network Coding [29] is an energy oriented cluster based DAT in which sensor nodes of a cluster are divided into *simple relay* and *network coder nodes*. Network coder nodes act as *Aggregator Nodes*. Data aggregation is based on level of data correlation. Whenever the lower data correlation factor is reported in the received packets, then network encoding is performed. Improved energy consumption and network lifetime are resulted due to reduced communication cost.

The DAT [129] is a hybrid approach that considers varying network dynamics and robustness in terms of packet loss. To achieve robustness, virtual rings are used which transfer a packet to multiple nodes. For dynamic data aggregation, Density Aware – Medium Access Control is used. It is specifically designed for event triggered based scenario where events are occurring at unknown time and unknown regions. Yoon *et al.* [130] proposed an adaptive data aggregation and compression technique for solar-power WSNs to improve energy efficiency of sensor network. In this technique, energy budget is determined to find out the value of residual energy and data is aggregated, compressed and transferred to the network. The effective management of data in WSN improves the network lifetime of network.

2.3.2 Cluster based Data Aggregation Techniques

The clustering techniques are very effective in terms of scalability and energy savings [131][132], which are used in number of energy efficient clustering algorithms for WSNs. Some of the routing protocols in this group include Low Energy Adaptive Clustering Hierarchy (LEACH) [133][134][135], Energy Efficient Clustering Scheme (EECS) [136], Energy Efficient Unequal Clustering (EEUC) [137] and efficient Cluster Head Selection approach for Collaborative Data Processing (CHSCDP) [138]. In LEACH for homogeneous

WSNs, nodes elect themselves as CHs randomly and thus every node gets a fair chance to be elected as a CH.

In Hybrid Energy Efficient Distributed (HEED) [139], the CH is chosen based on residual energy of the node. Member nodes choose a cluster to join using minimum value of cost of communication between CH and member nodes. EECS [136] uses the concept of weighted parameters for selection of CH for minimizing the consumption of energy while communicating to the sink. The clusters that are distant from the sink are assigned lesser number of member nodes.

In EEUC [137], the clusters that are more distant from the sink have more member nodes. This reduces energy consumption of communication between cluster members and CH. Additional overheads, due to aggregation of data in larger number of nodes, degrade the network efficiency in multiple hops scenario. Multi-hop Unequal Clustering (MUC) [140] addresses the issue of hot spots. The nodes possessing the maximum amount of energy becomes the CH. CHs which are in close affinity of the sink have smaller cluster in size.

The selection of CH in Hierarchical Cluster Control (HCC) [141] is done on the basis of a weight factor assigned to the nodes. When the energy level of CH is more than a prefixed threshold, then the formation of a new cluster is initiated. The two-level hierarchical model for heterogeneous network [142] suggests that each node can independently elects itself as CH by comparing its energy level with the energy level of neighboring nodes. In Distributed Energy Efficient Clustering (DEEC) [143], CH is selected using probability that depends upon the average energy of the network and remaining energy of nodes. In Developed Distributed Energy-Efficient Clustering (DDEEC) [144], CHs are selected dynamically on the basis of residual energy of nodes.

The Enhanced Developed Distributed Energy-Efficient Clustering (EDDEEC) protocol [145] is based on heterogeneity resulted by incorporating an additional energy level. There are three types of nodes namely *normal, super* and *advanced*. However, the probabilities of the CHs selection are not adjusted by energy level. The Stochastic Distributed Energy Efficient Clustering (SDEEC) routing protocol [146] proposed a balanced CHs selection method for heterogeneous WSNs. The hybrid clustering based data aggregation technique [147] selects adequate clustering technique on the basis of network status to improve data aggregation.

The Link-aware Clustering Mechanism (LCM) [148] proposes Predicted Transmission Count (PTC), a clustering metric, on the basis of node status and link condition. PTC plays an important role in cluster formation. A Stochastic and Optimized Energy Efficient Clustering Protocol (SOEECP) [146] reduces energy consumption and prolong the lifetime in WSNs.

The two-layer architecture [149] generates different routes based on the knowledge of geographic deployment. The routing protocol for WSNs considers the required transmission energy of a selected path from source to destination and the residual energy of nodes [150]. In the Self-Organized and Smart Adaptive Clustering (SOSAC) [151] protocol for WSNs, the fitness value is changed with time by sub-mechanisms. For any potential breakdown in the network, back-up information of routes can be easily extracted from fitness value.

In [152], an energy-efficient clustering based algorithm is proposed for WSNs. It is based on a regional competition, in which nodes can individually detect the events and can form clusters. In [153], it is assumed that the residual energy of nodes is based on a random distribution. The clusters are then formed by ensuring load balancing on the nodes. The CH is selected by considering the relative distance among the nodes and their density.

The distributed data aggregation mechanism [154] uses Clustered Slepian–Wolf Coding (CSWC) problem to choose a disjoint potential cluster to cover the network and to increase the compression gain. A low-complexity joint-coding technique is used to decrease the data redundancy generated by spatial correlation among various clusters. A hybrid DAT named Combined Clustering Based data aggregation [155] is proposed to enable dynamic aggregation by applying various techniques of clustering simultaneously. It provides both static as well as dynamic clustering depending upon the network status.

The Efficient Cluster Head Selection Scheme for Data Aggregation (ECHSSDA) [156] uses the concept of selection of CH and formation of cluster. The concept of an Associate CH is used to take responsibility of CH during the dropping down of energy level of CH below a threshold value. The Two Tier Cluster based Data Aggregation (TTCDA) mechanism [157] applies functions based on additive and divisible aggregation to data packets produced by every node through temporal and spatial correlation. Local Aggregators perform intra-cluster data aggregation. This aggregated data is then combined into one packet using division or additive functions as per application requirements.

The Energy Efficient and Balanced Cluster-based Data Aggregation Algorithm (EEBCDA) [158] used one-hop communication to transfer the data from CH to BS. Network is divided into different sizes of rectangular regions called *swim lanes* and each swim lane is divided into number of rectangular regions called *grids*. A node having maximum energy is elected as CH in every grid. CHs are rotated among the nodes in the grid based on energy consumption. Grouping nodes and Clusters for Efficient Data Aggregation (GCEDA) [159] suggests that grouping of CHs helps to reduce energy consumption and improves network stability without considering heterogeneity of nodes and data accuracy.

The Hierarchical Data Aggregation using Compressive Sensing (HDACS) [160] reduces the amount of data transfer by enabling several compression thresholds dynamically as per clusters size. Instead of configuring only one node as sink, a hierarchy of multi-level cluster is configured for intermediate data collection. DCT Based recovery algorithm is used for underlying domain. Evenly loaded distribution based data aggregation mechanism [30] uses fuzzy logic to select the CH and distribute the load equally among clusters. This approach addressed energy leakage problem of CHs that occurs due to uneven load distribution.

The Bandwidth Efficient Cluster-based Data Aggregation (BECDA) [161] technique focuses on energy consumption for heterogeneous network. As per different energy levels, nodes are classified as *Normal, Advance* and *Super Nodes*. A node with highest energy among the cluster members and a node with the highest number of neighbor nodes with one hop connectivity are considered as CHs.

The CLUstered diffusion with Dynamic Data Aggregation (CLUDDA) [162] is a diffusion and clustering based data-centric technique. It uses in-network processing to aggregate data and combines Directed Diffusion with clustering approach. Rather than sending an interest message to each node, it is communicated only to CHs. It supports dynamic, layered data aggregation and uses try and heal mechanism to restructure the network during path failure. Karen *et al.* [163] explored and compared three mechanisms based on the normalized least mean square, moving average and autoregressive models to forecast local sensed samples in order to decrease wireless data communication. Further, it analyzed the relevance of each estimator depending on the features of the samples acquired in a scenario where the leader and sensors are in communication range of a jump. Mantri *et al.* [164] proposed sink mobility and nodes heterogeneity aware cluster-based data aggregation technique for efficient bandwidth utilization and an increase in network lifetime. It further reduces the energy consumption and improves network lifetime without mobility of the sink. Khorasani *et al.* [165] proposed a neural network based data aggregation technique using three-layered neural network with three layers (input, hidden and output). Further, input layer neurons are resided in every cluster member, cluster head contains the hidden layer neurons and base station contains the output layer neurons. The experimental results show that it aggregates data with minimum consumption of energy.

2.3.3 Concealed Data Aggregation Techniques

Concealed data aggregation is proposed to provide end-to-end encryption and in-network processing. Ozdemir and Xiao [166] proposed Integrity Protecting Hierarchical Concealed Data Aggregation (IPHCDA) technique which uses holomorphic encryption based on elliptic curve cryptography to perform data aggregation in encrypted manner. Chen *et al.* [167] presented Recoverable Concealed Data Aggregation (RCDA) mechanism using homomorphism encryption. This technique uses the concept of recoverable aggregation. There are two variants of RCDA namely RCDA-HOMO and RCDA-HETE for homogeneous and heterogeneous types of networks.

Lin *et al*. [168] proposed holomorphic public encryption system based concealed data aggregation technique i.e. CDAMA for multi-application environment in which BS collect data from aggregated cipher texts for particular application. This technique reduces the impact of compromising attacks and unauthorized aggregations. Zhang *et al.* [169] proposed slicing and mixing technology based Balanced Privacy-preserving Data Aggregation (BPDA) model. It enhances the privacy-preserving efficacy by ensuring that slice can be sent to the nodes that have lower privacy preservation. Node degree and energy are considered as QoS parameters and performance of proposed technique has been evaluated in terms of communication overhead and privacy-preserving efficacy.

Sicari *et al.* [170] presented discrete-time control loop based privacy aware dynamic data aggregation mechanism. It uses Dynamic Data Aggregation with Privacy (DyDAP) functions to provide end-to-end secure data aggregation. Linear Discrete Time control theory is the underlying mechanism. Liu *et al.* [171] proposed an enhanced trust management method for data aggregation using the strength of trust between sensor nodes and BS. Strength of ties between the nodes is calculated from the information coming from other nodes using watch-dog mechanism. This model is used to increase the level of security for data aggregation by classifying the nodes as compromised or trustworthy nodes. Gopikrishnan and Priakanth [172] proposed hybrid secure data aggregation technique to offer high secure data aggregation which implements an end to end symmetric key cryptography for secure authentication using shared public key and it uses hop by hop asymmetric key cryptography with the private keys of each node for data integrity and confidentiality. Zhao *et al.*[173] proposed a lightweight and integrity protecting oriented data aggregation technique for WSNs which has easy, lightweight and secure operability to preserve integrity and data privacy during data aggregation in wireless sensor network. Moreover, compared with existing techniques, proposed technique has verifiable completeness, higher accuracy, less traffic and lower calculation.

2.3.4 Energy based Data Aggregation Techniques

Çam *et al*. [174] proposed Energy efficient Secure Pattern based Data Aggregation (ESPDA) mechanism to reduce duplicate data transmission from node to CH. The task of CH is to evaluate the pattern codes received and to select the nodes for data transmission. The CHs use pattern codes to transfer data from node to BS without using encryption/decryption key distribution.

Xu *et al.* [175] proposed cooperative communication based data aggregation mechanism to solve the traditional problem of cluster based data aggregation which is an NP hard problem. In cooperative communication, a node is deployed with single transceiver and multiple source nodes can send their data to that node. Li *et al*. [176] proposed distributed and latency based data aggregation technique to prevent conflicts among neighboring clusters during transfer. The well-known Minimum-Latency Aggregation Schedule (MLAS) problem has been addressed by an energy-efficient distributed scheduling algorithm. It constructs cluster based tree and its respective CHs for efficient transfer of data among clusters.

Misra and Thomasinous [177] presented LEO: Simple Least-Time Energy-Efficient Routing Protocol with One-Level Data Aggregation. It is time and energy based data aggregation mechanism in which shortest path is used to transfer data. Each node stores information about its neighbour only to save memory. It increases network lifetime as task of data aggregation is performed by next node of source node. The estimated time to reach the BS from that particular node and residual energy information is maintained in the neighbour node table of each node.

Chen *et al.* [178] proposed cost based technique by using in-network aggregation in which data is aggregated at the intermediate nodes from route to the sink using summation (sum, mean, and weighted sum) and extreme (max and min) functions. An algorithm has been presented to find the trade-off between finding a low cost path to the sink and local aggregation of flows in terms of cost taking as a QoS parameter. Chen *et al.* [179] proposed adaptive fault tolerance based data aggregation technique to fulfil QoS requirements of applications and have a trade-off between reliability and energy consumption.

Xiang *et al*. [180] presented greedy heuristic and mixed integer programming based compressed data aggregation technique that considers the concept of compressed aggregation and joint routing. The optimal aggregation tree is obtained to solve both large and small scale problems. Li *et al*. [181] proposed High Energy-Efficient and Privacy-Preserving (HEEPP) data aggregation technique. It provides security through encryption and decryption during data transmission. To encrypt a message, shared key is used and then it is sent to the neighbor nodes in a tree. All received packets are checked to get shared keys and decryption of data. The scheme works in four phases by constructing aggregation tree, dividing/slicing its data packet of particular node into K units, receiving slices of the neighbor nodes and summing up all the data received.

Kuo and Tsai [182] proposed tree based data aggregation mechanism by using relay nodes to construct a data aggregation tree to reduce power consumption. Tree construction with/without relay node is considered as a NP complete problem. Approximation algorithm is proposed based on shortest path mechanism in a distributed fashion.

Chao and Hsiao [183] proposed Structure Free and Energy-Balanced (SFEB) based data aggregation technique. It uses data aggregation along with dynamic aggregator selection scheme. The selection of dynamic aggregators is performed to collect the data from their respective regions then the data is aggregated and reported to the higher level node. Liu *et al*. [184] proposed reliability based data aggregation technique using iRTEDA protocol that combines residual energy, reputation system, recovery mechanism and link availability to improve security. The Beta distribution based reputation function is used to calculate trustworthiness and reputation of nodes. When an aggregator node is identified as compromised node, then re-selection of aggregator node is initiated based on residual energy level and availability of link.

Pourpeighambar *et al*. [185] presented Data Aggregation for Mobile Object using Rate Distortion (RD) with Static Clustering (DAMORD-SC). The predefined clusters are divided into grids and data aggregation is performed through pre-calculated correlation matrices. The data redundancy is reduced due to spatial correlation using RD theory. A target tracking component is installed in each node and each node is aware of location information. Tsai *et al.* [186] proposed delivery delay based DAT by using the concept of buffered packets and configured the parameters in Machine-to-Machine (M2M) networks. A periodic per-hop timing control method is deployed to reduce the energy consumption while there is increase in buffered data or buffering time is longer than aggregation rate.

Krishna and Doja [187] proposed Multi-Objective Meta-Heuristic approach for Energy-Efficient secure Data Aggregation(MH-EESDA). The divide-and-conquer is used to create the secure clusters. The technique works through *formation of clusters, selection of secure nodes* and *energy efficient data aggregation.* Ramachandran and Porter [188] proposed a remote component binding technique named Hitch Hiker to offer sustenance for multi-hop data aggregation. The model is based on priority information associated with *high-priority* and *low-priority bindings*. A meta-data component is used to determine bindings of remote component and to create a network of multi-hop overlay within the payload's unused space of current flows of traffic.

Lee *et al.* [189] proposed restructuring binomial trees and energy-efficient delay-aware data aggregation for energy consumption scenario. The roles are swapped by sensor nodes periodically to make tree balanced and rebuilt Bionomical tree to keep the network balanced. Kim *et al.* [190] presented Energy-aware Hybrid Data Aggregation Mechanism (EHDAM) in which data transmission is controlled adaptively with transmission based on timeout and burst length. The *Energy Hole* problem has been addressed that arises due to the heavy burden near the sink and CHs causing higher node failure rate.

Xiao *et al.* [56] presented Centralized Energy Allocation (CEA) technique based on the Immune-Genetic heuristic to find the effective strategy of energy allocation to maximize the precision of the aggregated data received by the sink. The underlying model of proposed approach assumes the link failure factor for considering unreliability in incoming and outgoing links. Zhang *et al*. [191] presented energy based DAT namely Data Aggregation Supported by Dynamic Routing (DASDR) in which *depth potential* and *queue potential* have been considered. The DASDR utilizes only local information for decisions based on potential dynamic routing mechanism. A queue potential field is implemented to make packets spatially convergent by exploring local information of queue length. Abbasi-Daresari *et al.* [192] proposed sparse random measurement matrix based weighted compressive data aggregation technique in which energy efficient routing trees are formed. Further, Cluster-based Weighted Compressive Data Aggregation mechanism is proposed and applied to every cluster which reduces the number of sensors during every compressive sampling measurement. Kandukuri *et al.* [193] developed pre-filtration method to suppress a huge amount of correlated or redundant data transmissions to maximize battery lifetime of nodes. Further, a data aggregative window and function and relative variation are used to exploit temporal and spatial redundancies in Cluster-Head (CH) nodes. This technique helps to provide reliable data towards the base station. Tang *et al.* [194] proposed bio-level Voronoi diagram based data aggregation technique which integrates routing, Multiple Access Control (MAC) and topology control to improve energy-efficiency.

2.3.5 Latency based Data Aggregation Techniques

Li *et al.* [195] presented distributed data aggregation algorithm namely Cell-Aggregation Scheduling (Cell- AS) which uses physical interference model. It uses time slots to finish the task of aggregation using the ratio between the shortest and longest link length in WSN, which divides the network into cells to alleviate the need of global information. Liu *et al.* [196] proposed approximation based conflict-aware DAT to reduce latency. It used switch beam and steering beam antennas with protocol interference model. Li *et al.* [197] proposed Markovian chain based waterfalls random partial aggregation technique to analyse the Trade-Off Index (TOI) of different applications. It is observed that the trade-offs among data accuracy, delay and energy depend upon the application's requirement.

2.3.6 Network Lifetime based Data Aggregation Techniques

Yum [198] presented distributed data aggregation technique using smoothing approximation function and underlying routing scheme. It combined data aggregation with maximum lifetime optimized routing. Reducing the traffic across the network and applying data aggregation to compress the traffic data extended network lifetime. Wang *et al.* [199] proposed a joint mobility and routing algorithm for aggregation to extend the network lifetime of WSNs. Heterogeneous networks have been considered as collection of both mobile and static nodes. Mobile nodes release the load of highly loaded network within a two-hop radius of BS and enhance the network lifetime. Only limited number of nodes is having the location of mobile node and hence decreases the computational cost.

Shan *et al.* [200] presented the problem of finding a shortest path tree while applying data aggregation with the maximum lifetime. The problem is transformed into a tree in a polynomial time and is solved by load balancing approach. Azad and Sharma [201] presented Pareto-optimal theory based multi-criterion decision-making approach for effective data aggregation. The selection of CHs is based on energy consumption and cluster density.

Asemani and Esnaashari [202] proposed Learning Automata based aGgregation (LAG) algorithm to find the route to transfer data after aggregation. It is capable of adapting itself dynamically in changing environment and to choose innovative paths towards the sink consequently. Every node is trained with Learning Automata (LA) which aids the sensor node to choose its next hop for transferring data towards the sink. Awang and Agarwal [203] proposed data aggregation technique where aggregator nodes were speculated without being dependent on global knowledge. Each data packet generated by a node is identified by an aggregation ID. The nodes that are closer to the aggregator node are favoured for data forwarding process. Ngoc-Tu *et al.* [204] proposed virtual data aggregation trees based data aggregation technique to improve network lifetime. In this technique, a local-tree-reconstruction-based scheduling algorithm is designed to solve *maximum lifetime data aggregation tree scheduling* problem. Further, network lifetime of different number of sensors is analysed and experimental results shows that proposed technique is effective in improving network lifetime.

2.3.7 Nature-inspired Optimized Data Aggregation Techniques

Misra and Mandal [205] proposed Ant colony based DAT to design aggregation tree. The Node Set act as an input to this algorithm and ants are assigned to the nodes. The routes are searched by ants and are communicated via pheromones. The node potential is considered as an estimate of distance to the final destination. Each ant iterates until an aggregation tree is constructed and converges to an optimal solution. Yucheng and Fan [206] presented energy based DAT in which data aggregation tree is constructed to transfer the sensed data to single sink node using the path designed by pheromone in ant colony algorithm.

Lin *et al*. [207] presented network lifetime based Data Aggregation Ant Colony Algorithm (DAACA) technique. The value of energy and amount of pheromone of neighboring nodes can be estimated to dynamically select the next hop for transmission. Based on local and global information, pheromones are adjusted after certain number of transmission rounds. Ho *et al.* [208] presented ladder diffusion and ACO based data aggregation technique to find path for transmission and data relay while preventing the generation of circle routes. Back-up routes are also maintained in case of path failure. Lu *et al.* [209] proposed semi-structured multi-objective tree based dynamic data aggregation protocol. Paul and Gopinathan [210] proposed Ant Colony Optimization (ACO) based hybrid data aggregation mechanism to aggregate fruitful messages through shortest path. Anomaly messages are identified and removed by Support Vector Machine based classification.

2.3.8 Scheduling based Data Aggregation Techniques

The energy consumption is reduced when the nodes are in idle state of listening by switching the nodes into sleeping state at regular intervals or randomly [211]. The algorithms can be *contention free*, *based on contention* or *hybrid* in nature. Two protocols Low Power Down link MAC Protocol (WiesMAC) [212] and Short Preamble MAC Protocol (X-MAC) [213], based on contention, maximize the duration of the state of listening by adaptive preface sampling. Higher energy consumption is incurred for multiple hops routing in WiesMAC. The energy in X-MAC is reduced at both transmitting and receiving ends by reducing the length of preface sampling and by utilizing an adaptive duty cycle.

The contention free protocols, namely energy saving Sensor-MAC (SMAC) and energy saving with adaptive nature Timeout MAC (TMAC) [214] etc. are able to record the sleeping and active period of nodes. There are few protocols that are periodic in nature [214]. The nodes themselves do the synchronization of time slots and the frame. TMAC reduces the duration of listening mode [214]. When no data is received during listening state then node enters into sleeping state but in case of an event driven application, convergence is not obtained. It is solved by Utilized- Media Access Control (UMAC) by harmonizing duty-cycle with the traffic. TDMA [215] solves this issue by modifying the frame length of TDMA. The count of source nodes is the basis of frame length. CH cannot provide more than a single time slot in a frame even if the node demands more.

The distributed scheduling algorithm [216] is based on colouring protocol TDMA-CA and uses the idea of dimensional re-use of the channel. Different colours are assigned for conflicting sensors in the network and distinct slots are arranged for transmissions for every colour. The extra overheads are added for synchronizing activities in the network. The cluster-wide correlated grouping algorithm [126] presents a combined notion for spatial and temporal partitioning of nodes. By making it adaptive in [217], the degree of temporal fusion is administered by the event frequency and fusion's spatial degree is administered by the ratio of fusion at the CH. The whole data fusion process is in the control of sink, which transmits the spatial and temporal degree. In Grid based Spatial Correlation Clustering (GSCC) protocol [218], nodes having highly similar values are placed into same clusters.

Hu *et al.* [219] presented timing control protocol based dynamic DAT to calculate the minimum aggregation period and to reduce energy consumption and latency. As per required quality of data aggregation, aggregation time period can be dynamically changed. An intelligent timer and high level knowledge of the network is considered. Data sink requests for certain number of reports with maximum latency period. If the number of reports is received before the maximum time, then the value of minimum aggregation time is changed.

Yu *et al.* [220] proposed maximal independent sets based distributed data aggregation technique to produce collision-free schedule. It uses greedy strategy to improve time latency. This algorithm works through Construction of distributed aggregation tree and distributed aggregation scheduling using underlying MAC protocol for distributed nature.

The underlying model can adapt itself and can schedule according to the network conditions as some node dies or joins the network. Bagaa *et al.* [221] proposed semi-structured and unstructured based data aggregation mechanism to construct and execute aggregation tree simultaneously. This algorithm works through network organization and simultaneous data aggregation tree construction and data scheduling.

Li *et al.* [222] modified the algorithm by Bo [220] and presented distributed DAT to generate a collision-free schedule. It is implemented by using breadth first search tree at sink node. Jhumka *et al.* [223] proposed collision free Data Aggregation Scheduling (DAS) in which sensor nodes aggregate data and transfer to a sink node by using TDMA. Joo *et al.* [224] presented delivery delay based DAT using node-exclusive interference model. Two types of approaches namely *Myopic* and *Non-Myopic* have been considered for scheduling present state and future requirements respectively.

Bagaa *et al*. [225] proposed Distributed scheme for Integrated tree Construction Algorithm (DICA) along with data aggregation. A collision-free schedule is determined to find the route with minimum time duration to transfer aggregated data to the BS from sensor nodes. During tree formation, a node has multiple available options for the selection of the parent node. Kwon *et al.* [226] presented timeout control based data aggregation mechanism which enables the changes in timeout period dynamically as per currently accumulated data at a node. The routing layer is given prominence over MAC layer that was followed traditionally.

2.3.9 Tree based Data Aggregation Techniques

Tan *et al.* [227] proposed energy based autonomic data aggregation mechanism named Localized Power-Efficient Data Aggregation Protocols (L-PEDAP). The Local Minimum Spanning Tree (LMST) [162] and Relative Neighborhood Graph (RNG) topologies have been used to construct a MST. Based on MST, the position of next adjacent hop is calculated to find the shortest path between sender and receiver to increase the network lifetime. Kumar and Rajkumar [128] proposed approximation based Semantic Correlation Tree (SCT) data aggregation technique.

Virmani *et al.* [127] presented Adaptive Energy aware Data aggregation Tree (AEDT). Hakoura and Rabbat [228] presented Collective Tree Protocol (CTP) based data aggregation mechanism and compared CTP with *Broadcast Gossip* and *Pairwise Randomized Gossip* algorithms. Intanagonwiwat *et al.* [27] proposed tree based data aggregation technique in which greedy approach is used to adjust points of aggregation to increase the amount of path sharing. Data centric reinforcement mechanism is used to construct an energy efficient tree. Lin *et al*. [229] proposed Evolutionary Game-Based Data Aggregation Model (EGDAM). The adjustment of weights between the sensors is considered as a game. An optimal weight distribution is achieved at Nash Equilibrium in weight allocation.

Paul and Gopinathana [210] proposed Ant Colony Optimization (ACO) based hybrid data protocol. Lu *et al.* [209] proposed semi-structured multi-objective tree based dynamic data aggregation protocol in which ACO is used to find the optimal route for data transmission. Concept of sliding window is used for prediction of arriving packets to improve the aggregation probability and to reduce the transmission delay. The convergence of packet transmission is achieved from both temporal and spatial point of view. Villas *et al.* [230] proposed scalability based data aggregation mechanism i.e. Dynamic and Scalable Tree (DST), to create a scalable and dynamic structure and to reduce the problem of load balancing. In this approach, there are *Collaborator*, *Coordinator*s and *Aggregators*. A *relay node* forwards the data towards BS, while *Aggregator nodes* forward the aggregated data from two or more *Coordinator*s. Ibrahim *et al.* [231] presented Bayesian belief network algorithm based data aggregation technique in which data is transferred from aggregator to sink in different parts. Further, OriginPro software is used for experimental purposes and energy is saved due to lesser transmission of data.

2.3.10 Prediction based Data Aggregation Techniques

Wei *et al.* [232] proposed prediction based data aggregation technique by using double-queue mechanism to enable synchronization between sensor node and sink node to reduce cumulative error of continuous predictions. Three algorithms are designed: Grey-Model-based Data Aggregation (GMDA) used to process data series noise, Kalman-Filter based Data Aggregation (KFDA) used to predict accuracy and Combined Grey model and Kalman Filter Data Aggregation (CoGKDA) used to increase prediction accuracy. Jiang *et al*. [233] proposed energy-efficient data aggregation technique in which adaptive mechanism is used to enable or disable the prediction of performance. For rapid propagation of aggregates, cluster to cluster propagation is used instead of node to node propagation.

Meng *et al.* [234] presented clustering based data aggregation technique DACP in which entire network is divided by sink node into different clusters and CH node is selected from every cluster to predict the data coming from different sources. The received predicted data is compared with sensed data to decide that whether to send this data further or not. Energy-Efficient Prediction Clustering Algorithm (EEPCA) [235] is proposed to reduce energy dissipation and prolong network lifetime. This approach is based on heterogeneous WSNs. It is adaptive in nature as it enables the nodes to select the CH according to energy and communication cost. A node with higher residual energy has higher probability to become a CH as compared to other nodes and thereby the energy dissipation is balanced.

2.3.11 Structure Free Data Aggregation Techniques

Yousefi *et al.* [236] proposed structure-free Real-time data Aggregation (RAG) protocol. Two mechanisms are considered for spatial and temporal convergence of packets. Judiciously Waiting policy is used for temporal convergence and Real-time Data-aware Anycasting policy is considered for spatial convergence of packets. Dietzel *et al.* [237] proposed fuzzy logic based approach for structure-free data aggregation technique for applications which require periodic dissemination of information. Fuzzy logic is used to take decisions based on flexible and extensible set of inputs and outputs to satisfy the different requirements of applications. The decision is made on the basis of correlation in the data. Haghighi *et al.* [238] presented a stochastic time-domain method for burst aggregation of data in IEEE 802.15.4 protocol under the assumption of arrival of burst traffic which helps to find a station level information such as failure distributions, collision and delay etc. using bottom-up approach. Ren *et al.* [239] proposed packet-driven timing algorithm based Attribute aware Data Aggregation (ADA) technique. The data has been gathered from different heterogeneous source nodes and therefore attributes of packets may differ. An attribute aware data routing is proposed that makes the packets with the same attribute spatially convergent.

2.3.12 Evolutionary Game based Data Aggregation Techniques

Engouang *et al.* [240] presented game based data aggregation mechanism by considering ZigBee and Self-Fulfilling Belief (SFB) for two player's game theory. It provided collision free communication to the nodes. Claude Shannon's information theory is considered for maintaining privacy. Based on SFB, every player plays its best with its energy parameter and also assumes that other players are also trying their best to win the game. EGDAM [229] is a data aggregation technique used to map the cooperation and competition in aggregation process into games through theoretic model to improve resilience. Based on this technique, Evolutionary Game based Adaptive Weighing Algorithm (EGWDA) is proposed for pixel level data aggregation. The adjustment of weights between the sensors is considered as a game. An optimal weight distribution is achieved when a Nash Equilibrium is achieved in weight allocation.

The comparison of DAT in reference to QoS parameters is presented in Table 2.3.

2.4 Multi-objective Optimization in WSNs

The aim of Multi-objective optimization algorithms is to optimize the conflicting goals in WSNs simultaneously[241][242][243]. Most of these algorithms try to optimize the consumption of energy while considering other conflicting goals together. Jin *et al.* [244] presented a technique for cluster formation to optimize the number of CHs and communication distance using Genetic Algorithm (GA). The fitness function is based on the total distance between the BS and CHs, total distance between the CHs and non-CHs nodes and the difference between the number of all nodes and the number of CHs. Weights are assigned to the input parameters, which transform the multi-dimensional optimization problem to single-dimension. Ouchitachen *et al.* [245] proposed an Improved Multi-Objective Weighted Clustering Algorithm (IMOWCA) in which network is divided into number of clusters and a set of sensors is selected based on residual energy. The major objective is to provide a good level of communication in a network for which a Base Station Genetic Algorithm (BGA) is proposed.

Table 2.3 Comparison of Various Data Aggregation Techniques in Terms of QoS Parameters

Data Aggregation Technique	Subtype	Energy	Delay	Packet Delivery Ratio	Degree of Data Aggregation	Reliability	Network Lifetime	Data Accuracy	Transmission Cost	Node Density	Computational Complexity	Throughput	Communication Cost
Adaptive DA	Energy and Latency	✓	✓	✓									
	Application	✓	✓		✓								
	Cluster					✓							
	Energy	✓	✓				✓						
	SCT							✓					✓
	Distributed and Energy						✓						
	Robust	✓	✓										✓
Cluster based DA	Adaptive					✓							
	Distributed												✓
	Hybrid	✓		✓	✓		✓						
	Head Selection Scheme	✓					✓						
	Two Tier Clustering	✓											
	Balanced Clustering	✓					✓						
	Group of Clusters	✓											
	Energy			✓			✓		✓				
	Adaptive and Energy						✓						
	Bandwidth	✓-										✓	
	Load Distribution						✓						
	Bandwidth and Energy	✓											
	Diffusion & Clustering		✓								✓		

Data Aggregation Technique	Subtype	Energy	Delay	Packet Delivery Ratio	Degree of Data Aggregation	Reliability	Network Lifetime	Data Accuracy	Transmission Cost	Node Density	Computational Complexity	Throughput	Communication Cost
	Hierarchical	✓											
Concealed DA	Hierarchical					✓							
	Recoverable								✓				
	Multiple-Application	✓				✓							
	Trust management										✓		
	Balance privacy-preserving					✓							✓
	Privacy Awareness	✓							✓				
Energy based DA	Secure Pattern	✓										✓	
	Hierarchical	✓					✓						
	Co-operative						✓						
	Distributed and Latency	✓	✓										
	Compressed	✓											
	Data Accuracy						✓						
	Tree Based	✓											
	Privacy Based							✓			✓		✓
	Design of Structure based	✓			✓						✓		
	Game based		✓	✓								✓	
	Reliability based	✓					✓	✓					
	Static Clustering	✓						✓					
	Delivery Delay based		✓				✓						

Data Aggregation Technique	**Subtype**	**Energy**	**Delay**	**Packet Delivery Ratio**	**Degree of Data Aggregation**	**Reliability**	**Network Lifetime**	**Data Accuracy**	**Transmission Cost**	**Node Density**	**Computational Complexity**	**Throughput**	**Communication Cost**
	Multi-objective meta-heuristic	✓				✓							
	Remote component binding	✓	✓										
	Delay-aware	✓	✓										
	Heterogeneous network	✓						✓					
	Dynamic	✓											
Latency based DA	Adaptive and Energy	✓	✓	✓									
	Distributed		✓										
	Conflict aware		✓										
Network Lifetime based DA	Energy	✓					✓						
	Linear Programming	✓											
	Precision Constrained	✓					✓						
	Distributed				✓		✓						
	Shortest Path based												
	Multi-criterion decision-making	✓	✓										
	Learning Automata based	✓					✓						
	End-to-end delay	✓		✓									
Network Density based DA	Tree	✓											
	Cluster					✓							
Nature-inspired Optimized DA	Search Space based												
	Energy	✓											
	Network Lifetime	✓		✓							✓		

Data Aggregation Technique	Subtype	Energy	Delay	Packet Delivery Ratio	Degree of Data Aggregation	Reliability	Network Lifetime	Data Accuracy	Transmission Cost	Node Density	Computational Complexity	Throughput	Communication Cost
	Ladder Diffusion	✓											
	Tree		✓		✓								
	Hybrid							✓					✓
QoS based DA	Energy	✓	✓					✓					
	Waterfalls Random partial aggregation	✓	✓					✓					
	Network Lifetime and Fault Tolerance						✓						
	Time & Energy	✓	✓				✓						
	Cost				✓								
Scheduling based DA	Dynamic	✓											
	Distributed		✓										
	Semi-Structured and Unstructured		✓				✓						
	Improved Distributed		✓										
	Collision Free		✓										
	Delay based		✓										
	Integrated tree construction	✓	✓										
	Timeout Control Scheme	✓	✓										
Tree based DA	Energy	✓					✓						
	Adaptive and Energy	✓					✓						
	Hybrid							✓					✓

Data Aggregation Technique	Subtype	Energy	Delay	Packet Delivery Ratio	Degree of Data Aggregation	Reliability	Network Lifetime	Data Accuracy	Transmission Cost	Node Density	Computational Complexity	Throughput	Communication Cost
	Collective Tree Protocol								✓				
	Energy oriented cluster based	✓											
	Greedy approach	✓	✓										
	Prediction based							✓					✓
	Structure-free real-time	✓	✓	✓	✓								
	Evolutionary game-based		✓	✓									
	Dynamic		✓	✓	✓								
	Scalability				✓								✓
Prediction based DA	-	✓						✓					✓
Structure-free DA	Real time	✓	✓	✓	✓								
	Stochastic time-domain			✓			✓						
	Attribute-aware			✓						✓			
Evolutionary game-based DA	-		✓		✓								
Hybrid DA	-						✓						

Hussain *et al.* [246] presented a technique for cluster formation based on GA, in which fitness function is calculated in terms of the total distance between all the BS and nodes, total distance between CHs and non-CH nodes and distance between CHs and BS, standard deviation of the distances from CHs, the energy dissipation for the delivery of gathered data in CH to BS and the number of transmission. A weighting coefficient is assigned to each parameter in fitness function, which is adaptively updated in iterative manner and therefore the multi-objective optimization problem is transformed into single-objective problem.

Peiravi *et al.* [247] proposed an algorithm which is based on Pareto-optimal method to address multi-objective clustering problem. It results in generation of number of optimum solution points rather than a single optimum solution. The network lifetime and the number of the hop between nodes and BS are optimized using centralized control. Ozdemir *et al.* [248] presented a Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) to optimize the energy conservation and to maintain required coverage in clustering based WSNs. Lu *et al.* [249] presented Jumping Particle Swarm Optimization (JPSO) using an adaptive double layer encoding scheme to obtain Pareto-optimal solution. Jameii *et al.* [250] presented Multi-Objective Optimization Coverage and Topology Control(MOOCTC), in which conflicting objectives namely number of active sensor nodes, coverage and network connectivity were optimized. The domain-specific knowledge is considered while obtaining the required solution along with LA to adapt the mutation and crossover rates dynamically without external control.

Prasad *et al.* [251] proposed Multi-Objective Particle Swarm Optimization Differential Evolution (MOPSO-DE) approach for efficient clustering of nodes in WSNs. The CHs are selected based on fitness functions in terms of transmission energy and distance between source and destination. Residual energy and signal strength of neighbouring nodes are also considered while routing the data and selection of CH. Attea *et al.* [242] utilized Non-dominated Sorting Genetic Algorithm-II (NSGA-II) to take decision regarding location of mobile sensor nodes to provide longer network lifetime and high coverage by maximizing both number of detected targets and network lifetime. Xue *et al.* [252] utilized a Multi-Objective Differential Evolution (MODE) algorithm in which latency and energy are considered to generate optimal routes between source and destination. Konstantinidis *et al.*

[253] presented an optimized solution for the deployment and for determining the energy levels of transmission of sensors based on MOEA/D.

2.5 Research Challenges

Although research in WSNs is evolving over the past few decades, still there are number of research issues that need to be addressed [254][255]. Few of these research areas are listed as follows:

- **Route Discovery and Maintenance:** The main issues in multipath routing are: how many paths are sufficient to provide reliability and fault tolerance; how paths are discovered and maintained and based on what parameters optimal paths are discovered. Based on the signal strength and residual energy of node, intermediate node selects the next node to forward the data to the destination. It is also checked that the selected node should not be malicious node based on trust factor; otherwise it will cause transmission delay and will hamper confidentiality of data. Due to continuous topological changes, route discovery is more challenging in WSNs. Multipath routing protocol selects another optimal path in case of path failure. Path maintenance is a complex and continuous task which needs updated information of network to be maintained.
- **Efficient Routing:** In order to provide energy efficient data dissemination, routing plays a vital role. Under high network load conditions, traffic can be split over multiple paths which are discovered during route discovery phase using data dissemination algorithms. But, the concurrent use of multiple alternate paths results in inter-path interference which results in degradation of performance of the network. When multiple paths are concurrently used, then there must be some data aggregation procedure to merge the data together before sending to the BS or Sink for efficient utilization of resources like bandwidth, time and energy etc.
- **Redundancy:** Sensor nodes sense the similar data in densely deployed regions and forward it to the Sink node. Sink node and CHs dissipate their energy unnecessarily in processing this redundant data. So, to improve the network lifetime, there is a need of redundancy elimination. Data aggregation functions reduce redundancy but may increase delay as the data from the nearer sources may have to be held back at an

intermediate node in order to be aggregated with data coming from sources which are far away. In the worst case, the delay in case of aggregation will be proportional to the number of hops between the sink and the farthest source.

- **Quality of Service (QoS):** QoS demands based on applications while performing data aggregation is an open research issue in WSNs. The various QoS parameters which can be considered are energy, delay and reliability etc. Some researchers made efforts to provide the required level of QoS parameters based on the underlying application.
- **Data Accuracy, Reliability and Fault-Tolerance**: Data aggregation accuracy means how closely the aggregated data matches with the original sensed data. In data aggregation process, as the level of aggregation increases, the data accuracy of the sensed data decreases. Due to time-varying network conditions and dynamic topology, providing reliability and fault tolerance is a very challenging task in data aggregation. Depending upon the power or energy available several methods have been used to determine the residual energy of the network for their self-organization.
- **Power-saving Modes of Operation:** A sensor node enters into a sleep mode, when there is reduced or no activity running. This will save the energy and can prolong the network lifetime. The transition management for these nodes is open to research.

2.6 Chapter Summary

In this chapter, the existing research work on multipath routing in WSN is presented. The existing work has been analysed in various ways like types of multipath routing protocols and QoS based comparison of multipath routing protocols. The load balancing and data aggregation techniques and their sub types in detail are explored. Key discoveries in these areas occurred after 2001 are found. The existing research is presented in chronological order. The design of data aggregation techniques for a specific network application will depends on the aims and objectives.

Chapter 3

Dynamic Adaptive Hierarchical Data Aggregation for Wireless Sensor Networks

The WSNs sense and collect the various types of attributes such as light, temperature, humidity and seismic activities etc. for processing, analysis and decision making. This captured data is used to detect occurrence of certain events and to take the appropriate actions accordingly. The efficient methods in routing, data aggregation and localization are needed to design a reliable WSN. In this work, the problem of data aggregation has been considered to improve the network lifetime. The main goal of data aggregation algorithm is to minimize the amount and size of data transfer. These techniques decide which data reading to be transmitted and what is the most appropriate timing to transmit the collected data. Since aggregation algorithms can decide not to send some data records at certain time periods in order to save battery power, the aggregated data may not reflect the actual recorded values. This might affect the accuracy of the decisions, taken by the system. Data accuracy of the aggregated data is, therefore, a major concern in the field of data aggregation algorithms to ensure accurate system decisions.

The WSNs can be deployed via uniform and non-uniform distribution of nodes. In uniform networks, some nodes die with time when their energy is exhausted, which may result into a non-uniform distribution at some point. WSNs can, hence, be networks of evolving nature since the distribution of the sensors in the networks may change with time. The values of the sensed data can, also, change distributions. The data readings can either change gradually or have sudden bursts. Sudden changes in values of specific sensors of the network can happen due to the occurrence of any event, for example, natural hazards in field applications or sudden medical disorder in health applications can result in such changes. These changes would be of interest to capture their data values in order to ensure that no important events in the sensed field are missed. Therefore, a clustering based data aggregation is required which can provide the required level of data aggregation along with adequate accuracy.

A novel Dynamic Adaptive Hierarchical Data Aggregation (DAHDA) algorithm has been presented for evolving, uniform and non-uniform networks while maintaining the data accuracy. The functionality of the proposed technique is divided into four phases: Information Propagation phase, Network Partitioning phase, Cluster Formation phase and Weight based Data Aggregation (WDA) phase. The proposed algorithm is able to handle sudden bursts in the data by recording the information in the area of interest for the whole event duration. The algorithm performs well in terms of extending the lifetime of the network, maintaining the original distribution of the sensors as long as possible, maintaining the accuracy of the sensed data and being able to detect and handle sudden bursts of data.

The proposed data aggregation technique DAHDA has been modified into Extended DAHDA (EDAHDA) and Modified EDAHDA based on the level of underlying functionalities.

3.1 Preliminaries for DAHDA

The new algorithm, Dynamic Adaptive Hierarchical Data Aggregation (DAHDA) is proposed to reduce the energy consumption of sensors and increase the network lifetime without critically affecting the data accuracy. DAHDA extends the functionality of LEACH-C [134] as it introduces the concept of weighted sensors. The weights are assigned to the clusters to decide which nodes will be selected as Cluster Heads (CHs) and which nodes will send their data at certain rounds. The assignment of weights to nodes is based on the residual energy and density. The preliminaries for the development of DAHDA has been reviewed and summarized in this section. These preliminaries include the review on:

- Network model
- Energy model
- LEACH algorithm
- Density based clustering

3.1.1 Network Model

A WSN is monitored and commanded by a sink node or Base Station (BS). Sensors nodes are divided into several clusters after being deployed. The existing literature shows that a cluster-based WSN has several advantages such as efficient energy management, better scalability of Medium Access Control (MAC) or routing etc. Within each cluster, a Cluster Head (CH) is elected which is responsible for collecting and aggregating sensed data from other nodes within the same cluster. A CH forwards the aggregated results to the BS. This technique saves energy, since not all sensors are required to communicate with the BS directly. In a homogeneous WSN, CHs behave as normal sensor nodes. In a heterogeneous WSN, CHs act as powerful nodes, in which various types of nodes having varying capabilities are deployed. In this proposed work, each node is equipped with GPS unit to provide the location information to the BS.

3.1.2 Energy Model

An energy model designed in physical layer is used for calculating energy loss in each sensor node for various network operations while communicating with the other nodes[256]. Two channel propagation models which are used are the *free space* (d^2 power loss) for the purpose of one-hop or direct transmission and the *multipath fading channel* (d^4 power loss) for packet transmission via multi-hop. Thus, the energy exhausted for transmission (E_{TX}) of l-bit packet over distance d is calculated, as given by Equation (3.1):

$$E_{TX}\,(l,d) = \begin{cases} lE_{elec} + l\mathrm{E}_{fs}d^2, & d < d_0 \\ lE_{elec} + l\mathrm{E}_{mp}d^4\,, & d \geq d_0 \end{cases} \tag{3.1}$$

where E_{fs} is free space energy loss, E_{mp} is multipath energy loss. The electronics energy (E_{elec}) includes many factors such as modulation, digital coding, the filtering and the spreading of the signal etc. d is distance between source node and destination node and d_0 is the crossover distance given by Equation (3.2):

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

The energy dissipated for the radio to receive (E_{RX}) this message is given by Equation (3.3):

$$E_{RX}(l) = lE_{elec} \tag{3.3}$$

Therefore, the transmission power and receiving power energy levels are designed in physical and MAC layer of the WSN.

3.1.3 LEACH Algorithm

Heinzelman *et al.* [133] presented a data aggregation algorithm based on cluster routing named Low Energy Adaptive Clustering Hierarchy (LEACH) algorithm. This approach works in terms of rounds, where each round is further divided into two phases, namely, *setup phase* and *steady state phase*. In the first phase i.e. setup phase, based on a threshold, *k%* of total number of nodes (n) are selected randomly to be CHs in a uniform manner as given by Equation (3.4):

$$T(n) = \begin{cases} \dfrac{k}{(1 - k(r_c \, mod \left(\frac{1}{k}\right))}, if\ n \in S, \\ 0, \quad otherwise \end{cases} \tag{3.4}$$

where k denotes the required number of CHs, current round is represented by r_c and S represents those set of nodes which have not been chosen as CHs in the last $\frac{1}{k}$ rounds. It gives an assurance that a node which is selected as CH, will not be selected in the following rounds until all remaining nodes become CHs in a WSN. This constraint results into fair and uniform energy consumption, hence, results into increased network lifetime. In steady state phase, actual data transmission takes place. In LEACH, non-uniform networks are not considered as CHs are selected randomly and uniformly. After the selection of all CHs, clusters are formed dynamically based on distance. In the steady state phase, each non-CH node will join the cluster of the nearest CH. According to the Time Division Multiple Access (TDMA) schedule, each CH gathers the data from all nodes in its cluster as shown in Figure 3.1.

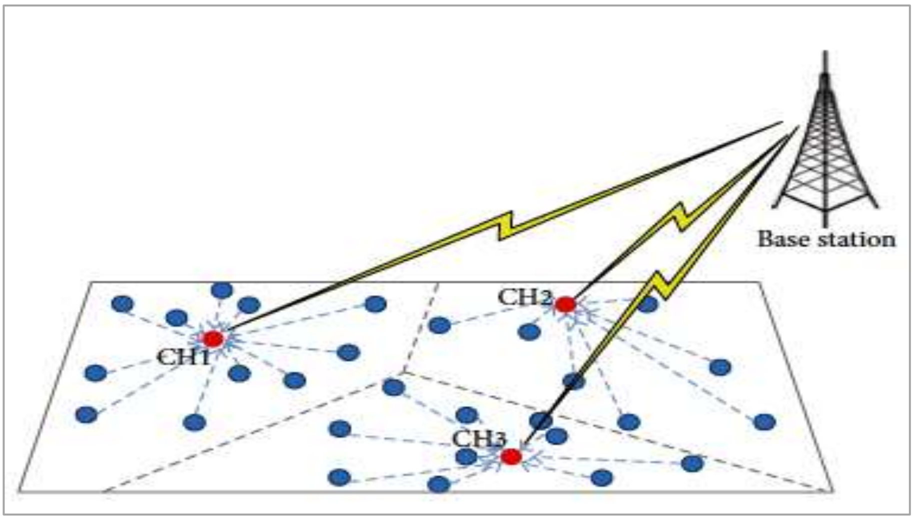


Figure 3.1 Routing Mechanism of LEACH Protocol

3.1.4 Density based Clustering

Density based Clustering (DC) algorithm detects those areas where the objects are densely populated in the regions. Low-density regions distinguish these clusters from each other. DC based approaches perform significantly better as compared to the hierarchical and partitioned algorithms. The arbitrary shaped clusters can be defined as well as can also effectively identify noise points. However, two input parameters need to be specified which are difficult to estimate: the range of the cluster (R_1) and minimum number of points needed within a cluster ($MinNodes$). The general idea of DC approach [257] is as follows:

Definition 1 (*R_1 locality of a point*): R_1 locality of a point x is represented by $R_1(x)$. It is defined as a set of objects i.e. A, as given by Equation (3.5):

$$R_1(x) = \{\mathrm{y} \in \mathrm{A} \mid \mathrm{distance(x,y)} \leq R_1\} \tag{3.5}$$

Definition 2 (*Directly density-accessible*): A point x is directly density-accessible from point y wrt. R_1 and $MinNodes$, in the set of points A, if Equation (3.6) and Equation (3.7) are satisfied:

$$x \in R_1(y) \tag{3.6}$$

$$|R_1(y)| \geq MinNodes \tag{3.7}$$

Definition 3 (*Core object and edge object*): A point is defined as a *core point* if it satisfies Equation (3.7). An *edge point* is density-accessible from another core point and not a core point itself.

Definition 4 (*Density - accessible*): A point x is density accessible from a point y wrt R_1 and $MinNodes$ if there is a chain of points $x_1 \dots \dots . x_n;$ $x_1 = y;$ $x_n = x$ such that x_{i+1} is directly density accessible from x_i.

Definition 5 *(Density-connected):* A point x is density-connected to a point y, if there is a point $z \epsilon A$ such that x and y are density-accessible to z.

Definition 6 *(Noise):* Let $C_1 \dots . . C_{nCluster}$ be clusters of the data set A wrt. R_1 and $MinNodes$. Here, $nCluster$ denotes number of clusters. Noise is defined as the set of data points in A which does not belong to any of the clusters C_i .

$$noise = \{x \in A \mid \forall i : x \notin C_i\};\ i = 1,2, \dots ., nCluster$$

Definition 7 *(Cluster):* Let A denotes a dataset of data points. A cluster C is a non-empty subset of A wrt. R_1 and $MinNodes$ if the following constraints are satisfied:

(1) Maximality: $\forall x;\ y, if\ x \epsilon C$ and y is density-accessible from x wrt. R_1 and $MinNodes$, then $y \in C$.

(2) Connectivity: $\forall\ x;\ y \epsilon C,\ x$ is density-connected to y wrt R_1 and $MinNodes$.

The process of DC algorithm is summed up in Algorithm 3.1.

Algorithm 3.1 Density based Clustering Algorithm

Function DC(Dataset $A, R_1, MinNodes$)

Begin

1. Choose an arbitrary point x from a dataset A;
2. Find all the points density-accessible from x by random/arbitrary values of $R_1$and $MinNodes$ values;
3. **if** x satisfies Equation (3.7), **then**
   Consider it as a *core point* and cluster formation will occur.
4. **if** x satisfies definition 3, **then**

Take it as an *edge point* and other points are not density accessible from x and DC retrieve the next point in A;

else

Noise point is assigned to x;

5. **repeat** the procedure for all of the data points in A.

End

3.2 Proposed DAHDA

After reviewing the preliminaries for the proposed Dynamic Adaptive Hierarchical Data Aggregation (DAHDA) in section 3.1, the workflow of the proposed DAHDA algorithm is shown in Figure 3.2. DAHDA algorithm works in four phases namely *Information Propagation phase, Network Partitioning phase, Cluster Formation phase* and *Weight based Data Aggregation (WDA) phase.*

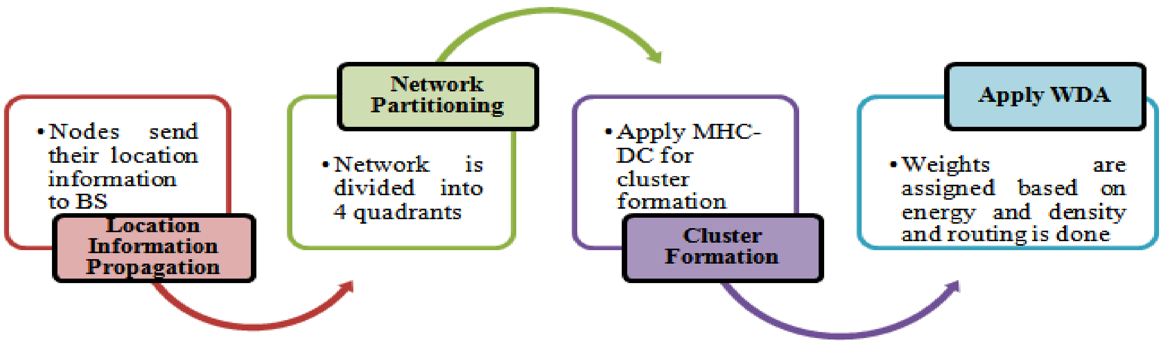


Figure 3.2 Workflow of Proposed DAHDA

The detailed explanation of working of this proposed DAHDA technique is depicted in Figure 3.3 as a flowchart. The flowchart shows the actual tasks performed in the four aforementioned phases of DAHDA. The explanation of these phases is presented in the subsections. In the last step of flowchart, the number of nodes is calculated for each cluster for each round. If the minimum number of nodes (represented by $MinNodes$) required in a cluster are not present (in case any nodes die) then the process of cluster reformation will be initiated again using Meta-Heuristic Computational – Density based Clustering (MHC-DC) algorithm presented in section 3.3. The tasks performed in the four phases are detailed as:

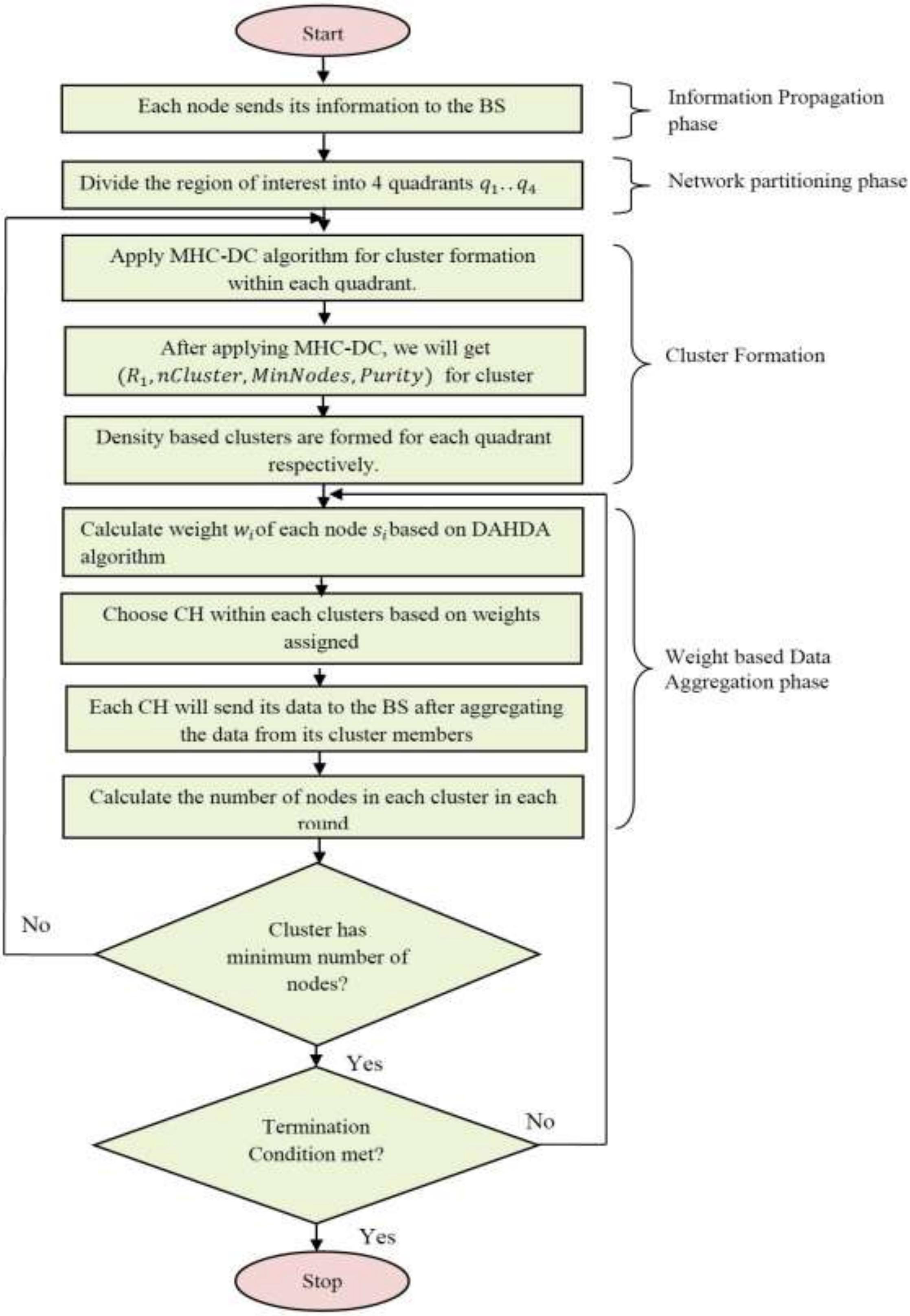


Figure 3.3 Flowchart of DAHDA Algorithm

3.2.1 Information Propagation Phase

In the proposed approach, nodes are deployed in the region of interest. Each node sends its location and residual energy information to the BS. The whole proposed algorithm is executed at power enriched BS.

3.2.2 Network Partitioning Phase

Based on the gathered location information, a network is partitioned into four equal quadrants (q_1, q_2, q_3, q_4) to achieve better load balancing [258]. The network partitioning provides better coverage of the whole region. Figure 3.4 depicts the optimal load distribution approach among nodes in a network in the form of quadrants.

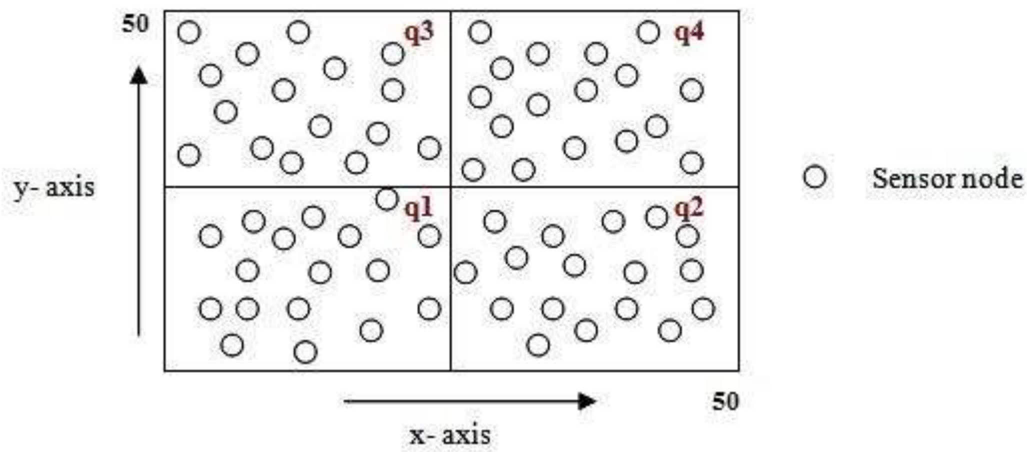


Figure 3.4 Network Partitioning in Quadrants

Nodes are deployed randomly in a $50m \times 50m$ field. The overall network area Q is defined in Equation (3.10) as follows:

$$Q = q_1 + q_2 + q_3 + q_4 \tag{3.10}$$

$$q_{n_q} = Q(x_d, y_d)$$

Where, n_q denotes the number of quadrants and is defined as $n_q = 4$. d denotes magnitude of distance and is defined as $d = 50$. Hence, the overall field is partitioned as given by Equation (3.11):

$$Q = {}_{X_d=0:25}^{Y_d=0:25}lim\, q_{n_q} + {}_{X_d=26:50}^{Y_d=0:25}lim\, q_{n_q} + {}_{X_d=0:25}^{Y_d=26:50}lim\, q_{n_q} + {}_{X_d=26:50}^{Y_d=26:50}lim\, q_{n_q} \tag{3.11}$$

The network partitioning in the form of quadrants leads to an energy efficient utilization of nodes. In LEACH, some of the cluster members are located very distant and clusters are of arbitrary size. Due to this dynamic cluster formation, the energy of distant nodes is exhausted very rapidly and leading to the performance degradation of the network. On the contrary, in the proposed approach a network is divided into quadrants. Hence, clusters formed within these quadrants are more deterministic in nature.

3.2.3 Cluster Formation Phase

Density-based Spatial Clustering is widely used in many areas of research due to its ability to discover clusters of arbitrary shapes, simplicity and robustness against outlier values. However, initially it requires two input parameters for cluster formation: radius of the cluster denoted by R_1 and the minimum number of nodes or data points required inside the cluster, represented by $MinNodes$. The performance of clustering algorithm is significantly influenced by these two parameters. It is very difficult to determine these parameters beforehand. The MHC-DC algorithm is used for determining these two parameters in proposed DAHDA as presented in section 3.3.

3.2.4 Weight based Data Aggregation in Proposed DAHDA

The proposed data aggregation technique i.e. DAHDA is based on centralized data aggregation approach. It consists of two phases namely *setup phase* and *steady state phase*. In the setup phase, DAHDA calculates a weight value, w_i initially and assigns it to each sensor s_i. The definition of k has been modified. k is defined as the percentage of the maximum number of CHs instead of the actual number of CHs as it is defined in the original LEACH. A maximum of k% of alive sensors are selected to be CHs based on the calculated weights, such that the higher are the weights, the better are the chances for nodes to become CHs. Unlike LEACH, DAHDA does not take into consideration the fact whether the node was a CH for previous rounds or not. After all CHs are chosen, clusters are formed such that each sensor is assigned to its nearest CH.

In the proposed algorithm weights, which are assigned to sensors, are calculated as given in Equation (3.12):

$$w_i = \begin{cases} d_i \times e_i \, , \; if \; d_i > d_{thresh} \\ d_i, \qquad\qquad otherwise \end{cases} \tag{3.12}$$

Where d_i is calculated as given by Equation (3.13):

$$d_i = \frac{(number\ of\ sensors\ nodes\ alive\ in\ range\ c + 1)}{n} \tag{3.13}$$

Here d_i represents the density of a sensor node s_i, c denotes the coverage range of sensor s_i, e_i is the residual energy of sensor s_i and d_{thresh} represents the threshold of density to determine the node set in areas of low density. $d_{thresh} = \frac{c_{thresh}}{n}$ varies as the number of alive nodes n varies, where c_{thresh} is the number of sensors that defines the low density areas and it is application specific. In this work, so c_{thresh} is selected as 1. Therefore, d_{thresh} substitutes to the value of $\frac{1}{n}$. As the maximum value of d_i is 1 and the maximum value of e_i is 1, so the term $e_i \times d_i$ cannot exceed the value of 1 and the higher the value of e_i or d_i, the higher is the value of weight w_i.

DAHDA selects a set of nodes within each cluster which will send their data readings to respective CHs. It is based on a predefined percentage $x = 50\%$ in steady state phase. These nodes are selected based on the assigned weights, so that the nodes having lesser weight are considered as better nodes to transmit their data to CHs. The regions which are having low densities and nodes which are distant from their CHs are also considered. Nodes having densities $d_i \leq d_{thresh}$ may be therefore selected more often because of low weights, and thus their lifetime may be shortened. To solve this problem, DAHDA requires that sensors in these areas send their data with some probability p, such that $x\%$ sensors are chosen in each round t; then if any of these chosen sensors has the value of $d_i \leq d_{thresh}$, then it does not send data to the CH unless it satisfies a probability ps, which is an application specific parameter and can be chosen empirically.

3.3 Meta-Heuristic Computational – Density Based Clustering

The performance of clustering algorithm is significantly influenced by radius of the cluster R_1 and the minimum number of nodes or data points required inside the cluster $MinNodes$. In this section, an efficient and effective clustering method i.e. Meta-Heuristic Computational – Density based Clustering (MHC-DC) algorithm is presented to find the optimal values of R_1 and $MinNodes$. This approach utilizes the features of Binary Differential Evolution and Density-Based Spatial Clustering algorithm to quickly estimate the appropriate values for input parameters i.e. R_1 and $MinNodes$ automatically. The blend of Tournament Selection (TS) method and an analytic way of estimating radius of cluster(R_1) is also utilized.

In this section, initially a basic Meta-Heuristic Computational (MHC) approach is presented for estimation of input parameters of DC algorithm. The blend of DC and MHC approach is presented as MHC-DC algorithm in which clusters are formed.

3.3.1 Meta-Heuristic Computational Algorithm

Meta-Heuristic Computational (MHC) algorithm is a meta-heuristic based method for global optimization over continuous spaces [259]. It is based on the Binary Differential Evolution (BDE) structure. The proposed algorithm comprises three basic operations: *mutation, crossover* and *selection*. The MHC algorithm can be summed up as follows:

1. **Initialization:** In this phase, an initial population is generated arbitrarily in s dimensional space as given by Equation(3.14) as follows:

$$y_i(j) = y_j^l + rand(0,1).y_j^u \tag{3.14}$$

where $y_i(j)$ represents the j^{th} variable of the i^{th} individual, and y_j^l represents the lower limit and y_j^u denotes the upper limit of j^{th} variable respectively. $rand(0,1)$ denotes a random value which is uniformly distributed within range [0..1].

2. **Mutation:** Two population vectors y_{p2}, y_{p3} are selected randomly and should be different from each other. The difference between these individuals y_{p2} , y_{p3} is used by scaling factor ŋ. Usually a value is set within a range [0.5 1] to mutate y_{p1} and result is stored in h_i as given by Equation (3.15):

$$h_i = y_{p1} + \eta(y_{p2} - y_{p3}) \tag{3.15}$$

3. **Crossover:** In this operation, y_i and h_i are recombined to generate a new individual $U_i(j)$, as given by Equation (3.16):

$$U_i(j) = \begin{cases} h_i(j); if\ rand(0,1) \leq C_{rate} or\ j = j_{rand} \\ y_i(j); otherwise \end{cases} \tag{3.16}$$

$rand(0,1)$ represents a random number within a range [0 1], j_{rand} is a randomly selected index to make sure that the train vector U_i does not duplicate y_i and Crossover Rate is represented by C_{rate}.

4. **Selection:** Greedy approach is applied to choose the best solution among the parent vector $y_i$and trial individual U_i for the next generation by choosing a cost function f as given by Equation (3.17):

$$y_i = \begin{cases} U_i; if\ f(U_i) \geq f(y_i) \\ y_i; Otherwise \end{cases} \tag{3.17}$$

The detailed algorithm of MHC is discussed in Algorithm 3.2 as follows.

Algorithm 3.2 Pseudo-code of MHC Algorithm

Function MHC
// Pop represents the population and N_p represents the number of individual in population.
Begin
1. Initialize the population Pop by using Equation (3.14);
2. **repeat**
3. **for** i $\leftarrow 1$ to N_p over $Pop(i)$ **do**
    Perform Mutation operation by using Equation (3.15);
    Perform Crossover operation by using Equation (3.16);
    Perform Selection operation by using Equation (3.17) using a cost function;
    end
    until terminating condition;
End

3.3.2 Meta-Heuristic Computational - Density based Clustering Algorithm

A Meta-Heuristic Computational - Density based Clustering (MHC-DC) algorithm has been proposed in which the binary coding scheme is adopted. In this approach, a bit string is used to represent each individual of population. Some definitions of the proposed algorithm are described as follows:

Radius of Cluster: R_1 Parameter

R_1 parameter affects the performance of the DC algorithm. In this approach, the blend of TS method and an analytical way for estimating R_1 is incorporated. TS approach creates more R_1 values with very high diversity until the required blend of $MinNodes$ and R_1 values is chosen based on its specifications. R_1 parameter is calculated by an analytical way as shown in Equation (3.18) [260]:

$$R_1 = \left(\frac{\left(\prod_{i=1}^{max(x_{mat})-min(x_{mat})} i\right) * nbr * \Upsilon(0.5 * n_{var} + 1)}{n_o \; * \; \sqrt{\prod^{n_{var}}}} \right)^{\frac{1}{n_{var}}} \tag{3.18}$$

Here, x_{mat} represents the data matrix of n_{var} number of variables and n_o number of objects. nbr denotes the number of nodes in a locality of a particular node and the factorial function is interpolated by Υ. In each iteration, value of R_1 is calculated and compared with the stored R_1 values obtained from previous iterations of TS method. The initial value of R_1 has high probability by default so that it can be selected among others in the tournament. It is highly suitable for the selection of R_1 and $MinNodes$.

Tournament Selection

It is the most commonly used strategy for selection in evolutionary algorithms [261]. TS selects the R_1 value based on the fitness function (purity) from a population of stored R_1 values. Better the purity value of each R_1, more are the chances of it to be chosen. The achiever of each tournament is chosen for executing the DC algorithm by the generated population ($MinNodes$).

Apply limit over real-coded vectors

The real-coded vectors are generated by standard GA operations but the required data should be in the form of bit strings. A simple method is presented to map these values to 0 or 1 is given by Equation (3.19):

$$Binary\_coded = min(max(Real\_coded, 0), 1) \quad (3.19)$$

Uniform Distributions of Scaling Factor

A random number of size $VarN$ bit strings is created from the continuous uniform distributions for scaling factor (η) in mutation operator. The lower and upper bound are represented by β_{min} and β_{max} respectively. This is given by Equation (3.20):

$$\eta = unifrnd(\beta_{min}, \beta_{max}, VarN) \quad (3.20)$$

Performance Metric

Clustering is an unsupervised classification and is generally harder to measure than a supervised approach. The performance of MHC-DC algorithm is measured in terms of Purity criterion to evaluate its effectiveness and accuracy. It evaluates the wellness of clusters generated and is defined as fitness function in the proposed algorithm. This metric calculates the frequency of the most commonly used class/category in each cluster as given by Equation (3.21):

$$Purity = \frac{1}{n_s} \sum_{cl=1}^{l} max_{1 \le j \le ct}\, n_{s,cl}^{j} \quad (3.21)$$

Where, n_s denotes the total number of samples; ct is the number of categories, $n_{s,cl}^{j}$ is the number of samples in cluster cl that belongs to the original class j ($1 \le j \le ct$). A large purity (approximately 100%) is required for a good clustering. After this phase, the required number of clusters denoted by $nCluster$ is calculated. In the next phase, Weight based Data Aggregation along with routing mechanism within each cluster is discussed in section 3.2.4.

The evolution process will terminate when the maximum number of generations is generated or when the maximum fitness value is achieved. The elaborated approach of MHC-DC algorithm is presented in the Algorithm 3.3. The entire process of MHC-DC algorithm is represented in Figure 3.5.

Algorithm 3.3 MHC-DC Algorithm for Cluster Formation

Function MHC-DC (Input Matrix: data set)
// Declare parameters of Meta-Heuristic Computational (MHC) algorithm;
//$MaxIter$ represents the maximum iterations of MHC algorithm;
//N_p represents the number of individuals in population;
//$VarN$ represents the number of decision variables;
//$Best\_Sol$ represents the best solution in each loop based on the maximum fitness value using Equation (3.21). Initially, $Best\_Sol$ =null;
// Structure of Population is defined
// $Pop(i).Position$ represents position of each individual point in the search space, $i = 1,2..N_p$;
// $Pop(i).Cost$ represents fitness/purity value of each individual;
// $Pop(i).Sol$ represents solutions of each individual stored which includes $MinNodes, R_1$, $Purity$ and $nCluster$;

Begin

1. $Pop \rightarrow [p_1, p_2 \ldots p_n]$, e.g., if $MinNodes_i = 28$ and $VarN = 8$ then $Pop_i = [00011100]$; //An array is defined for storing global information.
2. **for** $i = 1 \; to \; N_p$ **do**
   $Pop(i).Position = rand_i([0..1], VarN)$;
   //Generate the random initial position value (called MinNodes) based on the number of $VarN$ by using binary coding (0 or 1);
   $[Pop(i).Cost \; Pop(i).Sol] = Fitness\ Function(Pop(i).Position)$;
   end

//MHC Main Loop

3. **for** $i \leftarrow 1$ **to** $MaxIter$ **do**
   Call Function MHC as shown in Algorithm 3.2
   //Update Best Solution of each $Pop(i)$
   if $Pop(i).Cost > Best\_Sol.Cost$ **then**
   $Best\_Sol = Pop(i)$;

else if $(Pop(i).Cost\ ==\ Best\_Sol.Cost)$ && $(Pop(i).Sol.nCluster <$ $Best\_Sol.nCluster)$ **then**

$Best\_Sol\ = Pop(i)$;

end

if $Best\_Sol.Cost(i) == 1$ **then**

Exit; /* fitness value==100%*/

end

end

end

Fitness Function($MinNode$)

Begin

1. Compute an analytical formula of calculating locality radius (R_1) for DC by using Equation (3.18);
2. Execute TS approach to choose R_1 value by high probability (fitness value) among stored values of R_1;
3. Execute $DC(MinNodes, R_1)$ standalone Algorithm 3.1;
4. Execute fitness function over DC results by Equation (3.21);
5. **return** $MinNodes, R_1, nCluster$ and $Purity$;

End

3.4 Extended DAHDA

This variant is directed to add more to the technical aspects of the algorithm to achieve longer network lifetime. The proposed DAHDA is further improved to optimize the total energy consumption in the network and is termed as *Extended DAHDA (EDAHDA)*. It adds a dynamic nature to one important parameter of DAHDA.

A new metric, namely, CH-density (d_{CH_i}), is defined to represent the density of clusters as given by Equation (3.22):

$$d_{CH_i} = \frac{(number\ of\ \ sensor\ nodes\ \ alive\ in\ cluster\ i)}{(total\ number\ of\ alive\ sensors)} \tag{3.22}$$

This metric is used to dynamically adjust x, the number of sending nodes for each cluster, such that clusters with higher d_{CH} needs less percentage of nodes than clusters with lower d_{CH}. This is justified as follows: for clusters with high d_{CH}, the percentage of sending sensors could be safely decreased and yet it results in a reasonable number of sending sensors, while for clusters with low d_{CH}, the percentage is required to be increased in order to reach a fair and appropriate number of sending nodes.

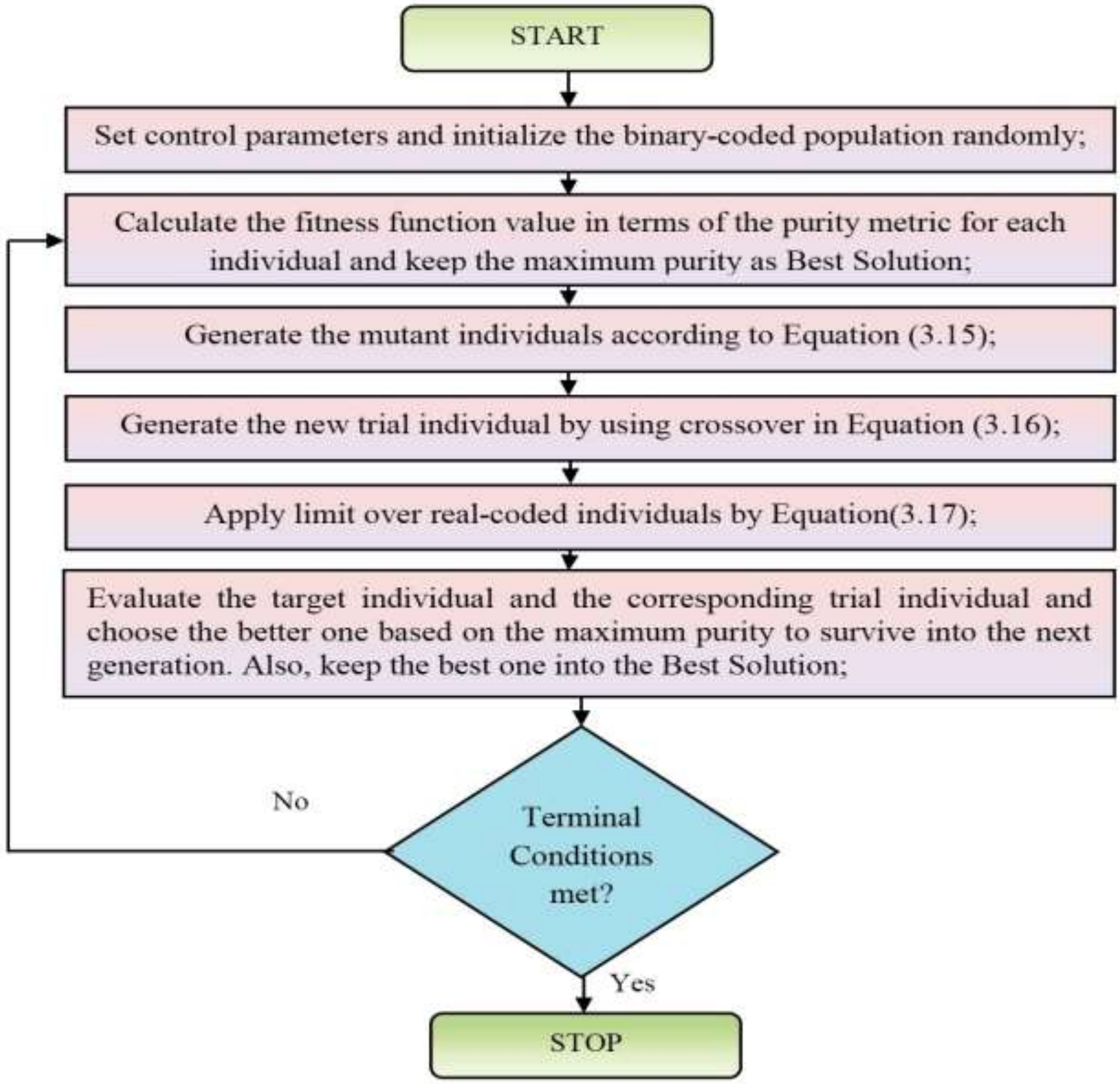


Figure 3.5 Flowchart of Meta-Heuristic Computational - Density based Clustering

The percentage of nodes used to transmit data in DAHDA is x, therefore, redefined as given by Equation (3.23):

$$x_{dynamic} = \begin{cases} 0.1,\ if\ (0.01)^{d_{CH}} < 0.1 \\ 0.5,\ if\ (0.01)^{d_{CH}} > 0.5 \\ (0.01)^{d_{CH}},\quad otherwise \end{cases} \tag{3.23}$$

Here $x_{dynamic}$ varies from 0.1 to 0.5 (10% to 50%). When $x_{dynamic} = 0.5$, then the algorithm acts as the original DAHDA that is presented in Section 3.2. Since the main target is to reduce the number of sending sensors whenever possible rather than increasing their density. The maximum limit is set in the above Equation (3.23). The minimum limit is application specific and it is chosen to be 10% to ensure that the number of selected sensors for big clusters at any time is not very low and therefore data is not lost.

As it is shown in the Equation (3.23), the value of d_{CH} is inversely proportional to $x_{dynamic}$ which means the more nodes a cluster has, the lesser is the percentage of its transmitting nodes to transmit the data. Solving the Equation (3.23) for d_{CH}, the minimum $x_{dynamic} = 0.1$ is achieved for those clusters which are having $d_{CH} \geq 0.5$, which means the clusters with ≥ 50% alive nodes are formed in the network. The maximum value of $x_{dynamic} = 0.5$ is achieved for those clusters having $d_{CH} \leq 0.15$ means clusters have less than or equal to 15% of alive nodes.

3.5 Modified EDAHDA

This variant considers the data accuracy while applying EDAHDA on non-uniform WSNs, to show that the algorithm does not affect the general accuracy of the collected data. This approach has been termed as *Modified EDAHDA.* The algorithm includes the adaptivity feature that can handle any sudden bursts in the data values making sure that the data in the areas of interest is captured continuously.

The proposed EDAHDA approach has been further modified in order to handle sudden bursts in the underlying values of the data. The changes in data may appear according to sudden events in the sensed field. Handling such sudden events is based on the actual values of the sensed data. The algorithm should, therefore, keep track of the values of the data for each sensor independently.

For each sensor s_i, at a specific round t, the change in values of the sensed data affects the weight w_i of the sensor in the network. The more significant the change is, the more is the weight reduced to have a better chance to be selected by the algorithm to send its data to CH for this specific round. The number of transmitting sensors in each cluster is not affected by this improvement. This phase only highlights the area of interest so that the algorithm is more attracted to choose sensors in the area of interest for transmitting data to their CHs in order to reflect the changes.

The algorithm keeps track of the changes in values of sensor s_i, such that for each round t, the accumulated average $V_i avg(t)$ is calculated as the average of values seen so far. The difference between the current recorded value and the average is defined as $\Delta V(t) = V_i(t) - V_i avg(t)$. Suppose that sensor s_i is located in cluster c_k at round t. So, it is defined as $\Delta V_{c_k max}(t) = max(V_i(t)|_{c_k})$. The ratio of the difference $\Delta V_i(t)$ to the maximum recorded difference in cluster c_k at round $t(\Delta V_{c_k max}(t){:}\, V(t)) = \Delta V_i(t) / \Delta V_{c_k max}(t)$ affects the weight. The higher is the ratio $V_i(t)$, the lower is the weight. The weight formula is modified accordingly as given by Equation (3.24):

$$w_i = \begin{cases} e_i \times d_i^{1+v_i}, & if\ d_i > d_{thresh} \\ d_i^{1+v_i}, & otherwise \end{cases} \tag{3.24}$$

The modification is a plausible solution to reflect the required constraints as follows:

(i) When there is no change in the readings of sensor s_i, the density is not affected as V_i is zero and therefore the weight remains the same.

(ii) As changes occur, the value of $V_i(t)$ starts to affect the weight, such that a larger value of $V(t)_i$ means a more serious change. Therefore, a higher is the value of the power and less is the overall value of the weight as the density is a fraction; $0 < d_i \leq 1$.

(iii) The weight reduction is limited, such that the maximum limit of reducing the weight is d_i^2 , which means to reduce value of d_i to its half. Comparing the accumulated average for a sensor to its recorded values helps to give an indication that this specific sensor has detected some event only if the current recorded value is far from the average value. If so, the algorithm is attracted to choose the sensor or group of sensors which have detected an interesting event and remains interested in choosing these nodes again until the event is over and the values are getting back to regular state.

3.6 Experimental Settings

The existing LEACH and the proposed DAHDA algorithm are simulated using NS2. While simulating the various scenarios, it is assumed that all nodes can directly communicate each other within a communication range.

3.6.1 Testing Parameters

The proposed DAHDA algorithm is executed on non-uniform WSNs only. The uniform networks are not considered since uniform networks are taken as a special case of non-uniform networks. While executing the proposed algorithm it is evident that the performance of evolving networks is similar to the performance of non-uniform networks, which shows its effectiveness to be utilized for evolving networks. The various performance parameters which are considered for evaluation of the proposed algorithms are discussed as follows:

(i) **Node distribution through network lifetime:** It shows the variation in the distribution of nodes in the region of interest by monitoring the network structure through the full execution for static regions only. The evolving networks are not considered in this measure.

(ii) **Lifetime of first node, last node and average sensor lifetime**: It includes the monitoring of the round at which the energy of the first node and the last node gets exhausted in the network. The average sensor lifetime for nodes is calculated by taking the sum of the lifetime divided by the total number of nodes.

(iii) **Number of alive nodes:** This parameter represents the number of nodes which are still alive at the end of each iteration.

(iv) **Residual energy:** This parameter represents the residual energy for the network at the end of each iteration.

(v) **Data accuracy:** This parameter measures the average of the data recorded by DAHDA and compares it with the actual average of data to ensure that the sensor readings which are aggregated at the BS represent the current situation accurately.

(vi) **Sudden bursts detection:** In this, behaviour of the algorithm is monitored to make sure that the variations in values are captured. The results are measured in terms of all performance parameters except for the first parameter i.e. node distribution.

3.6.2 Simulation Settings

The algorithms are tested by taking number of alive nodes $n = 100$ initially where each node has a radius of $5m$. A network of size 50×50 meters is configured, with a BS located at point [25, 50]. The initial energy $E_{initial}$ of the sensors is set to $1J$. The sensors' energy consumption for transmitting and receiving data in this work is based on the radio model as shown in Table 3.1. Apart from these parameters, energy dissipation for data aggregation $E_{DA}$is taken as 5nJ/bit.

Table 3.1 Simulation Parameters for Experimental Setup

Task/Operation	Energy dissipation
Receiver/ Transmitter circuit dissipation E_{elec}	50 nJ/bit
E_{fs}	10pJ/bit/m^2
E_{mp}	0.0013pJ/bit/m^4
Message Size l	2000 bits

The probability of selecting the maximum number of CHs k. is defined as 5%, based on results obtained for LEACH. However, k is increased to 15% for the real dataset to reflect better cluster functionality. Since using 5% would result in one cluster even at the early rounds of the iterations as the real data set which is used has only 18 sensors, so 5% of 18 results in only one cluster. Finally, the probability of transmitting the data in low density areas is set as p = 25%. The values of k (for the real data set) and ps are chosen empirically, while the value of k in the case of synthetic data is chosen based on the LEACH results.

3.6.3 Network Structure

The network structures for which the simulation tests are performed are shown in Figure 3.6, Figure 3.7 and Figure 3.8 for non-uniform, evolving and real data WSN, respectively.

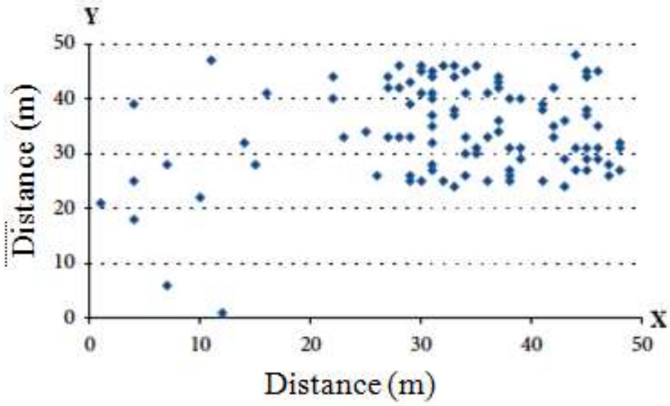
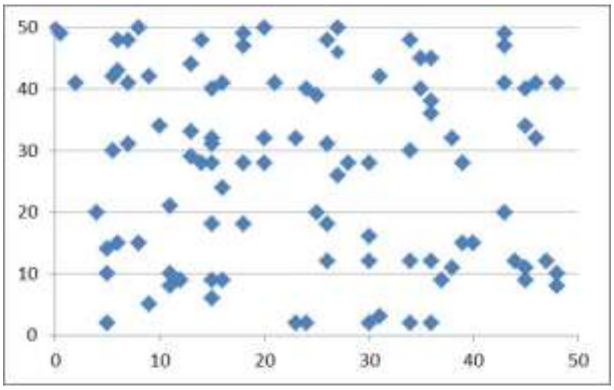
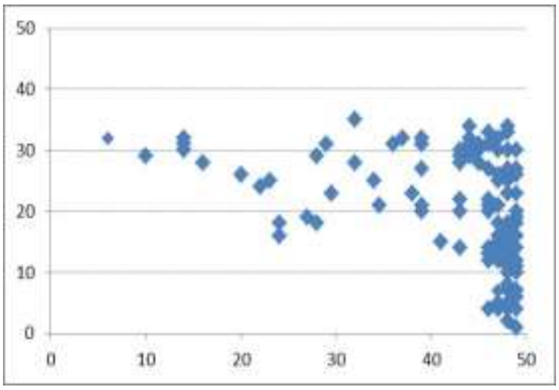


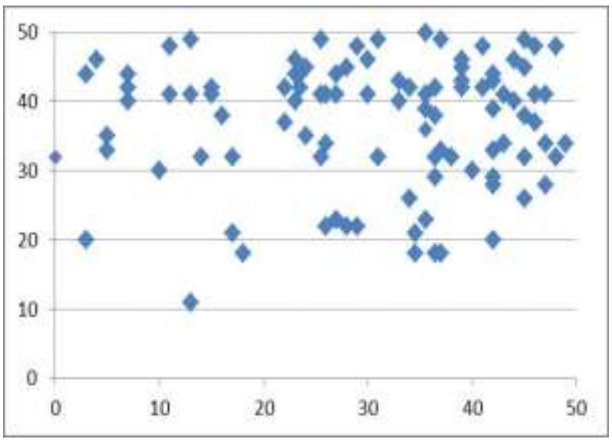
Figure 3.6 WSN with Non-uniform Sensor Distribution



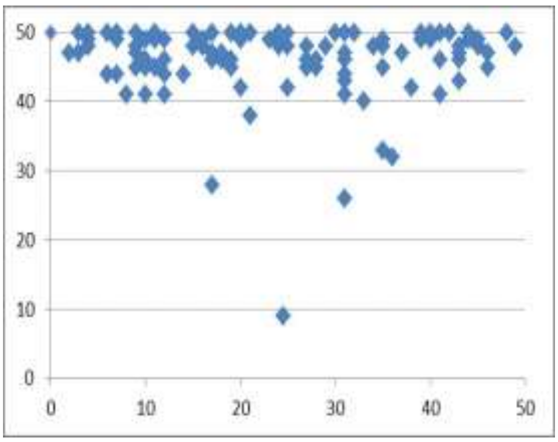
(a) Initial Network



(b) Round 2000



(c) Round 5000



(d) Round 6000

Figure 3.7 Evolving WSNs

The non-uniform network is used for synthetic dataset to show the adaptive nature of the algorithm using Modified EDAHDA as shown in Figure 3.6, while the network which is shown in Figure 3.8 is used for the real dataset accuracy test as well as for event detection.

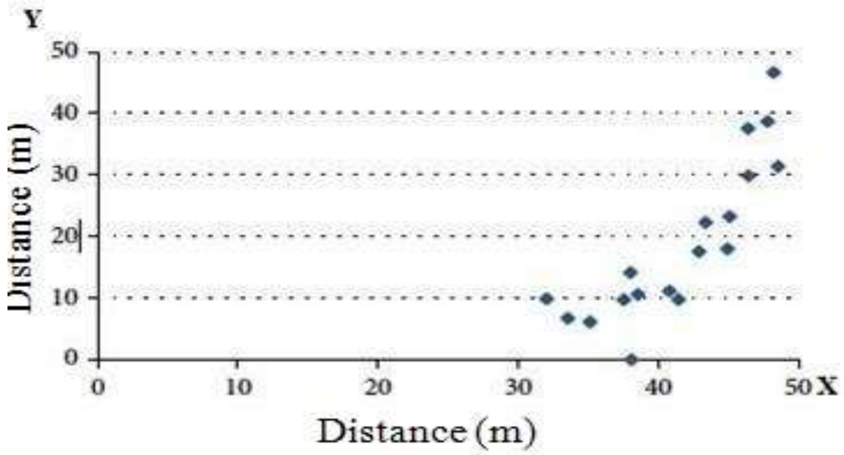


Figure 3.8 WSN with Sensors Positioned According to the Sensor Scope Deployment

3.6.4 Data Sets

The synthetic and real data sets are used to test the data accuracy of the EDAHDA algorithm and to evaluate the performance of the Modified EDAHDA algorithm with adaptivity feature.

(i) **Real Data:** To test the data accuracy of EDAHDA, data set is taken from a real-world deployment SensorScope (Grand-St-Bernard Deployment)[262]. The data set consists of readings which have been collected over 43 days by 18 weather stations with sensors. The ambient temperature values are used which are recorded by the sensors to test the data accuracy. The files which contain the sensor readings have been checked to eliminate any erroneous values which might exist. Also, any readings which are not synchronized due to certain sensors going off and back on have been removed before using the files. The maximum number of rounds in the simulation is taken as 9300. The GPS positions of the sensors in the SensorScope deployment have been used to recreate the actual WSN while testing EDAHDA. In all of the simulations, the actual sensor positions are mapped to a network size of $50 \times 50\, m$. Figure 3.8 shows the location of the different sensors on the $50 \times 50\, m$ grid for real data testing. Although the Grand-St-Bernard deployment is relatively small since it contains only 18 sensors, still it is used as data set as it is the most suitable data for the testing requirement since most of the recorded sensor readings are already synchronized and the GPS positions of the WSN are available. Therefore many pre-processing tasks are not needed which might result in major changes in the real sense of the data set.

ii) Synthetic Data: A synthetic data set has been used on a large WSN in order to better test the performance of the Modified EDAHDA with the adaptivity feature. The size of the WSN for the synthetic data set is taken as 100 sensors. To create the synthetic data for each sensor, the recorded ambient temperature values are used from one of the sensors in the SensorScope deployment and noise is added to the values. The noise is applied by taking each temperature value and shifting it randomly by fixed amount in the range [−2, 2]. This process is repeated to create the entire synthetic data set for WSN with 100 sensors. The total number of values in the synthetic data set is taken as 9300.The synthetic data set has been applied on the non-uniform WSN with the sensors positioned as shown in Figure 3.6.

3.7 Results and Discussions

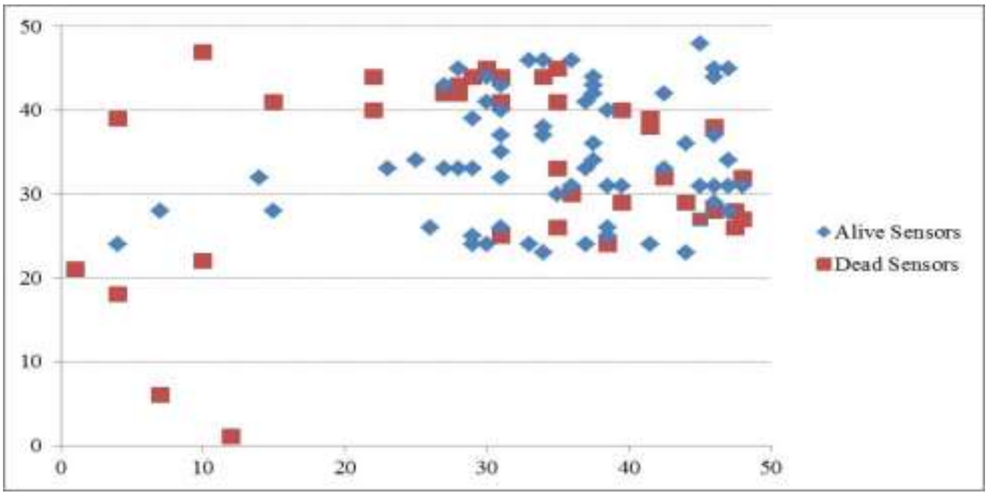
DAHDA has been simulated and its performance has been analyzed in terms of node distribution through network lifetime, lifetime of first node, last node and average node lifetime, number of alive sensors, residual energy, capturing data in areas with events of interests and data accuracy.

Test Case 1: Node Distribution during Network Lifetime

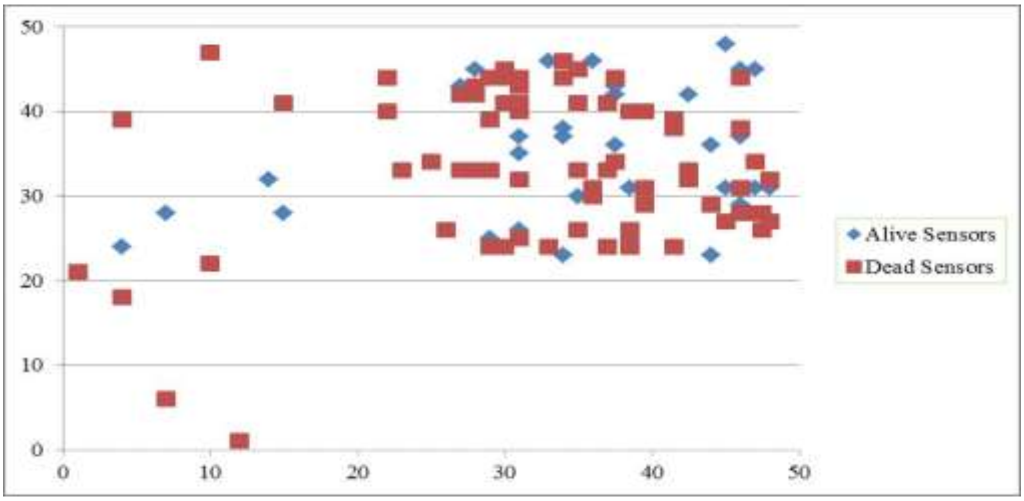
The initial node distribution in a non-uniform network is depicted in Figure 3.6. Figure 3.9 shows a structure while executing the EDAHDA. Figures 3.9(a) and 3.9(b) depicts the node distribution at rounds 8330 and 8355, respectively. The proposed algorithm can evidently successfully handle non-uniform networks as depicted in Figure 3.9. In round 8330, with 99.5% of the network lifetime, the network still operates with 65% of its nodes. Few of these nodes represent the areas with lesser densities. This shows that EDAHDA algorithm enhances the lifetime of the nodes at less dense regions so that data readings of these areas are not missed.

Test Case 2: Lifetime of First Node, Last Node and Average Lifetime of node

The lifetime of first node, lifetime of last node and average lifetime of nodes are compared when LEACH and EDAHDA are performed on the non-uniform networks as depicted in Figure 3.10. As it is clearly shown in the figure, EDAHDA dramatically extends the lifetime of first node, last node and the average lifetime when compared to LEACH. The most affected variable is the *lifetime of first node*, which is thrice more than its average value as it is shown Figure 3.10.



(a) Round 8330



(b) Round 8355

Figure 3.9 Node Distribution in Non-uniform WSN for EDAHDA

This shows that the proposed algorithm tends to save the battery power in the different areas of the network as much as possible to make sure that the network captures all data equally.

Even when the first node dies, the algorithm makes an effort to save the energy of network and operates with less number of nodes which are deployed all over the region of non-uniform network. As shown in Figure 3.10, the recorded results for lifetime of last node and average lifetime of the networks are very close to each other.

The non-uniform WSN has higher network lifetime as compared to the evolving WSN as it can be seen from the *lifetime of last node* values. Although the network deployment is different for evolving and non-uniform networks, both are presented in one figure 3.10 to show that the algorithm performs similarly on both evolving and non-uniform networks, as the non-uniform can be considered as variant of the evolving networks.

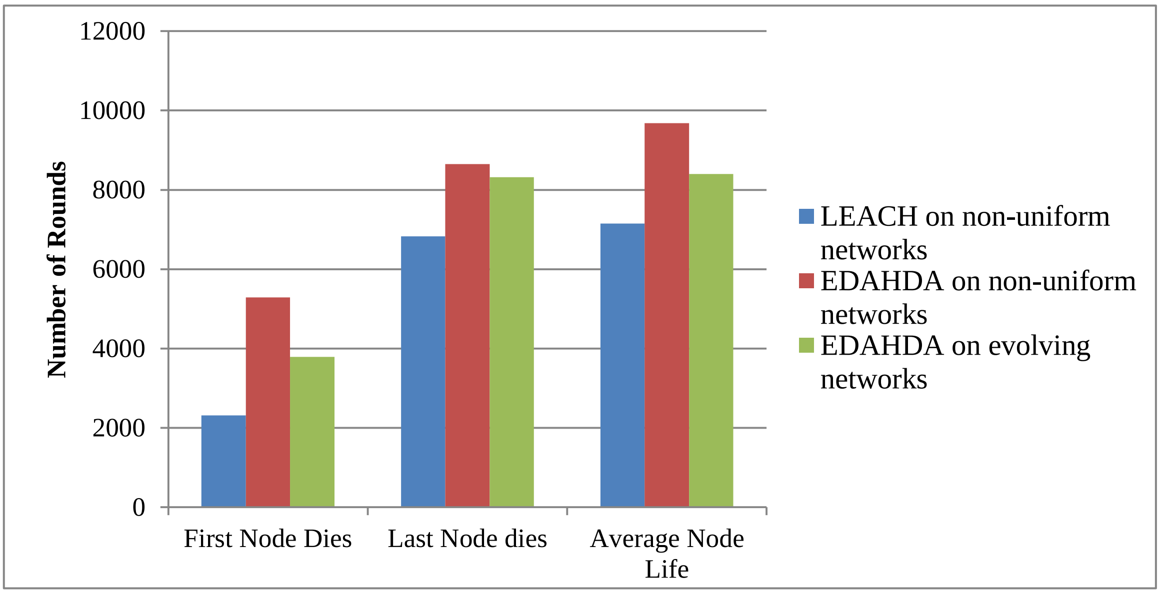


Figure 3.10 Lifetime of Last Node, First Node and Average Lifetime of Node for LEACH and EDAHDA

Test Case 3: Number of Alive Sensors

Figure 3.11 shows the number of alive nodes versus number of rounds in LEACH and EDAHDA for evolving and non-uniform networks. As it is depicted in the figure 3.11, the number of rounds of LEACH on non-uniform networks is around 5200 rounds with about the last 1000 rounds performing with less than 5 sensors. Whereas for EDAHDA, the number of rounds is increased by more than 3000 rounds resulting in approximately 8500 rounds as compared to the existing technique. 90% of the sensors are still performing till the last 200 rounds where the number of alive sensors starts to drop. Similarly, EDAHDA performs well for evolving networks. It saves the power of the sensors to later rounds. Even though the number of remaining alive sensors drops gradually at later rounds, the overall network lifetime is, however, increased.

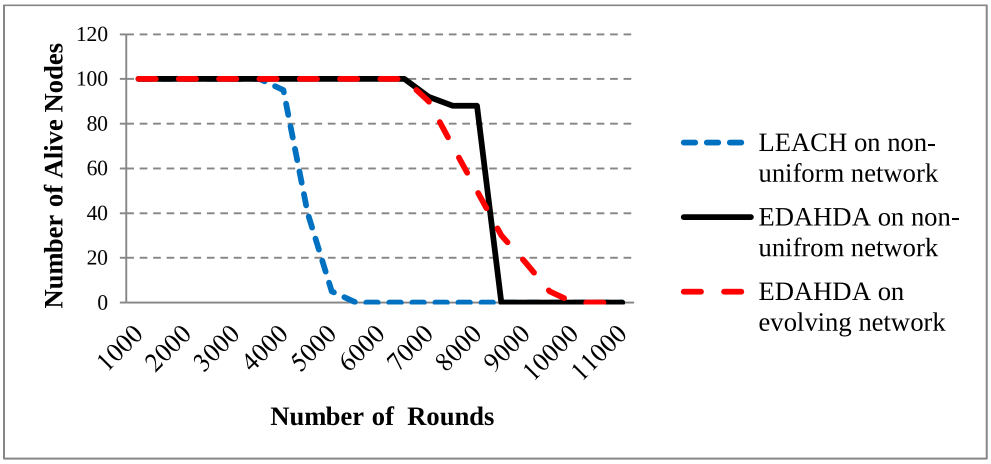


Figure 3.11 Number of Alive Sensors Versus Rounds for LEACH and EDAHDA

Test Case 4: Residual Energy

Figure 3.12 presents the total residual energy versus number of rounds in case of non-uniform networks for LEACH and EDAHDA. It also presents the total residual energy for EDAHDA when applied over evolving networks. The figure clearly shows that huge amount of energy is saved when EDAHDA is applied to evolving as well as non-uniform networks. These above illustrated results in terms of number of alive sensors are also supported by the results as shown in Figure 3.11. Since EDAHDA saves sensors' lives to later rounds of the network as it works with higher residual energy than LEACH.

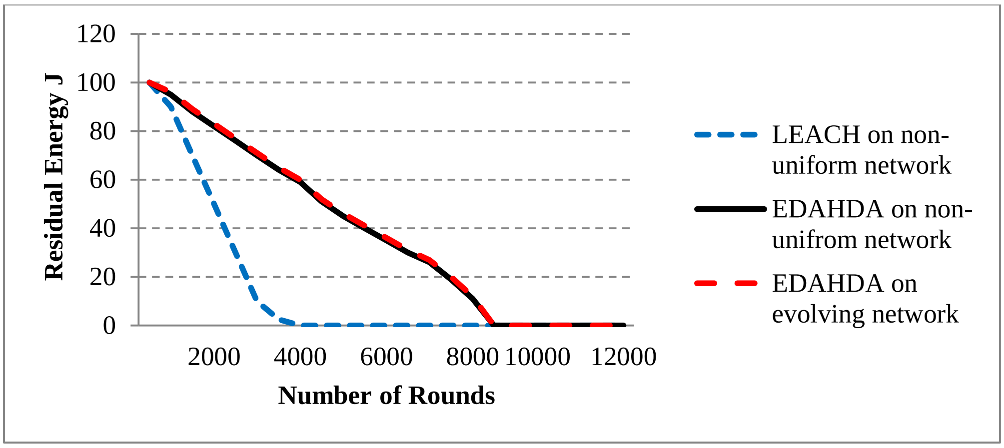


Figure 3.12 Residual Energy Versus Rounds for LEACH and EDAHDA

Test Case 5: Data Accuracy

The data accuracy of EDAHDA is evaluated using the ambient temperature values from the real dataset. The accuracy of the data collected by the EDAHDA is shown in Figure 3.13. The figure compares the average ambient temperature that is recorded by EDAHDA to the actual average temperature. The average temperature values of EDAHDA are calculated by finding the average values of the temperature which are collected from the sensors, which are selected by the algorithm in every round. The actual average temperature is basically the average value of all the temperature readings which are recorded by all the sensors in the field. The figure shows that EDAHDA maintains higher data accuracy. It records an average change of only 7% from the actual average.

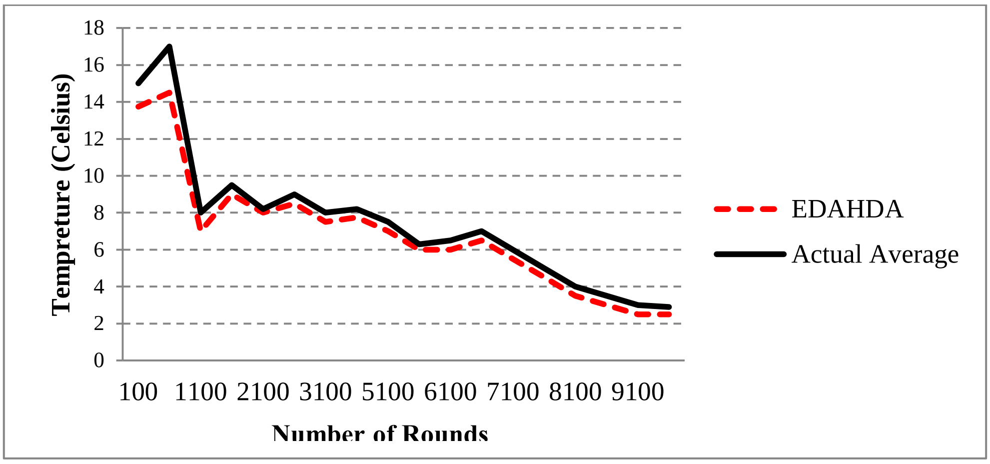


Figure 3.13 Data Accuracy of EDAHDA

Test Case 6: Capturing Data in Areas with Events of Interests

The performance of the Modified EDAHDA with adaptivity feature is evaluated by capturing sudden bursts or events in the underlying values of the data. The performance is evaluated by introducing sudden events into the synthetic and real dataset files and then running the new data on EDAHDA (original and modified version with adaptivity feature) to highlight the difference. In addition, the events take place over a specific area in the sensed field, means that only part of the sensor network is able to detect the event. For the simulation, an event is defined using three parameters: *change in the underlying temperature, event duration in simulation rounds* and *coordinates of the rectangular area* in which the event lies. The events which are introduced in the synthetic dataset are shown in Table 3.2. Figure 3.14 shows the area of the events on the sensed field. From the figure, there are ten sensors in the event area for events 1 and 2, while for events 3 and 4 the number of sensors is seven. One factor that affects the performance of EDAHDA in capturing sudden events is the number of sensors in the event area (event-area sensors) that eventually sends the event-data to their CHs. Another factor is capturing the event throughout its whole duration. The event is represented more accurately if a high number of event-area sensors send the event data and if the event data is recorded for the whole event. To evaluate the performance of EDAHDA in capturing events of interest, a new metric i.e. $Detection\_Ratio$ is defined in Equation (3.25):

$$Detection\_ratio = \frac{actual\ number\ of\ events\ \ sensors\ selected}{possible\ number\ of\ events\ \ sensors} \tag{3.25}$$

Table 3.2 List of Events

Event	ΔTemperature	Rounds	Event Coordinates
1	+25.0	2000-2020	(26,30),(34,22)
2	+15.0	4000-4020	(26,22),(34,22)
3	+20.0	5000-5020	(46,34),(50,24)
4	+30.0	7000-7020	(46,34),(50,24)

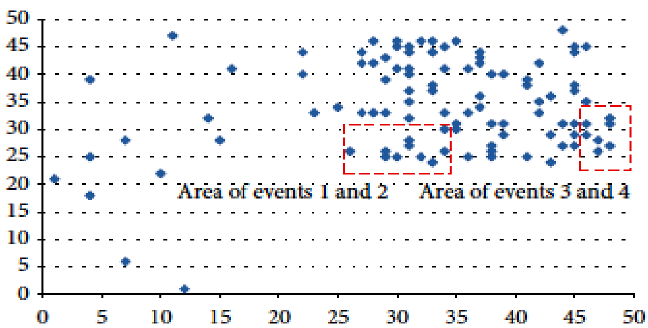


Figure 3.14 Position of the Events 1, 2, 3, and 4 in the Sensed Field

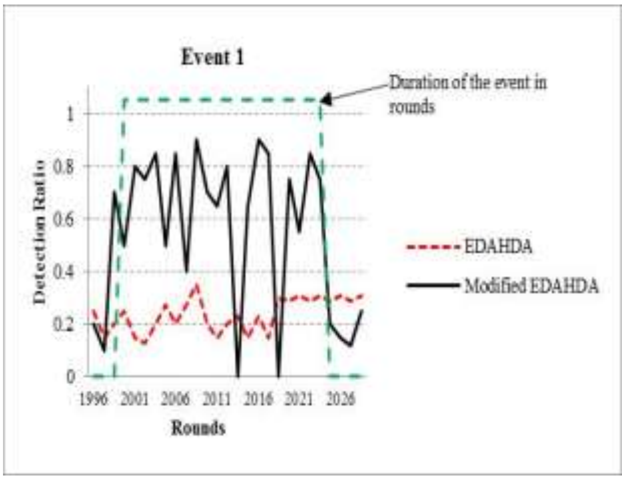
The possible number of event sensors is the maximum number of event sensors nodes which can be selected in a given round. This value depends on the number of sensors that will send data to the CHs. EDAHDA will choose the number of sending sensors for each cluster in each round.

To illustrate, consider the following case in which the current round has two clusters (C_1 and C_2). Cluster C_2 has three event sensors along with other sensors, while cluster C_1 has no event sensors. It is assumed that each cluster will have one sensor which will send the sensed data to the CH. In this case, the possible number of event sensors is one because at most one event sensor can be selected from cluster C_2. The actual number of event sensors selected can be either zero or one depending on whether or not any event sensor will eventually be chosen from cluster C_2. The possible number of event-sensors cannot be zero because in each round at least one sensor from each cluster will send the data. The $Detection\_Ratio$ value becomes 1, if all the possible event sensors are actually selected by EDAHDA to send the event data to their respective CHs.

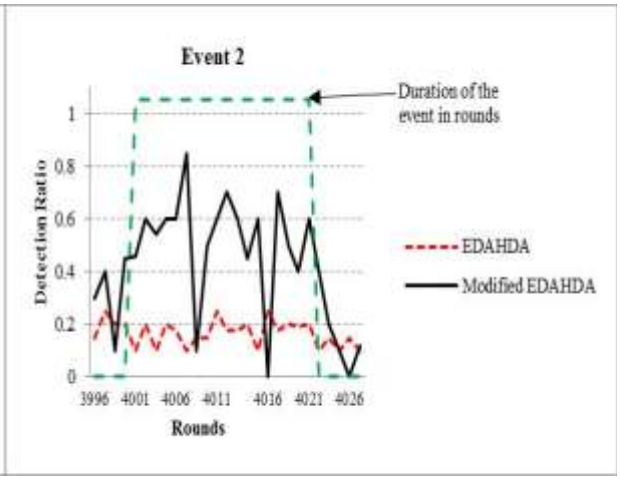
Figures 3.15 show the $Detection\_Ratio$ for the original and Modified EDAHDA algorithms under the events which are specified in Table 3.2. From the figures, it is evident that the detection ratio for the Modified EDAHDA is higher than that of the original EDAHDA for the duration of the different events with the exception of few rounds. Outside the event window, the detection ratio value for the Modified EDAHDA becomes closer to the original EDAHDA. There are a few cases, rounds 2013 and 2017 in Figure 3.15(a) and round 4009 in Figure 3.15(b), in which the $Detection\_Ratio$ of the Modified DAHDA is equal to or lower than that of the original DAHDA.

These six rounds might reflect some of the cases where far away sensors (sensors with low density that is less that d_{thresh}) are having low weights (lower than the sensors in the area of interest) and therefore attract the algorithm to choose these nodes for sending data to their CHs. Again, it is emphasized that a higher detection ratio during events means that the events are captured more accurately. Thus, it is clear that the Modified EDAHDA performs better than the original EDAHDA in detecting sudden events by biasing its sensor selections towards the sensors in the events area.

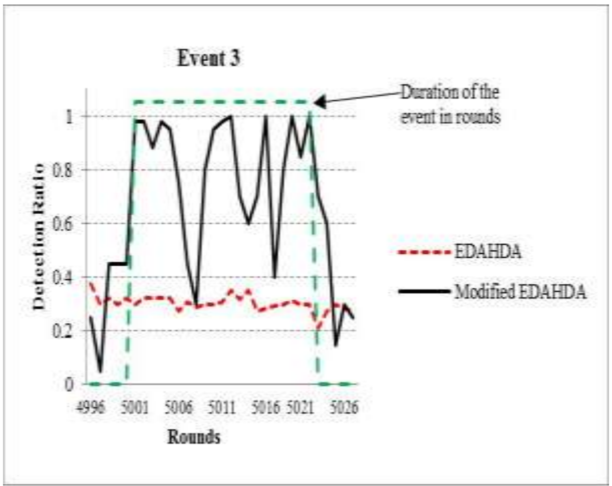
The modified EDAHDA algorithm is also simulated on the real dataset with a single event. Figure 3.15(e) shows the detection ratio for both versions of EDAHDA during the event. In Modified EDAHDA, $Detection\_Ratio$ is constantly one throughout the duration of the event resulting in better event detection.



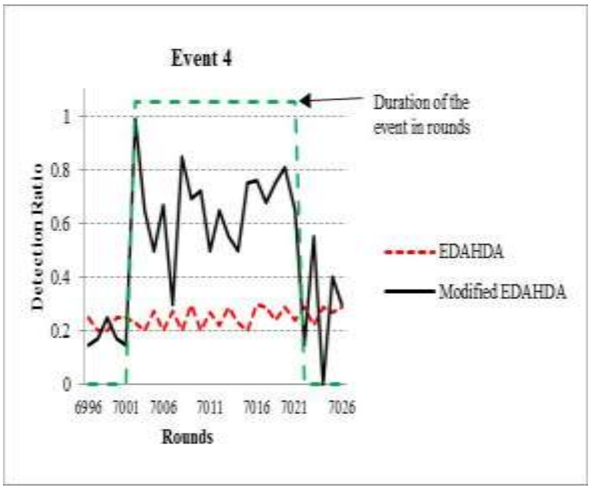
(a) Event 1



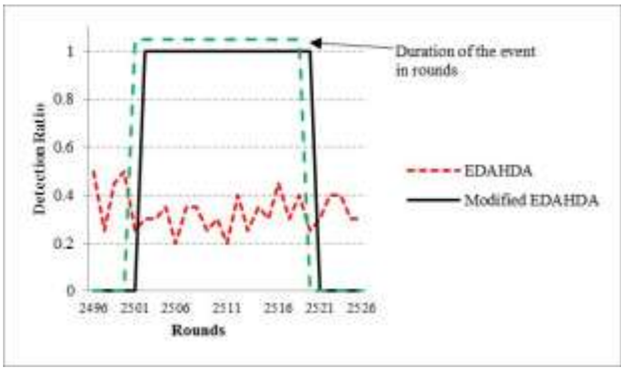
(b)) Event 2



(c) Event 3



(d) Event 4



(e) Detection ratio for Real Data Set

Figure 3.15 Sudden Burst Detection Ratio of EDAHDA and Modified EDAHDA with Adaptivity for the Synthetic Dataset

Test Case 7: Packet Delivery Ratio Vs. Number of Iterations

The value of Packet Delivery Ratio (PDR) with respect to *number of iterations* has been calculated for Modified EDAHDA, EDAHDA, DAHDA, VIBE and EHEED. As shown in Figure 3.16, with the increasing number of iterations, the receiving rate of data packets is decreasing due to decreasing residual energy. Initially, in Modified EDAHDA technique maximum number of packets are received with Packet Delivery Ratio (PDR) of 0.931 (93.1%) in 20 iterations; but existing techniques EHEED, EDAHDA and EHEED are receiving almost same number of data packets with PDR value of 0.88 approximately. The minimum number of data packets is received in Modified EDAHDA at 100^{th} iteration with value of PDR as 0.68. Average number data packets received in Modified EDAHDA are 7.18%, 20.35%, 13.14% and 6.06% more than EHEED, DAHDA, EDAHDA and VIBE respectively.

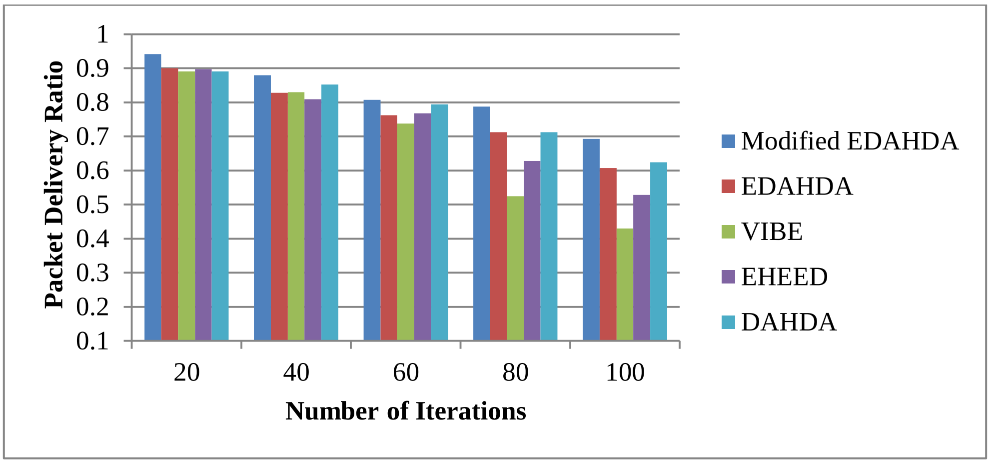


Figure 3.16 Packet Delivery Ratio Vs Number of Iterations

Test Case 8: Throughput

It is a ratio of total amount of data transferred successfully to the total amount of time required to transfer data. Throughput is calculated using Equation (3.26).

$$Throughput = \frac{Total\ amount\ of\ data\ transferred\ successfully}{Total\ amount\ of\ time\ required\ to\ transfer\ data} \tag{3.26}$$

The value of throughput has been calculated for Modified EDAHDA, VIBE, EDAHDA, EHEED and DAHDA with different number of nodes (20 to 100). Figure 3.17 shows the comparison of throughput of Modified EDAHDA, VIBE, EDAHDA, EHEED and DAHDA. It is clearly shown that Modified EDAHDA performs better than the selected existing techniques. The maximum value of throughput at 20 nodes i.e. Modified EDAHDA has 6.41%, 9.17%, 17.66% and 18.22% more throughput as compared to EDAHDA, DAHDA, VIBE and EHEED respectively. The rationale behind this performance is the higher PDR and energy efficiency of proposed Modified EDAHDA as compared to the existing techniques which already have been analysed via results depicted in Figure 3.12 and Figure 3.16.

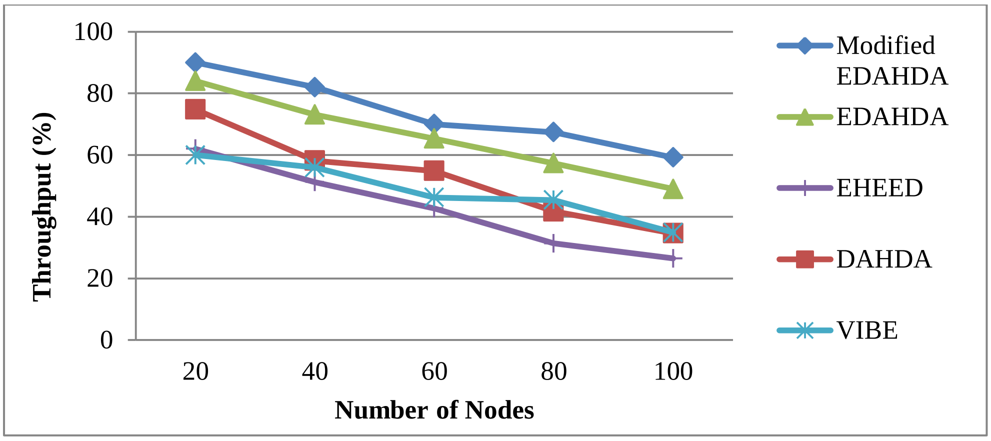


Figure 3.17 Throughput Vs. Number of Nodes

3.7.1 Comparison with Selected Extensions of LEACH Algorithm

Unlike the above mentioned algorithms, the proposed algorithm, DAHDA, is mainly based on extended LEACH in the way of data aggregation. It is, however, similar to the group of algorithms which extend LEACH in the way of selecting the CH, since the weights which are introduced by DAHDA affects the CHs selection as well. A comparison between proposed algorithm and existing variants of LEACH algorithm is presented in Table 3.3.

As it can be seen from Table 3.3, the two best algorithms in extending LEACH are EHEED and proposed EDAHDA. EHEED extends the network lifetime by 87% approximately and provides 60% better energy consumption. The proposed DAHDA is tested on non-uniform and evolving networks. It also performs with 120% more in average node life than LEACH and 155% more in first node dies, which shows that the nodes are kept to later stages.

Table 3.3 Comparison between DAHDA and Other Selected LEACH based Algorithms

Algorithm	Extension over LEACH
Modified LEACH based stochastic cluster-head selection [263]	22% longer network lifetime
ER-LEACH [264]	30% longer network lifetime
VIBE [265]	18% longer network lifetime
EHEED [266]	87% longer network lifetime
EBRAMS[267]	82% longer network lifetime
Proposed DAHDA	120% longer average node lifetime 155% more delay in first node dies

3.8 Chapter Summary

DAHDA is an adaptive data aggregation algorithm for WSNs that is based on LEACH-C algorithm. Three variants have been presented: Basic DAHDA, Extended DAHDA (EDAHDA) and Modified EDAHDA. EDAHDA has added an extension to the basic variant and is based on the assumption that not all sensors readings are important in every round since the readings might be similar with readings of nearby sensors. Another extension of DAHDA i.e. Modified EDAHDA can handle sudden changes in the underlying data which can happen due to occurrence of events in the sensed field. Simulation results show that EDAHDA and modified EDAHDA extend the network lifetime and perform well in both evolving as well as non-uniform WSNs while maintaining the accuracy of the data. The proposed algorithm performs well in terms of residual energy, number of alive nodes, data accuracy, sudden burst detection, sensor distribution, lifetime of last node, first node and average node lifetime for uniform, non-uniform and evolving networks. The results also demonstrate the adaptivity of the algorithm by being able to capture the events of interest. The proposed algorithm has been simulated and compared with basic LEACH and other existing variants of LEACH in terms of the network lifetime without affecting the accuracy of the results.

Chapter 4

Cross-layer Energy based Clustering for Heterogeneous Wireless Sensor Networks

Increasing the lifetime of the network is one of the key research issues in Wireless Sensor Networks. It arises due to the limitations on the energy in the sensor nodes and the occurrence of hot-spots due to high network traffic. Sensor nodes residing in hot spot locations get drained out of energy very quickly, which leads to disruption in operations of the network. Thus, a cluster based technique for increasing the energy efficiency is required to extend the lifetime of the network.

In this chapter, Cross-layer Energy based Clustering (CEC) technique is proposed to form clusters of sensor nodes in hexagonal shape. The cluster head is opted on the basis of the ideal cluster head distance and remaining energy of sensor nodes. In order to make a balance between energy consumption and network traffic, the rotation of cluster heads is performed. In the proposed technique, cluster head selection probability changes dynamically.

The energy level of a node also gets affected due to collisions happening during the transmissions. These collisions can be prevented by applying protocols which are free of contention. In addition to this, slots are allocated by cluster head to all member nodes within a cluster on the basis of their remaining energy so that nodes can switch to sleep mode. In order to reduce the consumption of energy, data aggregation is used on the basis of the remaining energy of the cluster head.

This chapter presents an energy efficient clustering based solution for data aggregation problem in heterogeneous type of networks. The underlying heterogeneous network consists of three types of nodes: *Optimal Nodes*, *Distinguished Nodes* and *Divine Nodes*. The energy levels of these types of nodes are different. The proposed technique works in three phases namely Setup Phase, Slot Allocation phase and Steady Transmission Phase. The level of data aggregation varies according to the residual energy level.

4.1 Preliminaries for CEC Technique

The proposed Cross-layer Energy based Clustering (CEC) technique addresses several issues with the following advantages over most of the existing techniques:

1. The proposed CEC algorithm utilizes the concept of dynamic data aggregation which increases the scalability and reliability in networks with high density. The level of data aggregation changes dynamically according to the current energy level.
2. There is a proper selection of MAC protocol while scheduling the tasks of MNs. Time Division Multiple Access (TDMA) is used for scheduling and transmission which performs better than Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) in terms of sleeping time, energy efficiency, reliability and lesser packet drop ratio.
3. The proposed technique is based on heterogeneous networks in which different types of nodes are considered according to the varying energy level to address the issue of hot spot regions.

The preliminaries for the development of CEC has been reviewed and summarized in this section. These preliminaries include the review on:

- Network Model
- Energy Consumption Model

4.1.1 Network Model

A network consists of n number of nodes in a network field Q. Here the nodes are of three types based on energy level namely: *Optimal Nodes, Distinguished Nodes* and *Divine Nodes* and having no node is common. Thus, a three level heterogeneous network is assumed. E_o denotes the energy of an optimal node. f_1 denotes the percentage of nodes having energy more than optimal nodes, i.e. percentage of Distinguished and Divine nodes. Therefore, count of divine and distinguished nodes can be derived as $n.f_1$. Hence, the total number of optimal nodes will be $n.(1-f_1)$. The distinguished nodes are derived as f_2 percentage of $n.f_1$ nodes and have $g_1$times more energy than that of optimal nodes. The total number of distinguished nodes can be derived as $n.f_1.f_2$. Thus, the remaining $(1-f_2)$ percentage of $n.f_1$ nodes which come out to be $n.f_1.(1-f_2)$ in number, possess g_2 times higher amount of energy when compared to optimal nodes are known as divine nodes. Therefore, the count of divine nodes is $n.f_1.(1-f_2)$.

The total energy present at the initial stage of this heterogeneous network is represented by E_{total}, also known as *total initial energy* is given by Equation (4.1) as:

$$\begin{aligned} E_{total} &= n(1-f_1)E_0 + n.f_1(1-f_2)(1+g_2)E_0 + n.f_1.f_2.E_0(1+g_1) \\ &= n.E_0(1+f_1.(g_2+f_2.g_1)) \end{aligned} \tag{4.1}$$

Here, the energy of this three levelled heterogeneous network is $f_1(g_2+\ f_2.g_1)$ times higher than homogeneous networks.

4.1.2 Energy Consumption Model

The methodology proposed in the research work utilizes the model presented in [135] to calculate the consumption of the energy. This model considers three main components for energy consumption namely: power amplifier, transmitter and receiver. The consumption of energy in the transmitter comes under circuitry of the transmitter and the amplifier. The consumption of energy for receiver includes the reception of data comes under circuitry of the receiver. The power of the signal received by the receiver, represented by $p_r(d)$ is given by Equation (4.2):

$$p_r(d) = \frac{p_t G_t G_r \lambda^2}{(4\pi)^2 d^\beta ED_{loss}} \tag{4.2}$$

Here d denotes the distance between transmitter and receiver, p_t denotes the power of the transmitter, G_r represents gain of receiver and G_t represents gain of transmitter. ED_{loss} denotes energy dissipation in the transmission of the packet and λ as $wavelength\_carrier$. The variation of β (propagation loss factor) varies from 2 to 4 [133]. Thus, when G_r, G_t and ED_{loss} equals to unity, the power of the signal received is given by Equation (4.3):

$$p_r(d) = \frac{p_t\lambda^2}{(4\pi)^2 d^\beta} \tag{4.3}$$

For the successful reception of data packets, the power of the received signal should be greater than a threshold (p_{thresh}). Therefore, the power of the signal transmitted by the transmitter should be greater than threshold; $\frac{p_{thr}(4\pi)^2 d^\beta}{\lambda^2}$. Thus, transmitter's absorbed energy(E_t) is given by Equation (4.4):

$$E_t = \left(e_e + \frac{p_{thr}(4\pi)^2 R^\beta}{\lambda^2 d_R}\right) \times P\_length \; = \left(e_e + e_a R^\beta\right) \times P\_length \tag{4.4}$$

Here $e_a = \frac{p_{thr}(4\pi)^2}{\lambda^2 d_R}$, representing per bit energy utilized in the RF amplifier of the transmitter, d_R denotes the sending /reception bit rate for every node of the network. The maximum range of transmission for node is denoted by R. Here, P_length denotes bit count of packet to be transmitted and e_e denotes the consumed per bit energy of the transmitter. e_l denotes energy spent per second to listen to radio environment. e_a, e_l and e_e are derived from designed transceiver characteristics. Thus, on the basis of [218], sensor node's absorbed energy per second in the three states namely transmit, receive and listen is represented as Equation (4.5-4.7):

$$E_t = (e_e + e_a \mathrm{R}^\beta) N_t \tag{4.5}$$

$$E_r = (e_e) N_r \tag{4.6}$$

$$E_l = e_l T_l = e_l(1 - T_t - T_r) = e_e d_R (1 - T_t - T_r) \tag{4.7}$$

E_t, E_r, E_l denote energy consumption of transmission, reception and idle state for traffic burst respectively. Reception/Transmission time of traffic in between clusters is represented as T_r and T_t respectively, N_r and N_t denote bits of data-traffic received/transmitted respectively.

$$T_t = \frac{N_t}{d_R} \tag{4.8}$$

$$T_r = \frac{N_r}{d_R} \tag{4.9}$$

Time spent to sense the radio channel (total time is taken as 1 second) $i.e. T_l = 1 - T_t - T_r, (0 \leq T_l \leq 1);\ 0 \leq (1 - \frac{N_t}{d_R} - \frac{N_r}{d_R}) \leq 1.$

Assuming the data traffic to be static and N_t and N_r equal to N, Equation (4.10) presents N as:

$$0 \leq N \leq \frac{1}{2} \times d_R \times second \tag{4.10}$$

Thus, the maximum transmitted data i.e. d_{R_max} in each second is given as $\frac{1}{2} \times d_R$ bits. The conditions for this amount of data being transmitted are:

(i) Members of the cluster become deaf to radio environment

(ii) $\frac{1}{2}$ second is spent for the reception of packets

(iii) Other $\frac{1}{2}$ second is spent for transmission.

4.2 Optimum Distance for Cluster Head

In proposed CEC technique, clusters are formed in hexagonal shape. In an equi-symmetric cell of hexagonal shape, the optimum route comes out be a straight line directing from $Source$ towards $Sink$ assuming multiple hops. The Figure 4.1, shows that transmission of data packets which takes place from $Source$ to $Sink$; where L denotes the distance between sink and source.

Inter-Cluster Heads distance is considered as D and number of clusters is denoted by $nCluster$ as given by Equation (4.11) and Equation (4.12):

$$R = \sqrt{13}\,.r,\ D = \sqrt{3}.r \tag{4.11}$$

$$nCluster = \frac{L}{D} = \frac{L}{\sqrt{3}.r} \tag{4.12}$$

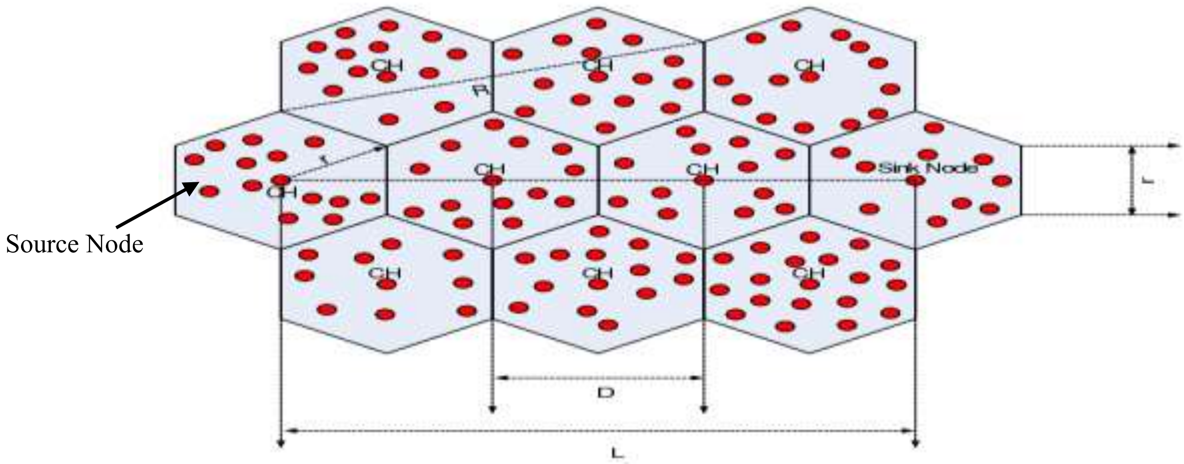


Figure 4.1 Clustering based Hexagonal Structure for End to End Multi-hop Transmission

The underlying considerations of proposed CEC are:

(i) The network traffic is still i.e. the data rate is unchangeable.

(ii) For the hexagonal model of cluster[138]; r denotes the length of hexagon's side and optimum distance for cluster head as shown in Figure 4.1; the maximum range of transmission for node is denoted by R which is derived in Equation (4.11). Within this range R, any two nodes located in neighboring clusters can receive/transmit data. The consumption of energy for the multiple hops transmission (E_s) on the basis of energy consumption model is given by Equation (4.13):

$$E_s = E_t + E_r + E_l$$

$$E_s = \left(e_e + e_a R^{\beta}\right)N + (e_e)N + e_e d_R\left(1 - \frac{2N}{d_R}\right)$$

Putting value of d_R in E_l component from Equation (4.10). So, E_s comes out to be:

$$E_s = \left(2e_e + e_a R_i^{\beta}\right)N \tag{4.13}$$

The total energy consumption can be calculated as given by Equation (4.14):

$$E_{overall} = nCluster.E_s = \frac{L}{\sqrt{3r}}\left[2e_e + e_a\left(\sqrt{13}.r\right)^{\beta}\right]N \tag{4.14}$$

1st derivative of $E_{overall}$ is calculated w.r.t 'r', to derive energy consumption's minimum value.

This is done by setting $\frac{d}{dr}(E_{overall}) = 0$

$$\frac{d}{dr}(E_{overall}) = \frac{L}{\sqrt{3}}(e_a N(\sqrt{13})^{\beta}(\beta - 1)r^{\beta-2} - \frac{2e_e N}{r^2}) \tag{4.15}$$

By using Equation (4.15), the optimum value of r is deduced, which is given by Equation (4.16):

$$r = \frac{1}{\sqrt{13}}\left(\frac{2e_e}{(\beta - 1)e_a}\right)^{1/\beta} = \frac{1}{\sqrt{13}}\left(\frac{2e_e\lambda^2 d_R}{(\beta - 1)p_{thr}(4\pi)^2}\right)^{1/\beta} \tag{4.16}$$

By calculating the right set of parameters in Equation (4.16), the optimum value of r is calculated. Here r in turn depends on β (propagation loss factor) having values within [2,4] and the traffic of the network. Figure 4.2 shows the relation between r and β, optimum value of distance for CH and side length of hexagonal cell goes down when β grows, while traffic in the network remains constant. Moreover, there exists an inversely proportional relationship between the number of CHs and the consumption of energy. On the basis of Equation (4.13), when CHs count decrements, the side length of cluster increases. It is concluded that the CHs dominate consumption of energy as the energy consumption in the transmitter amplifier present in every CH increases at a very fast pace.

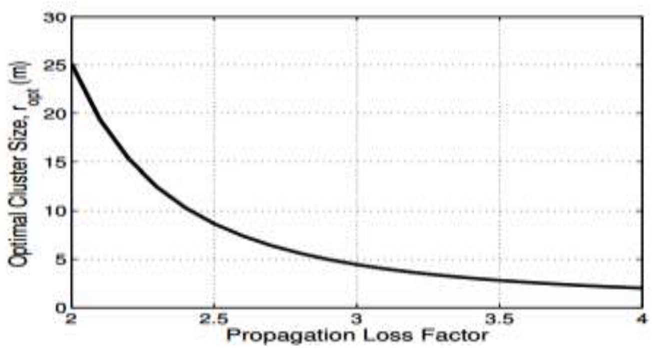


Figure 4.2 Optimal CH Distance r Variation with β

4.3 Cross-layer Energy based Clustering Technique

In WSNs, nodes are densely deployed and are operated in autonomic manner. Nodes are generally powered with batteries and left aloof in severe conditions, thus recharging or replacement of batteries is difficult. So, the major problem in WSNs is to extend the lifetime of the network due to limited energy. The nodes placed near the sink become hot spot regions, as these nodes have to relay huge volume of traffic and may result into network hole. The other major issue arises, when some nodes receive and transmit the data simultaneously which leads to data congestion and collisions. Hence, solving the problems of energy drainage, network hot spot and congestion are very critical for extending the lifetime of the network.

Clustering is one of the mostly used mechanisms to solve such problems, in which sensors nodes are divided into various groups on the basis of their location. It reduces the frequency of data transmission between the nodes and Base Station (BS). The data is collected by Cluster Heads (CHs) and then reported to the BS in an aggregated form. Hence, it decreases the load on the channel and increases the stability. But there exist computation and communication overheads while electing the CH and formation of clusters [138]. In addition to this, it seems that the lifetime of CH is shorter as compared to the other Member Nodes (MNs), as same CH works for a long period of time. These problems of clustering can be avoided by shuffling the role of CH among MNs and utilizing appropriate MAC protocol.

By the selection of correct MAC protocol, reliability and sleeping duration of nodes are increased which further increase the network lifetime. The MAC layer protocol can be *contention based* or *free of contention* or *hybrid* in nature. The protocols based on contention like Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) suffer from huge contention among MNs for the utilization of resources. In addition, it is observed that CSMA/CA performance degrades when the density of nodes and load on the network increases. Contention increases packet drop ratio and thus degrades the reliability of WSNs.

In order to resolve this issue, protocols like Time Division Multiple Access (TDMA) which are free of contention are preferred. Here time is assigned to MNs in the form of time slots for transmission, based on a slot scheduling algorithm. By increasing the sleeping time in TDMA, the reliability of the network can be improved. It is highly utilized in real time processes and the most properly used technique for communication i.e. converge cast, in which CH receives data from MNs and relays it to BS.

In this chapter, a clustering based methodology i.e. Cross-layer Energy based Clustering (CEC) Technique has been proposed. The main objective is to reduce the frequency of data transmissions to the sink by selecting an efficient Media Access Control protocol. A data aggregation technique is also incorporated based on the energy present in the nodes. By removing the redundancy, packet size is reduced, which further brings down the congestion of the network.

The proposed CEC works for heterogeneous WSNs, in which there are three types of sensor nodes namely: *Optimal Nodes, Distinguished Nodes* and *Divine Nodes*. The initial energy levels of these nodes are in the given order: *Distinguished Nodes* > *Divine Nodes* > *Optimal Nodes*. In order to minimize the energy consumption, the optimum distance from MNs to the CH is deduced. The energy consumption is proportional to the deduced optimal distance. The entire WSN is divided into clusters which are hexagonal in shape. Here every MN selects a CH on the basis of deduced optimum distance and the residual energy in the nodes having value greater than threshold value (average residual energy). Every MN is assigned a time slot on the basis of remaining energy and sleep duration slot for increasing the lifetime of the network.

CEC presents a self-organizing approach for clustering in WSNs, in which every node performs tasks like field parameters anticipation, formation of packets of data and communications with the CH. A unique identifier is assigned to every node. Each node is capable of switching between sleeping and active state. CH can relay the aggregated information to the successive hop cluster head leading towards BS. Every node transmits its current location, current residual energy, probability of becoming CH based on remaining energy, Cluster-Identity and the CH-Identity with neighbours who are one hop distant to reduce the cost of idle listening. CEC works into three phases namely *Setup Phase, Slot Allocation phase* and *Steady Transmission Phase* as described below:

4.3.1 Setup Phase

In this phase, the grouping of sensor nodes is done in the form of clusters in a hexagonal shape. It helps in selecting the CHs nearer to the optimum CH location. The optimum length of the side of every cluster is calculated based on mathematical model described in Section 4.2. Here the value of β is considered as 2. After this, cluster is formed by utilizing the optimum side length for solving the problem of hot spot formation and to balance the energy dissipation. Initially, the CHs are opted arbitrarily from distinguished sensors as these nodes have maximum energy present in the network. After this selection is done, the selected CH broadcasts its selection status to all the neighbours who are one hop apart; nodes receiving this status will change their CH status.

Determining the probability of CH selection on the basis of Residual Energy

In LEACH, the process of selecting the cluster head is divided into rounds. In every round, node takes a decision of becoming a CH on the basis of threshold $T(n_i)$, determined by taking into account the predetermined CHs percentage and count of previous selections of the sensor node as a CH. The decision of becoming a CH is done on the basis of an arbitrary number chosen between zero and one. If the selected number is less than $T(n_i)$, node is opted as CH for present round as given by Equation(4.17):

$$T(n_i) = \begin{cases} \frac{k_i}{1-k_i\left(r_c \bmod \frac{1}{k_i}\right)} & if\ n_i \in G \\ 0 & Otherwise \end{cases} \tag{4.17}$$

Here k_i signifies the chances/probability of a node (n_i) to elect as a CH or required CH percentage, r_c denotes the ongoing round and G denotes the node set that has not been opted as CH in previous rounds; expressed as $1/k$. The usage of the threshold calculation is to determine that every node is going to be elected as a CH, only once in the span of $\frac{1}{k}$ number of rounds. In heterogeneous network model, each node type has a distinct value of reference for k_i. The values of probabilities (k_i) for optimal, divine and distinguished nodes are calculated as given by Equation (4.18):

$$k_i = \begin{cases} \frac{k_{opt}E_i(r_c)}{\left(1+f_1(g_2+f_2.g_1)\right)E_{avg}(r_c)} & \text{if } n_i \text{ is an optimal node} \\ \frac{k_{opt}(1+g_2)E_i(r_c)}{\left(1+f_1(g_2+f_2.g_1)\right)E_{avg}(r_c)} & \text{if } n_i \text{ is a divine node} \\ \frac{k_{opt}(1+g_1)E_i(r_c)}{\left(1+f_1(g_2+f_2.g_1)\right)E_{avg}(r_c)}, & \text{if } n_i \text{ is a distinguished node} \end{cases} \tag{4.18}$$

Thus threshold value for electing the CH is derived for optimal, divine and distinguished nodes by putting Equation (4.18) into Equation (4.17):

$$T(n_i) = \begin{cases} \frac{k_i}{1-k_i.\left(r_c \, mod \frac{1}{k_i}\right)} \text{if } n_i \in G1 \\ \frac{k_i}{1-k_i.\left(r_c \, mod \frac{1}{k_i}\right)} \text{ if } n_i \in G2 \\ \frac{k_i}{1-k_i.(r_c \, mod \frac{1}{k_i})} \text{ if } n_i \in G3 \end{cases} \tag{4.19}$$

where $G1, G2, G3$ represent the chunk of optimal nodes, divine nodes and distinguished nodes which has not been opted as CHs in the span of previous $\frac{1}{k_i}$ rounds respectively. n_i denotes the count of particular type of node. However, it can be seen that after few rounds, the residual energy level of some distinguished and divine nodes comes down to same level as that of optimal nodes because of the nodes are being repeatedly selected as CHs. The energy of divine and distinguished nodes gets continuously exhausted. Therefore, to prevent the unbalanced energy dissipation in the network, an enhanced function has been proposed for the calculation of probabilities of optimal, divine and distinguished nodes. The changes are proposed on the basis of level of absolute remaining energy i.e. $T_{absolute}$. Here, $T_{absolute}$ denotes the value where both types of nodes (divine and distinguished) have same energy level as that of the optimal nodes. This suggests that with $T_{absolute}$ the probability for all optimal, divine and distinguished nodes to become CH comes down to the same level. c_o controls the number of clusters. If c_o is high, then more number of CHs are sending their data to BS. The probabilities proposed for selecting the CH are given by Equation (4.20):

$$k_i = \begin{cases} \frac{k_{opt} E_i(r_c)}{(1+f_1(g_2+f_2.g_1))E_{avg}E(r_c)} \text{ for optimal nodes; if } E_i(r_c) > T_{absolute} \\ \frac{k_{opt}(1+g_2)E_i(r_c)}{(1+f_1(g_2+f_2.g_1))E_{avg}(r_c)} \text{ for divine nodes; if } E_i(r_c) > T_{absolute} \\ \frac{k_{opt}(1+g_1)E_i(r_c)}{(1+f_1(g_2+f_2.g_1))E_{avg}(r_c)} \text{ for distinguished nodes; } E_i(r_c) > T_{absolute} \\ c_o * \frac{k_{opt}(1+g_2)E_i(r_c)}{(1+f_1(g_2+f_2.g_1))E_{avg}(r_c)} \text{ for optimal, divine, distinguished nodes; otherwise} \end{cases} \tag{4.20}$$

$T_{absolute}$, is expressed as:

$$T_{absolute} = zE_o \tag{4.21}$$

where, $z \epsilon (0, 1)$. In the real scenarios, it is possible that divine and distinguished nodes may not become CHs in some number of rounds. Also there can be a scenario, when only some of these nodes become CHs and same is applicable to the optimal nodes. Therefore, the exact value of z cannot be derived. However, by performing number of simulations in randomized topologies, the value nearest to z can be estimated by changing it to achieve the finest results. The best result for z is attained at *0.7*. Hence, $T_{absolute} = (0.7)\ E_o$.

The calculation of probabilities is done on the basis of the average energy present in network (represented by E_{avg}) in the current round denoted by r_c. Therefore, the average energy is calculated as:

$$E_{avg}(r_c) = \frac{1}{n} E_{total}(1 - \frac{r_c}{R_{total}}) \tag{4.22}$$

Here R_{total} signifies the total number of rounds in the network lifetime expressed as:

$$R_{total} = \frac{E_{total}}{E_{overall}} \tag{4.23}$$

Here $E_{overall}$ is the dissipated value of energy in the network during current round given by Equation (4.14).

Re-Election of Cluster Head

The re-election of cluster head is done on the basis of probability (k_i) and derived optimum Cluster Head distance for the node to minimize the consumption of energy and to improve the lifetime of the network. Every MN calculates a value based on Equation (4.24), here n_c denotes the number of nodes which are single hop apart, the distance from node i to j is represented by R_{ij} and r represents the mean distance value from sensor node to all neighbors which are one hop apart. Thus all MNs broadcast a message with the calculated probability (k_i), remaining energy, mean distance value from sensor node to all neighbor which are one hop apart, the Cluster-Identity, the CH-Identity and location:

$$\sum_{j=1}^{n_c} \frac{R_{ij}}{n_c} = r_i \;\; for\ i = [1, n_c] \tag{4.24}$$

$$f\ (k_i, r_i) = \alpha \frac{r_i}{2r} - ((1-\alpha).\,k_i) \tag{4.25}$$

Here, r_i represents the mean distance value from sensor node to all neighbor which are one hop apart. α is weight which is assigned to the parameters considered: distance and residual energy based on k_i. Here, $k_i$is the probability of a node to become CH based on the basis of residual energy level. The relative importance of input parameters is decided by the weight function α. ($\alpha = 0$) means only residual energy parameter is considered and $\alpha = 0.5$ means residual energy and average distance are considered while making decision as given by Equation (4.25). ($\alpha = 1$) means only the average distance parameter is considered without paying any attention to energy consumption. The cluster heads are opted by following a rule according to which a node closest to optimum location of cluster head with minimum value of distance and maximum probability (k_i), is selected for the successive round. MN which minimizes the value of function given by Equation (4.25) is opted as a cluster head.

4.3.2 Slot Allocation Phase

After the election of cluster head, there is a need of synchronization among all elected CHs, members of the clusters and BS for communication. In addition, all the members of the clusters have an additional state for listening in order to synchronize their acts with neighboring MNs. In every period of contention, all MNs turn their radios on. A short preamble is utilized by cluster head to keep its cluster members in sync.

A newly elected CH broadcasts a message for synchronization, which will contain its identity and Cluster-Identity. MNs will send a message of acknowledgement upon receiving the message from cluster head. The cluster head will start the allocation of time slots to its cluster members on the basis of remaining energy. A channel allocation strategy is adopted on the basis of free of contention communication to minimize the consumption of energy. It partitions the time in slots of size ts. Consecutive time slots construct a frame. The length of the frame is taken as T. Therefore, every node repeats its actions after a time period T, assuming that all cluster heads contain equal length of frame. Energy is also consumed by nodes in listening state. Thus for minimizing the consumption of energy these MNs will enter the sleep state. Here the number of time slots is same as the count of MNs in a cluster.

A TDMA schedule is generated by the CH and is broadcasted to the MNs of the cluster. The scheduling algorithm assigns slots $1 \leq t \leq T$ of time to all members. MNs can switch among the four states: *receiving, listening, sleeping* and *transmitting.* It is assumed that all MNs co-exist in perfect synchronization and every member of a particular cluster which is not in transmitting state, goes to sleep state. The main goal of this scheduling is to reduce the state transitions of the nodes in order to reduce the energy consumption. The schedule is generated in such a way that the members gets out of sleep mode only two times: first for receiving the data from its CH and second for the transmission of data to its CH. According to this methodology, time slots are assigned by BS to CHs which in turn assign slots to MNs. Figure 4.3 shows the packet header of MN packets.

Figure 4.3 shows cluster head as starter (CH1), forwarder (CH4) and sink as cluster head's successive hop (CH7). The cluster head of the cluster at the utmost end is represented by CH1. CH1 doesn't require hearing data from other cluster heads for sending data to BS. CH1's frame is partitioned in five slots: for (i) hearing messages from members (ii) hearing messages from sink, (iii) transmitting data to successive hop cluster head and (iv) sending data to cluster nodes and (v) sleep state.

The time slots are assigned to MNs based on their residual energy. MN having minimum amount of energy gets the 1st slot for transmitting the data to cluster head. Member node goes to sleep state after transmitting the data and rises up for the reception of data from cluster head. After this, forwarding state of cluster head is initiated towards the sink. Sleeping state for every CH is decided on the basis of its location. CHs at the utmost end get longer sleep in comparison to those which are nearer to the BS. Light green colored time slots in the frame are reserved for listening; dark green slots represent up-link; yellow color represents down link. The gap between CH1 and CH4 shows the presence of an additional state for listening and waiting for the data coming from CH1.

State of listening for cluster head, only relays three cluster heads' data because of the hexagonal structure of cluster. The count of slots present in the sink is also six due to its hexagonal structure. Inside down-link, sink sends information to all cluster heads via single hop transmission. System stabilizes as soon as all MNs get slots in the cluster head frame and every cluster head get a slot in the frame of the sink. The size of every packet and length of every time slot is same, except for the CH slot and the slot of sink transmission.

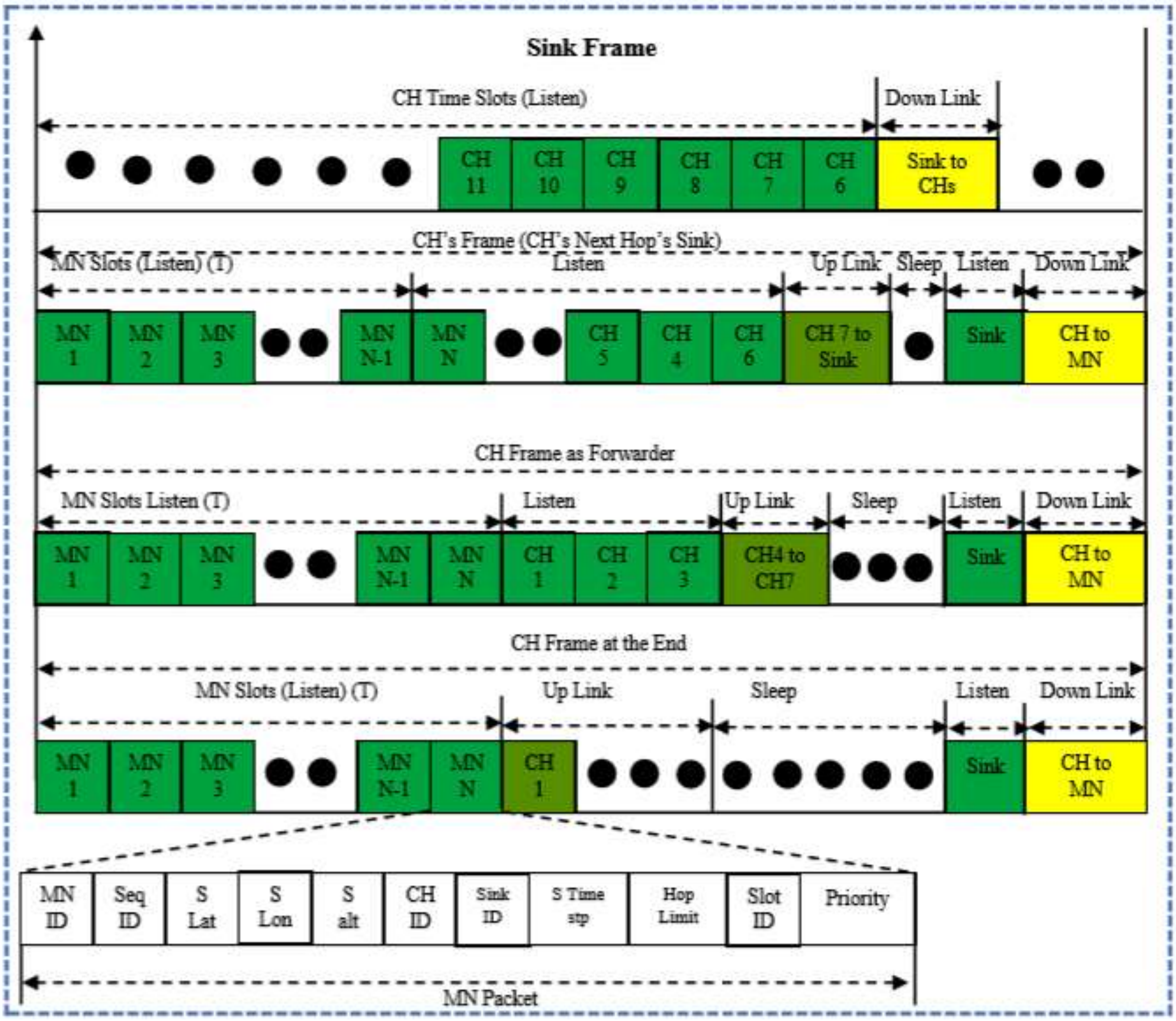


Figure 4.3 TDMA for CH and Sink Frames

4.3.3 Steady Transmission Phase

In this phase, two types of data transmission take place: *inter-cluster* and *intra-cluster* routing. MNs forward their data to cluster head in intra-cluster routing while cluster head forwards the fused data to the BS or the next hop cluster head in inter-cluster clustering using data aggregation. This aggregation/fusion scheme comprises of CH, cluster members and the sink. This scheme is comprised of distinct levels of clustering. The level of clustering depends on the residual energy of the cluster head. The level of aggregation is higher when there is low residual energy. Window of energy of a particular level is calculated by the division of overall energy of the nodes by count of aggregation levels.

Aggregation of data comprises of distinct components for the reduction of data. For instance, temperature of node is taken as parameter and the objective is to aggregate the data of different nodes with the required level of precision. For this fusion, data with high similarity is preferred. A decision making scheme is proposed which is aware of the cost to opt two nodes to aggregate data for reducing the packet size. Additionally, this scheme prevents extreme values from getting ignored and thus increases the precision of data. The level of aggregation of every cluster head is utilized for controlling the aggregation rate on the basis of remaining energy of each cluster head. The cost function utilizes a weight parameter for the calculation of data aggregation cost for two MNs a and b. p_{set} is taken as the parameter set and the value for parameter i of nodes a and b are represented as a_i and b_i respectively. The weight for i^{th} parameter is considered as w_i. Here, max_tdenotes the maximum value reported for parameter i in time interval t. The cost is then calculated as given in Equation (4.26):

$$cost = \sum_{i \in p_{set}} w_i * \left|\frac{a_i - b_i}{max_t}\right| \tag{4.26}$$

Here,$w_1 + w_2 + w_3 + \ldots + w_j = 1$. Utilizing the notation and considering the system which utilizes only two parameters p_{set}: {temperature, position}. The weights for the position and temperature are considered as 0.5. Moreover, considering the weights to vary from 0 to 1, the point for best results of the components of the decision can be determined. The size of Window of Energy (EW) for every level of aggregation is determined in Equation (4.27), here E_{cap} represents the average energy holding ability of MN and the count of levels of aggregation is denoted by l_a. EW is then calculated and assigned to each level of aggregation. Initially, each and every cluster head checks E_{res} each time before transmitting to the sink node and compares it with the size of level of aggregation. If a change is found, then an updation is made to its level of aggregation. Table 4.1 shows the varying level of data aggregation based on the residual energy of CH. As the residual energy decreases, the level of data aggregation is increasing to minimize the energy consumption.

$$EW = \frac{E_{cap}}{l_a} \tag{4.27}$$

Table 4.1 Aggregation Levels based on Residual Energy Range

Aggregation Level	Residual Energy of Cluster Head
Level 0	1.6 to 2.0 J
Level 1	1.1 to 1.5 J
Level 2	0.6 to 1.0 J
Level 3	0.1 to 0.5 J

After data aggregation, it will be transmitted to the next CH towards the sink. For informing all probable cluster head candidates heading towards sink, cluster head broadcasts signal to the next cluster head. As the final step, cluster heads form a path in the direction of the sink. Transmitting data via this path reduces consumption of energy as compared to direct-forwarding by cluster heads to sink. The cluster heads perform their functions while switching into these three modes:

- **Initial mode:** During the initialization phase, every cluster head is assigned with predefined values.
- **Route broadcasting mode:** Here signals are broadcasted to create a route between the clusters.
- **Establishment of routes mode:** Routes are generated with the collaboration of neighbour routes.

Algorithm 4.1 shows the complete description of the steps followed in the proposed CEC algorithm working in three phases:

Algorithm 4.1 Cross-layer Energy based Clustering Technique

Input Data: Number of nodes and their values of residual energy, node id, location (distance from the BS in X and Y position),E_{total} denotes the initial energy of the network. $nCluster$ denotes the number of clusters formed. n_slot denotes the total number of slots assigned to member nodes. n denotes total number of nodes, $E_{avg}(r_c)$ denotes average energy of current round.

Begin

1. Initialize Node List and residual Energy value for each node.

2. **while** $E_{avg}(r_c) > 0.10 \times E_{total}$

3. **for** $i \in n$ **do**

Calculate its probability of becoming cluster head for present round as given in Equation (4.20).

end

4. Calculate the optimal side length for hexagonal clusters using Equation (4.16).

5. Select the optimum number of clusters based on optimal side length and (propagation loss factor) based on mathematical model of Section 4.2.

6. Select cluster head for each cluster based on the probabilities calculated in Step 6 and optimum cluster head location calculated based on mathematical model of Section 4.2.

7. **for** $i = 1\ to\ nCluster$ **do**

CH broadcasts message containing *cluster_id* to members nodes.

Every member sends acknowledgement in return.

Consecutive time slots of size ts are allocated to each member node via channel allocation strategy on the basis of residual energy.

TDMA schedule is generated and broadcasted to each member node.

end

8. **for** $i = 1\ to\ nCluster$ **do**

for $i = 1\ to\ n\_slot$ ***do***

Cluster member send its data to CH in its respective slot.

end

a. *Each* Cluster Head calculates its residual energy.
b. Based on the residual energy Cluster Head calculate its level of aggregation and energy window pertaining to it using Equation (4.27).
c. Aggregation cost is calculated by Cluster Head utilizing weight function for aggregation level using Equation (4.26).
d. The aggregation of data is done by Cluster Head by packet size reduction.
e. Cluster Head forms path heading in the direction of the sink.

end

End

4.4 Experimental Setup and Results

In this section, simulation results are presented for proposed CEC and existing approaches SOEECP[146], LCM[148] and EEPCA[235] for three-level heterogeneous WSNs. WSNs consist of $n = 1800$ nodes which are randomly deployed for region of interest. The performance of proposed CEC has been evaluated in terms of network lifetime (centered and non-centered sink), energy consumption, number of packets received and data accuracy using 100 iterations. The experimental settings for simulations are described in Table 4.2.

Table 4.2 Simulation Settings for Experimental Setup

Simulation Parameters	Value
Total number of nodes	1800
Area of deployment	$400\ m \times 400\ m$
Count of Sinks	2
Location of 1^{st} Sink	In the center (200,200)
Location of 2nd Sink	Randomly deployed
Optimal Node's Initial Energy	1J
Divine Node's Initial Energy	2J
Distinguished Node's Initial Energy	3J

Frequency of carrier (f)	$0.24 \times 10^5\ Hz$
Rate of Data Transmission (d_R)	$2.5 \times 10^5\ bps$
Factor of Propagation (β)	2
Time of Simulation (t)	$2 \times 10^{-2}\ s$
Number of member nodes in each cluster formed	20 (equal count for each type of node)
Individual cluster area on the basis of optimal hexagon side length (25.07)	$1631m^2$
Number of clusters having same number of cluster	90
Threshold power (p_{thr})	$0.8 \times 10^{-16}\ W$
Average Node Energy	2J
e_e	$3.32 \times 10^{-7} bit^{-1}$
e_a	$8 \times 10^{-11}\ J/bit/m^2$
d_{R_max}	1.25×10^5 bits / sec

The association of MNs with their cluster does not change in the whole simulation as the distance between MNs remains same. A periodic transmission of messages to the sink node takes place. While comparing the technique with SOEECP, LCM and EEPCA, 802.15.4 MAC is utilized. The optimal length of the side of the hexagon is obtained by Equation (4.16); ($r \to 25.07m$). The analysis of consumption of energy and network lifetime with variable number of MNs is performed for proposed CEC and existing SOEECP, LCM and EEPCA techniques. A slot for transmission is provided to every cluster head in every frame of sink. As per the structure which is a hexagon in shape, sink has 6 neighboring cluster heads. Frame length required by these six CHs comes out to be $ts \times 6s$; here duration of the slot is denoted by ts.

4.4.1 Performance Evaluation

The proposed CEC is compared for distinct values of weight factor α with existing techniques such as SOEECP, LCM and EEPCA. In order to observe location effect on lifetime of the network and consumption of energy, the sink is deployed at different location randomly.

Test Case 1: Energy Consumption Vs. Number of Nodes

The value of energy consumption is calculated for proposed CEC technique and existing techniques LCM, EEPCA and SOEECP with different number of nodes. The value of energy consumption is calculated on the basis of energy model presented in Section 4.1.2 and values of parameters are shown in Table 4.2. Initially, the average energy for each node $(Initial\ energy\ of\ (optimal\ node\ +\ distinguished\ node + \ divine\ node)/3)$ is taken as 2J. With the increase in the count of sensor nodes, the change in consumption of energy is shown in Figure 4.4 and Figure 4.5 for centered and non-centered Sink. Each node consumes energy for communication and computation. So, with the increase in the number of nodes, the value of energy consumption also increases.

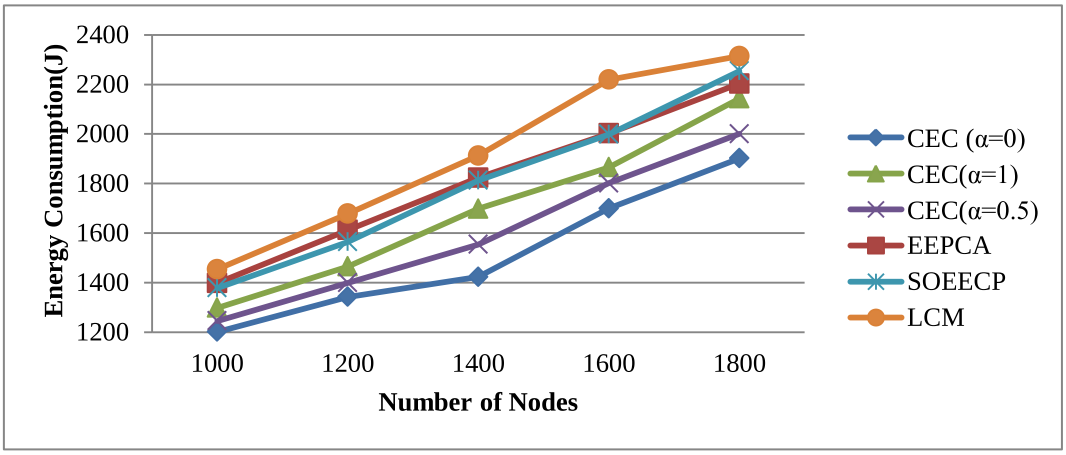


Figure 4.4 Analysis of Energy Consumption (centered – Sink)

The energy consumption in case of non-centered Sink is more as compared to the network scenario in which Sink is deployed in the center position. The rationale behind this performance is, for transmitting the data over a long distance more energy is consumed as compared to the shorter distance. It is evident in Figure 4.5, that proposed CEC $(\alpha = 0.5)$ better as compared to the existing techniques.

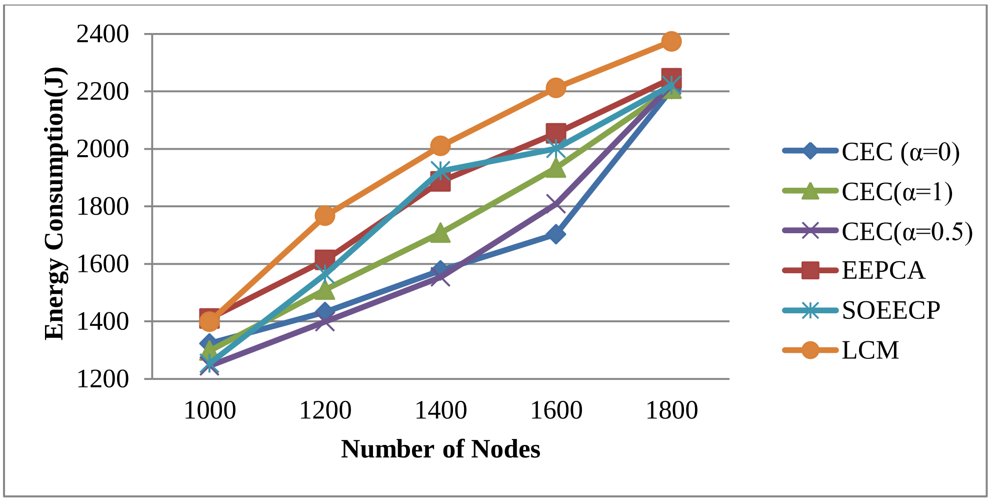


Figure 4.5 Analysis of Energy Consumption (Non-Centered – Sink)

Test Case 2: Fraction of Survived Nodes

The fragment of nodes remaining alive over the time is termed as *Fraction of Survived Nodes*. It provides the estimation of the time duration for which network remains active. Setting α to an optimum value prolongs the fraction of survived nodes as shown seen in Figure 4.6 and 4.7 for centered and non-centered sink placement respectively.

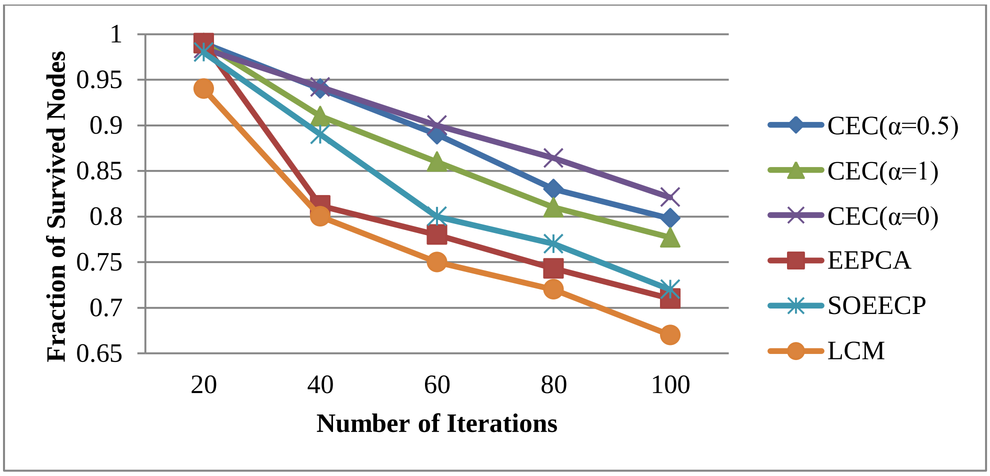


Figure 4.6 Comparative Analysis of Fraction of Survived Nodes (centered sink)

It can be observed that CEC ($\alpha = 0$) results in lesser energy consumption when compared to other existing techniques. CEC with α as 0.5 and 1 performs better than the existing techniques. CEC with α as 0 makes 64% consumption of the entire network's energy before the death of 30% of the network nodes. CEC increases the energy efficiency by making the MNs to sleep between transmitting and receiving time of packets.

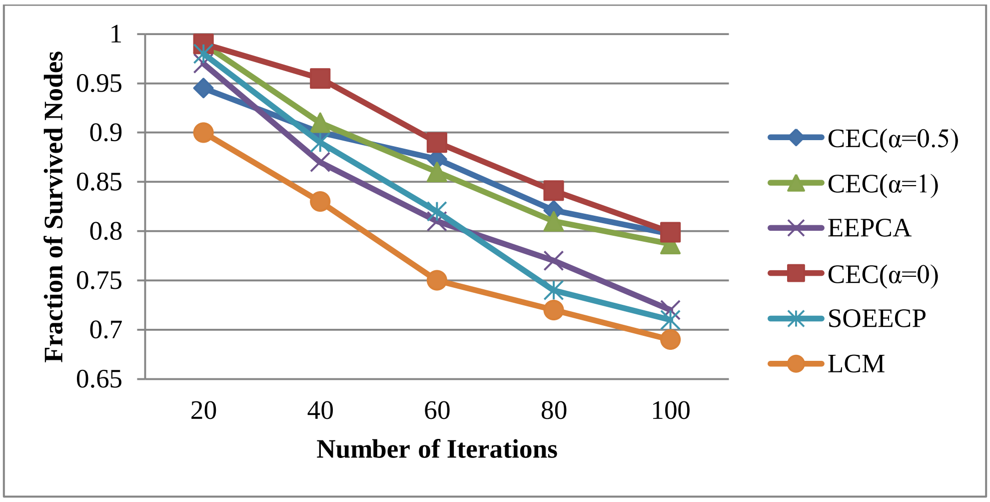


Figure 4.7 Comparative Analysis of Fraction of Survived Nodes (Non-centered sink)

Figure 4.6 shows the fraction of survived nodes (assuming sink in center) on the basis of changing values of α, where $\alpha\,\epsilon(0,1)$ and existing techniques LCM, EEPCA and SOEECP. Figure 4.6 depicts that for LCM, SOEECP and EEPCA respectively, the fraction of survived nodes decreases in a drastic manner as compared to proposed CEC variants. The rationale behind this output is that SOEECP, EEPCA and LCM do not consider the remaining energy of the nodes while selecting the cluster head, which may results into the election of cluster head with lesser energy. It results into the early death of a node and loss of data. CEC $\alpha = 0)$ only considers the remaining energy while opting for cluster head, resulting into higher lifetime of the network as compared to the existing techniques.

In CEC with α as 0, the nodes on the low energy side will never be elected as cluster heads when there is a presence of nodes with higher energy in the cluster. Whereas, in the case of CEC with α as 0.5, the cluster head selection is done on the basis of cost of communication and remaining energy of the nodes; resulting in the moderate lifetime of the nodes in comparison to other methodologies. Placing the sink in the center reduces the count of packets to be sent to sink by fusing these packets and thus increasing the fraction of survived nodes. Considering the sink in center as shown in Figure 4.6, fraction of survived nodes said to be 30% of nodes drained completely out of energy. The fraction of survived nodes of the proposed or the said techniques is as follows:

LCM:EEPCA:SOEECP:$CEC(\alpha = 1)$:$CEC(\alpha = 0.5)$:$CEC(\alpha = 0) =$ 0.62: 0.58:0.88:0.94:0.96:1; it proves that CEC when α is set as 0, sink is placed in center performs better with respect to fraction of survived nodes as compared to the existing techniques. Since CEC with α as 1 and SOEECP does not consider the remaining energy of the nodes while opting the cluster head, the fraction of survived nodes shows drastic decrement in comparison to other techniques. Apart from this, the fraction of survived nodes for CEC (α = 0.5) is almost 3% more when sink is placed in center as compared to non-centered sink as clearly depicted in Figure 4.7. The rationale behind the difference in this performance is: less energy is consumed in centered-placed sink as the data transmission involves lesser distance as compared to the non-centered sink.

Test Case 3: Packet Delivery Ratio

To determine the effect of loss of packets due to election of cluster heads with less amount of energy, here the count of received packets by the sink for the different techniques is analysed and compared.

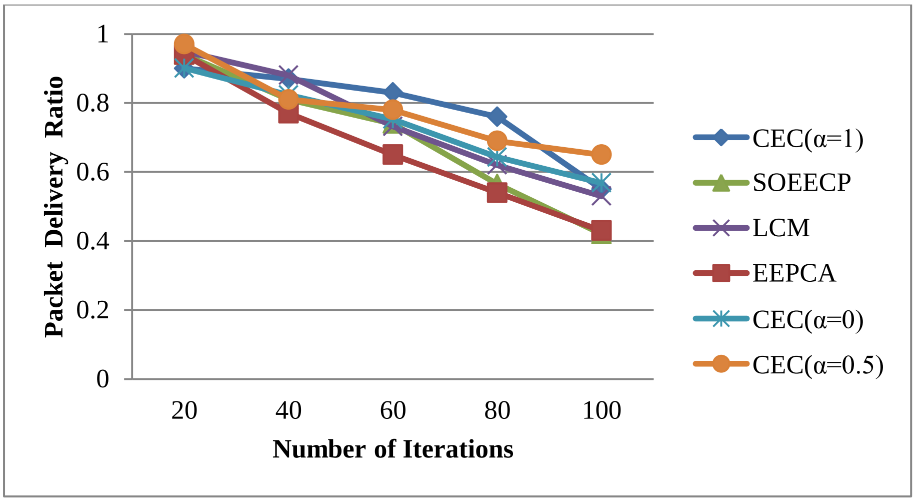


Figure 4.8 Comparative Analysis of Packet Delivery Ratio (Centered Sink)

The value of number of packets received with respect to number of iterations has been calculated for proposed CEC, EEPCA, SOEECP and LCM as given by Equation (1.3). Figure 4.8 and 4.9 clearly show that the number of packets received is higher when the sink is placed at the center as compared to when it is not placed at the center.

In the figure 4.8 and 4.9 it is shown that before WSN gets disconnected to BS, CEC with α as 0.5 collects greater count of packets of data than other techniques. Figure shows that the PDR grows more rapidly in CEC with α set as 0.5 and sink at the center than other algorithms.

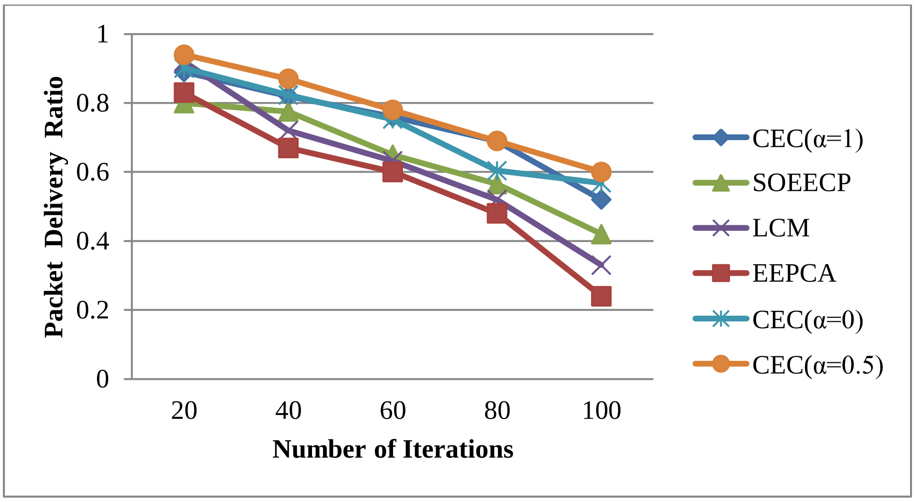


Figure 4.9 Comparative Analysis of Packet Delivery Ratio (Non-Centered Sink)

In figure 4.9 (non-centered sink), the highest value of PDR is 0.94 ($\alpha = 0.5$) while, for centered sink it is calculated as 0.97, which concludes that CEC provides better results when sink is placed at the centre. The proposed CEC technique adapts the probability of CH selection dynamically and chooses better CHs as compared to the existing techniques. Hence, it results in prolonged network lifetime and stability time period, due to which more number of packets are received in the proposed technique as compared to the existing techniques.

Test Case 4: Data aggregation Vs. Accuracy

Data accuracy means the ratio of summation of data after aggregation and the summation of the original sensed data without aggregation. The higher value of accuracy metric defines the higher quality of data aggregation technique deployed. The increment in the count of slots of time for the MNs in the cluster heads results in the greater size of the packet therefore increasing the consumption of energy in the transmission of the packet. The size of the packet is maintained by the fusion of data into packets. Higher level of data aggregation will result in less accuracy and more error at the BS. In figure 4.10 the correctness of data collected at sink is shown. The level of accuracy increases over the period of time. The graph of precision (correctness) is plotted by the comparison between the collected data at sink and the originally sent data by the MNs. The factor which contributes to the accuracy is: With longer time interval, the data messages to be sent within this duration will have less chance to collide and have a better chance of being delivered within the deadline. Hence the chance of occurring collisions is also decreased and results in more aggregation accuracy. Here, the values of CEC ($\alpha = 0.5$) are considered only, as this variant outperforms the other proposed variants in other aspects. Experimental results in Figure 4.10 depicts that CEC maintains better data precision as compared to the existing techniques LCM, EEPCA and SOEECP.

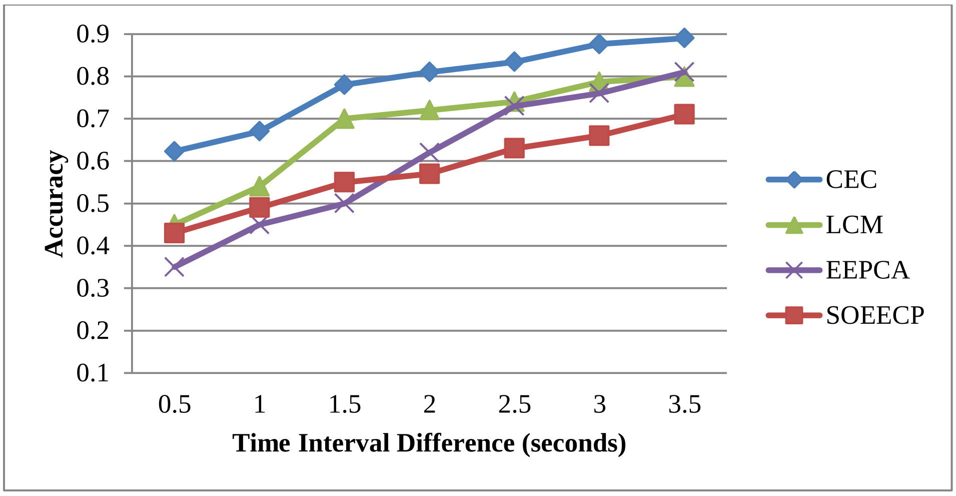


Figure 4.10 Data Accuracy Vs Time Period

Test Case 5: Statistical Analysis

Coefficient of Variation ($Coff.\ of\ Var.$) is used to analyze the statistical significance of the results. It is a statistical measure which is used to analyze the data dispersion about the mean value and for comparing the different means. It also provides an overall performance analysis of the technique being used for generating the statistics. It defines the data deviation as a proportion of its mean value and is calculated as given in Equation (4.28):

$$Coff.\ of\ Var. = \frac{SD}{M} \times 100 \tag{4.28}$$

Where SD is a standard deviation and M is mean. $Coff.\ of\ Var.$ of energy consumption has been studied of proposed data aggregation technique (CEC) and existing techniques (EEPCA, SOEECP and LCM) as shown in Figure 4.11.

$Coff.\ of\ Var.$ is calculated for energy consumption results attained by CEC, EEPCA, SOEECP and LCM. Range of $Coff.\ of\ Var.$ (0.52% - 1.55%) in terms of execution time approves the stability of CEC as shown in Figure 4.11. Small value of $Coff.\ of\ Var.$ signifies CEC is more energy efficient in data aggregation in the situations where the number of nodes are varying. The value of $Coff.\ of\ Var.$ decreases as the number of nodes is increasing. Statistical analysis illustrates that the CEC outperforms existing load balancing techniques for large numbers of nodes.

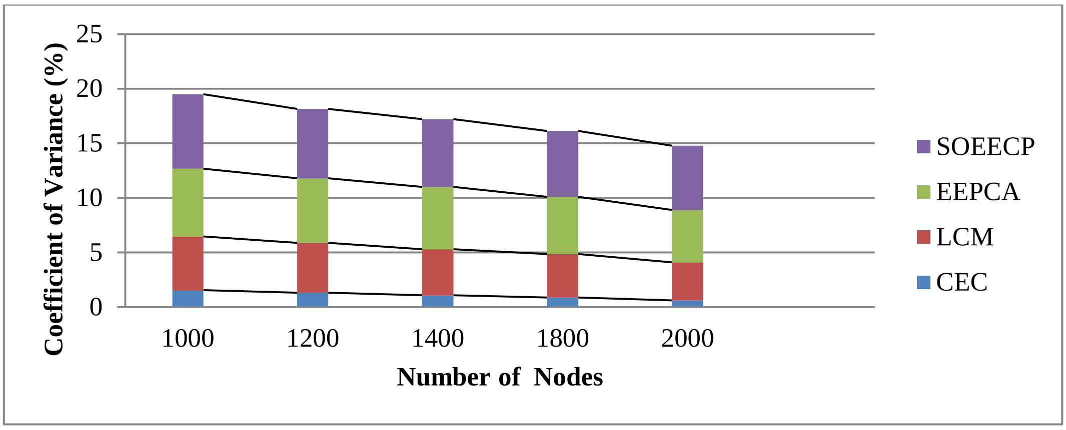


Figure 4.11 CoV for Execution Time of Various Clustering Techniques

4.5 Chapter Summary

The Cross-layer Energy Based Clustering (CEC) technique is based on dynamic data aggregation for heterogeneous networks. The whole network is divided into clusters that are hexagonal in shape. The size of the cluster is based on the side length of the cluster. Cluster head is elected by the member nodes of the cluster based on residual energy and distance. Additionally, a weight parameter (α) makes the decision for the importance to be given to the two parameters namely: probability of becoming the cluster based on remaining energy and cluster head's consumption of energy, in reaching to or from member nodes. TDMA based MAC scheme is utilized, in which time slot allocation is done by sink for the six adjacent cluster heads on the basis of the remaining energy. The member nodes go to sleep after transmitting the data and rise up only when these nodes are in listening state (cluster head to member nodes transmissions). Before sending the data to the sink or the next hop cluster head, it fuses the data on the basis of the remaining energy and appropriate level of aggregation. Simulation results show that the performance of CEC $(\alpha = 0.5)$ is better as compared to the CEC $(\alpha = 0)$ and CEC $(\alpha = 1)$ as equal importance is given to residual energy and distance for selection process of next hop. In case of CEC $(\alpha = 0)$, nodes with low energy level are not selected and in case of CEC $(\alpha = 1)$, the selection of distant nodes is avoided. The results of simulation evaluations demonstrate that CEC technique is effective in terms of energy consumption, network lifetime, number of active nodes and number of packets received as compared to existing techniques LCM, EEPCA and SOEECP with varying number of nodes and number of iterations.

Chapter 5

PSO Based Energy Efficient Load Balancing in Wireless Sensor Networks

The path for transmitting the data between the source and destination is decided based on underlying routing mechanism by taking into account the certain QoS parameters. Multipath routing is the most suitable routing mechanism for energy efficient and reliable data transmission. Load balancing also plays an important role for the maximization of network lifetime as limited energy is consumed rapidly if the whole traffic is redirected towards a single path. Due to the relationship between energy consumption and network lifetime of the sensor nodes, energy consumption in a network should be minimized and balanced in order to increase the network lifetime. To solve this problem, there is a need of energy efficient load balancing technique.

The energy dissipation for communication is always more than the energy required for computation. Hence, energy can be conserved via in-network aggregation of data using clustering, rather than sending the whole data to the Base Station by the sensor nodes individually. By utilizing the multipath routing along with the load balancing approach, the traffic or load can be distributed over the multiple paths. This approach leads to balanced energy dissipation and avoids network hole problem which is the result of the overutilization of the same path or nodes in a repetitive manner.

In this chapter, a PSO based technique has been proposed to analyse and divide the network traffic over the multiple paths so that load balancing can be performed in an energy efficient way. To provide more reliability in case of node or path failure, data is split over the multiple paths using Erasure Coding. The proposed load balancing technique works in three phases namely *cluster formation, cluster head selection* and *data transmission* between the source and the sink. The sink/cluster head aggregates the data coming from multiple paths at the destination end. The various fitness functions are defined in terms of distance and residual energy for making routing decisions.

5.1 Preliminaries for PSO-EELB

Particle Swarm Optimization (PSO) based Energy Efficient Load Balancing (PSO-EELB) technique is proposed in this chapter. The motivation of this research work is to design an energy efficient load balancing technique for efficient transmission of data which considers energy consumption as an optimization parameter. The main contributions of this research work are to:

(i) Propose an energy efficient load balancing technique, in which required number of routing paths and energy consumption of different paths is calculated,

(ii) Select different routing paths based on optimal value of residual energy and distance and further PSO based load balancing is performed for data transmission,

(iii) Optimize the network lifetime, throughput, number of alive nodes, number of data packets received, execution time and energy consumption.

Therefore, Particle Swarm Optimization (PSO) based Energy Efficient Load Balancing (PSO-EELB) technique is proposed. The preliminaries for the development of PSO-EELB has been reviewed and summarized in this section. These preliminaries include the review on:

- Network Assumptions
- Estimation of number of paths
- Erasure Coding

5.1.1 Network Assumptions

The proposed work simulates a WSN consisting of "n" number of sensor nodes deployed for monitoring applications in a rectangular field of interest. Some assumptions are made regarding the deployment of nodes as given below:

I. All the chosen nodes are considered as static after deployment.
II. Two types of nodes are deployed as follows: *Sensor Nodes* are used for monitoring the environment and another type of node is *Sink* or *Base Station* (BS) which is fixed in the center of the sensor network for collecting the data and taking the decision accordingly.
III. Sensor nodes are assigned with a distinctive identification (ID) to each node and similar preliminary energy.
IV. Node is allowed to use transmission power with different levels to communicate with the target node. The radio energy model is utilized for calculating the energy consumption for various network operations (as described in Chapter 3, Section 3.1.2).
V. The BS once in a while sends a request message in terms of the packet to the cluster head for getting sampled data from sensors.
VI. Links are assumed to be symmetrical.

5.1.2 Estimation of number of paths

A data packet is routed from a source node to the BS via number of intermediate nodes which act as forwarder nodes in single-path routing. In multi-path routing scheme, the same packet is routed via multiple paths discovered between source and destination. In single-path routing, there is a probability of failure of intermediate nodes due to which reliability of data transmission is decreased. In multipath routing, a packet is divided into n_d number of sub packets of equal size with some added amount of redundancy and is communicated over n_d disjoint paths using network coding technique. A small number of sub-packets are required, to reconstruct the original packet at the destination.

Let S_{n_d} be a random variable which represents the number of successfully data delivering paths. S_{n_d} is upper bounded by n_d, that is, $S_{n_d} \leq n_d$. The process of transmitting a data packet is considered as a Bernoulli experiment. For the i^{th} path, if the transmission process is successful, 1 is assigned to sub-run; otherwise, 0 is assigned. The value of S_{n_d} is the sum of the values assigned to the n_d sub-runs for n_d disjoint paths. Thus, the expected number of successful data delivering paths can be calculated as given by Equation (5.1):

$$E(S_{n_d}) = \sum_{i=1}^{n_d} P_i \tag{5.1}$$

where P_i is the probability of delivering a packet to destination node successfully using the path i. Let β is an upper bound for required probability of successfully reconstructing the sent message at the destination. In order to compute the value of E_{n_d} for a given β bound by a standard distribution $N\ (\mu\ , \sigma\ )$ is calculated, where the mean (μ) is given by Equation (5.2):

$$\mu = E(S_{n_d}) = \sum_{i=1}^{n_d} P_i \tag{5.2}$$

and the standard deviation (σ^2) is calculated as given by Equation (5.3):

$$\sigma^2 = \sum_{i=1}^{n_d} P_i\ (1 - P_i) \tag{5.3}$$

The degree of multipath routing n_d determines the total number of sub packets. A given pair $(n_d, \{ p_1 , \dots , p_{n_d} \})$ results in a different normal distribution $N\ (\mu\ (n_d), \sigma\ (n_d))$ each time. So, to address this issue, the random variable S_{n_d} is transformed into $S_{n_d}^* = \frac{(S_{n_d} - \mu)}{\sigma}$, which is a standard normal distribution variable, $N(0,1)$. However, the values of the bound x_β are given for any β such that $P\big(S_{n_d}^* \geq x_\beta\big) \geq \beta$ is satisfied. As a result, $S_{n_d}^* = \frac{S_{n_d} - \mu}{\sigma} \geq x_\beta$ implies that $S_{n_d} \geq x_\beta \times \sigma + \mu$ and hence probability is given by Equation (5.4):

$$P\big(S_{n_d} \geq x_\beta \times \sigma + \mu\big) \geq \beta \tag{5.4}$$

By equating this probability with $(S_{n_d} \geq E_{n_d}) \geq \beta$, an estimation of E_{n_d} can be obtained for a given bound β is given by Equation (5.5):

$$E_{n_d} = \max(\lfloor x_\beta \times \sigma + \mu \rfloor , 1) \tag{5.5}$$

By using values of Equation (5.2) and Equation (5.3), the value of E_{n_d} is given by Equation (5.6):

$$E_{n_d} = \max(\lfloor x_\beta \times \sqrt{\sum_{i=1}^{n_d} P_i\,(1 - P_i)} + \sum_{i=1}^{n_d} P_i, 1) \tag{5.6}$$

which represents an estimated number of paths successfully delivering data for a given value of β and data is sent over the multiple paths using Erasure coding.

The proposed technique creates different paths to transfer data. The different activities of sensors need to be scheduled in an efficient manner to improve residual energy. So, Time Division Multiple Access (TDMA) method is used to schedule the tasks of a subset of nodes into different groups with successive time slots.

In proposed technique, entire WSN is divided into number of different groups. Each group consists of parent nodes (Cluster Heads) and children nodes (Cluster Members). Parent nodes collect the data from the children nodes and transfer that data to BS for further processing as shown in Figure 5.1. In this network model, load (L_i) is calculated for every group (G_i) as given in Equation (5.7):

$$L_i = \sum_{[j=1]\ \varepsilon\ G_i}^{n_c} L_j \tag{5.7}$$

Here n_c represents the total number of nodes used in a group. Let L_j is the load at particular sensor j is calculated as given by Equation (5.8):

$$L_j = \frac{rS_{n_d} + L_{pc}}{P_{n_d}} \tag{5.8}$$

where rS_{n_d} represents the packets which are generated by the sensors of the group (G_i), L_{pc} represents the number of packets received from the child nodes of the group (G_i) and P_{n_d} is total number of packets transferred in one time slot using TDMA scheduling. L_{total} is total amount of load transferred in entire network as given by Equation (5.9):

$$Fitness = L_{total} = \sum_{i=1}^{n} L_i \tag{5.9}$$

The energy consumption is calculated as the amount of energy used to transfer the total load (L_{total}) as per the radio energy model. In this model, the residual energy is considered before transferring the data in assigned time slots.

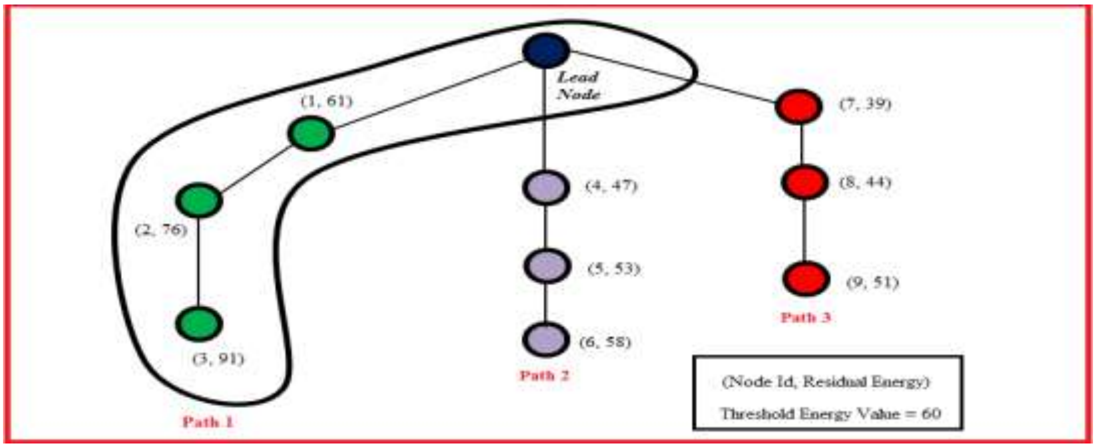


Figure 5.1 Selection of Path with Maximum Value of Residual Energy

5.1.3 Erasure Coding

Erasure code is based on FEC code in which redundancy of n_k words is added to the original n_m data words. The total number of words which are transmitted to receiver is $n_m + n_k$ words. The original data word will be recovered if any n_m combination out of $n_m + n_k$ words is received correctly. Reed Solomon (RS) based erasure coding is used as it needs less number of packets to reconstruct the original data and also reduces delay.

High-Speed Coding (HSC) is an approach which is utilized. In HSC, erasure coding technique is customized to make it more suitable for sensor nodes [268]. HSC reduces the encoding/decoding time of data without imposing any limit on the data size which can be sent. An erasure XOR-based optimal coding mechanism is utilized. The number of XOR operations needed for data encoding is significantly less in case of HSC as compared to Cauchy Reed-Solomon (CRS) technique.

In the research work,n_k, n_m and w denotes the number of redundant words, data words and size of the words, respectively. The working of conventional RS technique is as follows: It is assumed that n_m data words are represented as $D = \{D_1 .. D_{n_m}\}$ and the n_k code words are denoted by $B = \{B_1 .. B_{n_k}\}$ in the form of vectors respectively, then the redundancy code words can be calculated as given by Equation (5.10):

$$Y \times D = B \tag{5.10}$$

Here Y denotes $n_k \times n_m$ distribution matrix and its row vectors are linear independent. Hence, Equation (5.11) becomes:

$$\begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1n_m} \\ y_{21} & y_{22} & \cdots & y_{2n_m} \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ y_{n_k1} & y_{n_k2} & \cdots & y_{n_kn_m} \end{pmatrix} \times \begin{pmatrix} d_1 \\ d_2 \\ \cdots \\ \cdots \\ d_{n_m} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ \cdots \\ b_{n_k} \end{pmatrix} \tag{5.11}$$

The calculated binary code words B are encoded with the original D data words as given by Equation (5.12):

$$M \times D = Z \tag{5.12}$$

where the vector Z and matrix M are defined as given by Equation (5.13):

$$M = \binom{I}{Y}, Z = \binom{D}{B} \tag{5.13}$$

I denotes the $n_m \times n_m$ identity matrix. Matrix M comprises two sub-matrices as: the $n_m \times n_m$ identity matrix (I) in the upper part, and the $n_k \times n_m$ distribution matrix Y in the lower part, while the vector Z consists of the n_m data words D in the upper part and the n_k code words B in the lower part. Hence, it can be represented as the Equation (5.14):

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ y_{11} & y_{12} & \cdots & y_{1n_m} \\ y_{21} & y_{22} & \cdots & y_{2n_m} \\ \vdots & \vdots & \ddots & \vdots \\ y_{n_k1} & y_{n_k2} & \cdots & y_{n_kn_m} \end{pmatrix} \times \begin{pmatrix} d_1 \\ d_2 \\ \cdots \\ \cdots \\ d_{n_m} \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n_m} \\ b_1 \\ b_2 \\ \cdots \\ \cdots \\ b_{n_k} \end{pmatrix} \tag{5.14}$$

The resulting Z vector is sent to the destination. It is assumed that at the destination, only few code words are received successfully due to various reasons. The successfully delivered data/code words are mapped to the vector Z and matrix M as given in Equation (5.14) by only keeping those rows which correspond to the delivered data words and by replacing the rows of the missing data words with the rows which correspond to the delivered code words.

It results in a vector represented as Z' and a $n_m \times n_m$ matrix denoted by M'. Thus, to fetch the missing data words, the Equation (5.15) is utilized

$$M' \times D = Z' \Rightarrow D = [M']^{-1} \times Z' \tag{5.15}$$

The decoding in Reed-Solomon technique is performed by inverting the matrix M' and then multiplying with vector Z'. In Reed-Solomon technique, Vandermonde matrix Equation (5.16) is used as a distribution matrix. The elements of the matrix are defined over Galois field $GF(2^w)$, where $n_m + n_k < 2^w$. The Galois field consists of a finite number of elements and is denoted by $GF(P^w)$, where w is a positive integer number and P is a prime number. After the matrix multiplications, the resulting code words always belongs to $GF(P^w)$.

$$Y = \begin{pmatrix} 1 & a_1 & a_1^2 & \dots & a_1^{m-1} \\ 1 & a_2 & a_2^2 & \dots & a_2^{m-1} \\ 1 & a_3 & a_3^2 & \dots & a_3^{m-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_n & a_n^2 & \dots & a_n^{m-1} \end{pmatrix} \tag{5.16}$$

CRS encoding technique

The distribution matrix utilized in CRS technique is termed as Cauchy matrix $(n_m \times n_k)$. In order to define a Cauchy matrix, two sets of numbers are defined initially as given below:

$$X_{mat} = \{x_1, x_2, \dots, x_{n_1}\};\ x_i \in GF(2^w)$$

$$Z_{mat} = \{z_1, z_2, \dots, z_{n_2}\};\ z_i \in GF(2^w)$$

$$X \cap \mathrm{Z} = 0$$

Further, a Cauchy matrix U is defined in terms of its element $u[i,j]$ in the i^{th} row and j^{th} column as given by Equation (5.17):

$$u[i,j] = \frac{1}{x_i + z_i} \tag{5.17}$$

Hence, the Cauchy matrix is represented as given by Equation (5.18):

$$U = \begin{pmatrix} \frac{1}{x_1 + z_1} & \frac{1}{x_1 + z_2} & \frac{1}{x_1 + z_3} & \cdots & \frac{1}{x_1 + z_{n_m}} \\ \frac{1}{x_2 + z_1} & \frac{1}{x_2 + z_2} & \frac{1}{x_2 + z_3} & \cdots & \frac{1}{x_2 + z_{n_m}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{x_{n_k} + z_1} & \frac{1}{x_{n_k} + z_2} & \frac{1}{x_{n_k} + z_3} & \cdots & \frac{1}{x_{n_k} + z_{n_m}} \end{pmatrix} \tag{5.18}$$

In order to optimize the performance of RS coding, all operations from $GF(2^w)$ are converted into XORs in $GF(2)$. Each number $p \in GF(2^w)$ can be represented in two ways. In the first method, it can be represented as a column vector of binary representation $Y(p)$, of length w. The second representation has $w \times w$ elements of '1's and '0's as $M(p)$. The i^{th} column of $M(p)$ is equal to $Y(p * 2^i)$. The multiplication in $GF(2^w)$ is switched into multiplication in $GF(2)$, where addition and multiplication are mapped to XOR and bitwise-AND respectively. The multiplication of a vector and a matrix can be computed as given by Equation (5.19):

$$M(p1) * Y\,(p2) \;=\; Y(p1.p2) \tag{5.19}$$

In CRS, the encoding in binary form is represented by replacing each element p of the data words vector in Equation (5.10) by its binary representation $Y(p)$ and each element p of the Cauchy matrix is replaced by its binary representation matrix $M(p)$ in Equation (5.10). The size of the Cauchy matrix then is $wn_k \times wn_m$. In CRS, each data/code word is represented by w bits in its binary form. Cauchy matrix results in a map that indicates which bits in the data words should be XORed to generate each bit in the code words.

To create any bit of any code word, the data word vector is multiplied by its corresponding row of the binary representation of the Cauchy matrix and then the resulting '1's are XORed. The number of XORs required for the creation of one binary bit of code word B is equal to the number of '1's in the corresponding row of the Cauchy matrix.

High-Speed Coding (HSC) Technique

CRS technique has been adapted to make it suitable for WSNs and has been enhanced to reduce the time needed for encoding data.

In encoding, the data words are multiplied by distribution matrix Z in order to generate the code words which are used in encoding mechanism. The binary representation matrix $M(p)$ is utilized as distribution matrix. A new code word is generated offline and stored to be used later. Algorithm 5.1 shows the underlying process of HSC encoding technique.

Algorithm 5.1: Encoding process for High Speed Coding

//D is binary representation vector of Data words. B represents the binary form vector of code words.

Begin

1. Initialize Vector →0 and i →0
2. **if** ($D_i == 1$) // Check each bit of Vector D **then**

   Vector B = Vector B XOR Column i of Matrix M

   //otherwise, no XOR operation is performed
3. **if** (D_i is the last bit of the Vector D) **then**

   Vector B is generated.

   else

   $i = i + 1$ and Goto Step 2.

   end if

End

In HSC, creation the code words is based on the count of '1's in the data words. The number of XORs needed for the creation of the code words is same as number of 1's in data words. For decoding purpose, HSC implements CRS decoding technique which is more efficient than conventional RS technique in terms of speed.

5.2 Proposed PSO Based Energy Efficient Load Balancing Technique

PSO is very popular meta-heuristic approach due to its implementation simplicity as it needs tuning of only a few parameters and has a fast convergence rate as compared to the other meta-heuristic approaches. It is also very cheap in terms of computation in updating an individual, as it only needs two simple equations. To attain a quicker and efficient solution of clustering and routing problem, a meta-heuristic approach such a PSO is highly desirable.

In the proposed PSO-EELB technique, an effective and efficient *Deterministic* variant of the PSO algorithm is utilized, assuming limited computational resources. Basic PSO utilizes random coefficients to maintain the swarm dynamics variety and needs extensive numerical computations to attain statistically convergent outcomes which are too computationally expensive. Therefore, efficient deterministic approach has been utilized [269].

5.2.1 Basic formulation of PSO algorithm

The basic formulation of the PSO algorithm is presented as follows[270]:

$$v_i^{k+1} = wv_i^k + c_1r_1(X_{i,pb} - X_k^i) + c_2r_2(X_{gb} - X_k^i) \tag{5.20}$$

$$X_i^{k+1} = X_i^k + v_i^{k+1} \tag{5.21}$$

The above equations represent velocity and position of the i^{th} particle at the k^{th} iteration respectively: w is the inertia weight; c_1 and c_2 represent the social and cognitive learning rate respectively; r_1 and r_2 denote two random numbers in the range [0, 1]; $X_{i,pb}$ is the personal best position ever found by the i^{th} particle and X_{gb} is global best position ever found among all particles. The use of a constriction factor χ is necessary to ensure convergence of PSO. Accordingly, the system in Equation (5.22) and Equation (5.23) are amended as follows:

$$v_i^{k+1} = \chi[v_i^k + c_1r_1(X_{i,pb} - X_k^i) + c_2r_2(X_{gb} - X_k^i)] \tag{5.22}$$

$$X_i^{k+1} = X_i^k + v_i^{k+1}, \ \chi = \frac{2}{\left|\sqrt{2-\varphi-\sqrt{\varphi^2-4\varphi}}\right|} \text{; where } \varphi = c_1 + c_2, \ \varphi > 4 \tag{5.23}$$

Typically, when constriction method is used, φ value is set to 4.1, with $\chi = 0.729, c_1 = c_2 = 1.494$ [271].

5.2.2 Deterministic formulation (D-PSO)

In order to make the overall PSO more efficient, a deterministic algorithm is formulated by suppressing the random coefficients in Equation (5.24) and Equation (5.25), which becomes

$$v_i^{k+1} = \chi[v_i^k + c_1(X_{i,pb} - X_k^i) + c_2(X_{gb} - X_k^i)] \tag{5.24}$$

$$X_i^{k+1} = X_i^k + v_i^{k+1} \tag{5.25}$$

The complexity of the proposed PSO-EELB algorithm is $O(npX)$ for p number of paths having n number of nodes and X denotes number of iterations running on sink. The initialization of particles location and velocity is performed using a deterministic and homogeneous distribution, following the Hammersley sequence sampling [272]. PSO parameters and their values for the proposed PSO-EELB are enlisted in Table 5.1.

5.2.3 Cluster Formation in PSO-EELB approach

The cluster is formed by the BS on the basis of centralized clustering. For clustering, BS broadcasts *information collection message* to all the nodes. A node after receiving the message, starts to send its information such as node id, location (distance from the BS in X and Y position), energy loss and energy loss ratio (velocity) and current energy to BS. Then BS initiates the clustering process steps as follows:

Step 1. Conversion of problem into the PSO space in which the PSO particle has two dimensions such as particle position and velocity.

Step 2. Estimation of fitness value using *fitness function*:

The aim of the proposed fitness function for PSO based clustering is to optimize the average distance and average energy of the member nodes and distance from the current CH and headcount. The fitness value for cluster formation ($FitnessValue_{CF}$) is calculated for a particle by using Equation (5.26):

$$FitnessValue_{CF} = \alpha_1 . \frac{\sum_{i=0}^{n_c} d(current\ node, member\ i)}{n_c} + \alpha_2 . \frac{\sum_{i=0}^{n_c} E(member\ i)}{n_c} + (1 - \alpha_1 - \alpha_2) . \frac{1}{ncc} \quad (5.26)$$

where α_1 and α_2 are weighting parameters (normalized values) and n_c denotes number of members covered within the cluster and ncc denotes the number of members covered by current node.

Step 3. Generation of new particles from the initial solution:

Step 3.1. Estimation of new velocity: The current velocity of a taken particle is considered as the rate at which the particle's position is changed. Based on Equation (5.24), new velocity is calculated in Equation (5.27) as follows:

$$NewVel = \chi\, [OldVel + w_1(lbestpos - currbestpos) + w_2(gbestpos - currbestpos)] \quad (5.27)$$

where χ denotes constriction factor and w_1 and w_2 are basic PSO tuning parameters denoting social and cognitive learning rates respectively. $NewVel$ and $OldVel$ denote the new velocity and old velocity respectively. Local best position, global best position and current best position are denoted by $lbestpos$, $gbestpos$ and $currbestpos$ respectively.

Step 3.2. Based on Equation (5.24) and Equation (5.25), estimation of new position of the particle is calculated in Equation (5.28), as follows:

$$NewPosition = OldPosition + NewVel \quad (5.28)$$

Finally the new particle (new velocity and new position) arrives.

Step 4. Fitness value of the new particles is estimated by using fitness function in Step 2 with new velocity and new position.

Step 5. Fitness value of old particle and new particle is compared and the best one is selected for the next iteration.

Step 6. For each iteration, one best solution is selected as a local best solution. The particle which has optimal fitness value in the current iteration is selected as $lbest$ solution.

Step 7. The local best solution from all iterations of the particle which have optimal values among all solutions are selected as a global best solution $gbest$. The final solutions are decoded into clusters. The BS forms the cluster using PSO and broadcasts a *cluster-announcement* message to nodes which contains cluster information.

5.2.4 Cluster Head Selection in PSO-EELB approach

After clustering, each sensor node maintains "*cluster list*". It includes current cluster id, velocity, location and energy. Then, the round is initiated to perform CH selection by implementing PSO algorithm.

Step 1. The members which are covered by the current node communicate with each other to select a CH which follows the steps mentioned below.

Step2. Estimation of fitness value for CH selection ($FitnessValue_{CH}$) using fitness function is given by Equation (5.29):

$$FitnessValue_{CH} = \alpha_1 . \frac{\sum_{i=0}^{n_c} d(current\ node, member\ i)}{n_c} \gamma + \alpha_2 . \frac{\sum_{i=0}^{n_c} E(member\ i)}{n_c} \gamma + (1 - \alpha_1 - \alpha_2) . \frac{1}{ncc} \tag{5.29}$$

where $\gamma = \begin{Bmatrix} 1, if\ member\ i\ is\ covered\ by\ current\ node \\ 0, else \end{Bmatrix}$, α_1 and α_2 are weighing parameters (normalized values) and n_c denotes the number of members covered within the coverage range.

Repeat Step 3 to 7 of cluster formation for CH selection (same as discussed in Section 5.2.3). Finally, the particle which is having a global best solution, is chosen as a current CH.

5.2.5 Multi-hop Intra-cluster and Inter-cluster Data Transmission

Based on TDMA, the tasks of data collection from cluster members are scheduled by the CH and data collection from CHs is scheduled by the BS for different groups/clusters within successive time slots. CHs gather the data from the cluster members and transfer that data to

BS for further processing. As the BS already carries every node's information such as location, cluster id, cluster members, residual energy and its CH, it creates disjoint multiple paths for each node towards its destination (CH or BS) using PSO based on fitness value for data transmission ($FitnessValue_{DT}$) given by Equation (5.30):

$$FitnessValue_{DT} = \frac{d\,(S_i,S_j)^2 + d\,(S_j,S_N)^2}{\max\left(d\,(S_i,S_j)^2 + d\,(S_j,S_N)^2\right)} + (1-\omega) * \frac{\max(E\,(j)) - E(j)}{\max(E(j))}; \tag{5.30}$$

where ω is randomized tuning parameter in range [0,1] used to assign weights to the parameters considered: residual energy and distance between the source S_i and destination S_j node. Here, S_N denotes Sink node. After that the sink assigns TDMA time slot to each cluster, the CHs in turn assign TDMA time slot to each node in the cluster to send packets to it. A node is switched into sleep mode without waiting for the delivery of packets, to minimize the energy consumption as given by Equation (5.31).

$$Time\ slot\ duration\ for\ each\ node = \frac{Cluster\ TDMA\ time\ slot\ duration}{Number\ of\ cluster\ members} \tag{5.31}$$

A source node in order to sends its data to the destination follow certain rules: Let the distance limit for source and destination be d_0 as calculated in Equation (5.32). If d (actual distance between source and the destination) is less than d_0, then data can be transmitted in a single hop from the source to the destination, in the form of direct communication. If $d > d_0$, PSO based algorithm presented as Algorithm 5.2 is utilized for data transmission to minimize energy consumption and to enhance the network life cycle.

$$d_0 = \frac{\sqrt{s}}{\sqrt{n_c}} \tag{5.32}$$

s is the area of the cluster and n_c denotes the number of nodes in the cluster.

The proposed PSO-EELB algorithm for path selection is described in Algorithm 5.2.

This heuristic may lead the search into an infeasible state, as any node may be not selected. To lead the search back into its feasible state, the node must be released after data is transferred so that if it still has value of residual energy more than threshold value (considered as average residual energy), then it can move into the next iteration.

Algorithm 5.2: PSO - EELB Algorithm for Path Selection

Input: Number of nodes and their values of residual energy, node id, location (distance from the BS in X and Y position) and energy loss ratio.

Output: Selection of effective paths for load balancing.

// N_p = Population Size, R_v= Random Velocity, P_v = Particle Velocity, P_p = Particle Position, Pop = Population, $gbestpos$= Global Best Position, $lbestpos$ = Local Best Position

Begin

1. Initialize Node List and Residual Energy Value.
2. Initialize a random feasible solution and particles $P_i, \forall i, 1 \leq i \leq N_p$. /*As described in section 5.2.1 */

   // Initialization of particles and solutions
3. **for** $i = 1\ to\ N_p$ **do**

   $P_v \leftarrow R_v()$

   $P_p \leftarrow Random\_Position\ (N_p)$

   $lbestpos \leftarrow P_p$

   // Initializing the $gbest$ solution.
4. $gbest$= {$lbest_k$ |Fitness($lbest_k$) = min(Fitness($lbest_i$), i, $1 \leq i \leq N_p$}

   // Calculate the value of fitness function for each particle
5. **for** $i = 1\ to\ N_p$ **do**

   // Calculate the value of fitness function

   if Fitness ($gbestpos$) ≥ Fitness ($lbestpos$) **then**

   $gbestpos \leftarrow lbestpos$
6. **while** maximum iteration is not satisfied **do**

   for $i = 1\ toN_p$ **do**

$P_v \leftarrow$ UpdateVelocity (P_v, $gbestpos$, $lbestpos$) using Equation (5.24)

$P_p \leftarrow$ UpdatePosition (P_p, P_v) using Equation (5.25)

if $Fitness\ (P_p)\ \leq\ Fitness(lbestpos)$ **then**

$lbestpos \leftarrow P_p$

if $(lbestpos)\ \leq\ Fitness(gbestpos)$ **then**

$gbestpos \leftarrow lbestpos$

7. **Return** ($gbestpos$)
8. **while** there are un-selected nodes in the input queue **do**

for every node in the node list **do**

Get the next node from queue.

Select the node based on fitness value.

9. **Repeat** the process until the complete path is generated.
10. Store the path in a hash map; with index as key and path as value.
11. Modify N_p, P_p by removing intermediate nodes of the path selected.
12. **if** all possible paths are generated **then**

Calculate the number of successful paths using Equation (5.6),

Split the packet into n_d sub-packets using Erasure coding and send these sub- packets over multiple paths generated.

else

go to Step number 6.

End

A node list is obtained from the participating nodes by selecting only those nodes which fulfil node selection criteria. Once the node list has been obtained, a random feasible solution is initialized. The process of choosing the best heuristic from low-level heuristics is initiated. Each particle represents a node identifier with an initial solution in the solution space along with the fitness function. A low-level heuristic is selected at each particle location and its fitness function is computed i.e. Fitness (L_{BP}). If Fitness (L_{BP}) is better than Fitness (G_{BP}) then G_{BP} takes the value of L_{BP}. Fitness value of the particle at best global position is calculated next. The velocity and position of the selected particle is updated using Equation (5.24) and Equation (5.25). The fitness value is calculated for new position and is compared with its previous calculated position. If it is better than the local best value then particle's current position is assigned to the local best value.

5.3 Experimental Setup and Results

The performance evaluation of proposed PSO-EELB is done through NS2 Simulator. The various PSO parameters and their values for proposed approach are shown in Table 5.1. The parameters for experimental setup have been enlisted in Table 5.2. BS is assumed to be situated in the centre of the region.

Table 5.1 PSO Parameters

Parameter	Value
N_p	60
c_1	1.494
c_2	1.494
χ	0.729

5.3.1 Validation of Proposed Technique

To validate the proposed PSO-EELB technique, some existing approaches DESA[109], GSTEB[120], ACOLBR[114] and GACCTR[112] are selected, which also perform energy based load balancing based on other meta-heuristic methods. These techniques have been implemented in above discussed environment and the results are compared in terms of different parameters: energy consumption, throughput, convergence rate, number of data packets received, execution time, network lifetime and number of active nodes to prove the effectiveness of the proposed technique.

Table 5.2 Experimental Parameters and Values

Parameters	Values
Number of nodes	100
Area of deployment	$200m \times 200m$
Frequency	2.4GHz
Initial energy of sensor nodes	2.0J
Number of execution iterations	100
Communication range of node	60m using Lucent WaveLan DSSS radio
E_{elec}	50 nJ/bit
E_{fs}	10pJ/bit/m^2
E_{mp}	0.0013pJ/bit/m^4
Data Packet Size	4000 bits
Protocol used	802.11 MAC protocol
Traffic Source Type	Constant Bit Rate (CBR) sources
Sending Rate	1 - 4 packets per second
Evaluation time	Periodic sample time 100 sec to analyze the changes

Test Case 1: Energy Consumption Vs. Number of Nodes

The value of energy consumption is calculated for proposed PSO-EELB technique and existing techniques DESA, GSTEB, ACOLBR and GACCTR with different number of nodes (20-100). Each node consumes energy for communication and computation. So, with the increase in the number of nodes, the value of energy consumption also increases. The value of energy consumption is calculated on the basis of energy model presented (Chapter 3, Section 3.1.2) and values of parameters are shown in Table 5.2. The value of energy consumption in PSO-EELB is lesser as compared to other techniques at different number of nodes as shown in Figure 5.2. The minimum value of energy consumption is 7.52 J at 20 nodes in PSO-EELB. Average energy consumption in PSO-EELB is 6.34%, 9.721%, 10.54%, 12.64% and 16.66% lesser as compared to ACOLBR, GSTEB, GACCTR and

DESA respectively. The rationale behind this difference in performance is: In ACOLBR, Minimum Spanning Tree (MST) is used for intra-cluster routing, so each time cluster reformation takes place, there is a need for generation of MST. As the number of nodes is increasing, there is huge energy consumption for network structure formation. In GACCTR, parent selection is performed based on GA. For ensuring reliability, trust function is calculated for every path through message exchange which results in energy dissipation. Although GACCTR and GSTEB consume more or less same amount of energy, still it can be claimed that the proposed algorithm performs better in terms of energy consumption.

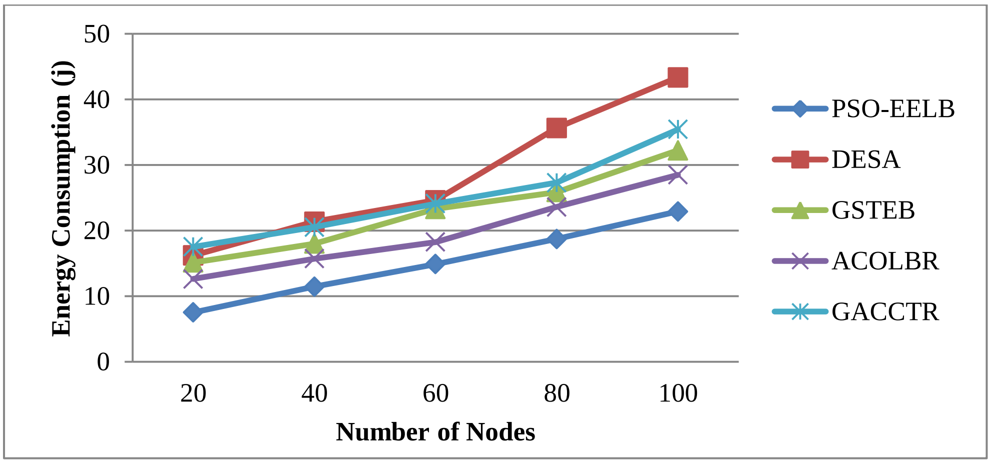


Figure 5.2 Energy Consumption Vs Number of Nodes

Test Case 2: Network Lifetime vs. Number of Iterations

The value of network lifetime has been calculated for PSO-EELB, DESA, GACCTR, GSTEB and ACOLBR with different number of iterations. With the increase in the number of iterations (20 to 100), the value of network lifetime is decreasing in case of existing techniques. The rationale behind this performance is the communication overhead involved while topology formation and decreasing residual energy. The value of network lifetime in PSO-EELB is more as compared to GACCTR, GSTEB, ACOLBR and DESA at different number of iterations as shown in Figure 5.3. The maximum value of network lifetime is 130 seconds at 20 nodes. Average network lifetime in PSO-EELB is 12.63%, 13.71%, 15.12% and 18.75% more as compared to DESA, ACOLBR, GSTEB and GACCTR respectively.

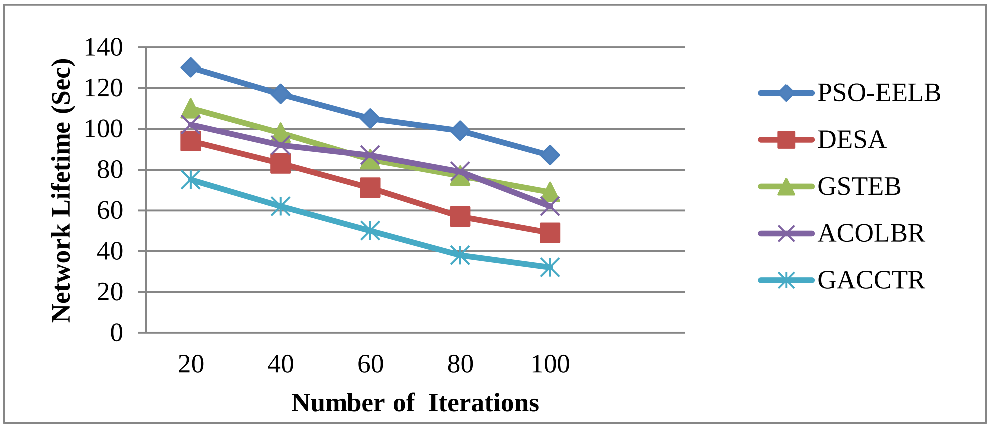


Figure 5.3 Network Lifetime Vs Number of Iterations

Test Case 3: Number of active nodes Vs Number of iterations

The value of number of active nodes has been calculated for PSO-EELB, GACCTR, GSTEB, ACOLBR and DESA with the increasing number of iterations (1 to 100). A node is termed as *active* if its current residual energy is above zero and there is at least one CH within its radius. With the increase in the number of iterations, the number of active nodes is decreasing due to energy dissipation in communication and computation operations. The value of active nodes in PSO-EELB is more as compared to GACCTR, GSTEB, ACOLBR and DESA with increasing number of iterations as shown in Figure 5.4.

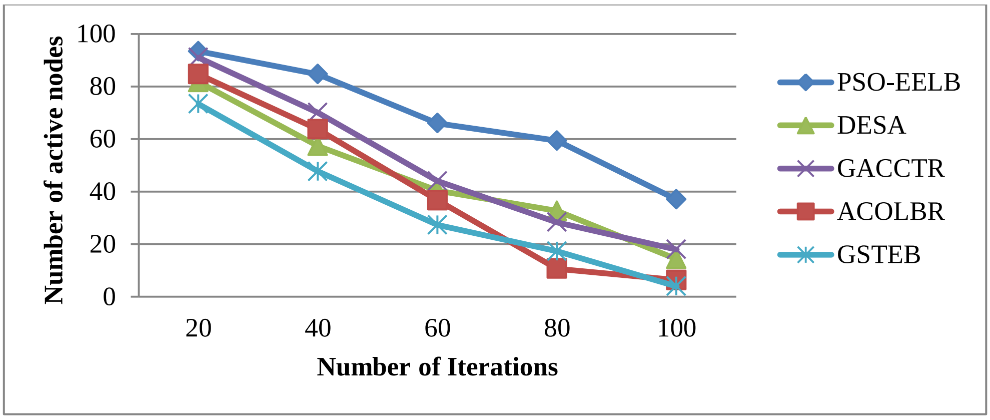


Figure 5.4 Number of Active Nodes Vs Number of Iterations

The maximum value of number of active nodes is 93 at 20 iterations. Number of active node in PSO-EELB is 15.98%, 14.22%, 12.97% and 10.16% more as compared to GACCTR, ACOLBR, DESA and GSTEB respectively.

Test Case 4: Packet Delivery Ratio Vs Number of Iterations

The value of Packet Delivery Ratio (PDR) with respect to *number of iterations* has been calculated for PSO-EELB, GACCTR, GSTEB, ACOLBR and DESA. As shown in Figure 5.5, with the increasing number of iterations, the receiving rate of data packets is decreasing due to decreasing residual energy. Initially, in PSO-EELB technique maximum number of packets are received with Packet Delivery Ratio (PDR) of 0.94 (94.01%) in 20 iterations; but existing techniques EELB, DESA and ACOLBR are receiving almost same number of data packets with PDR value of 0.89 approximately.

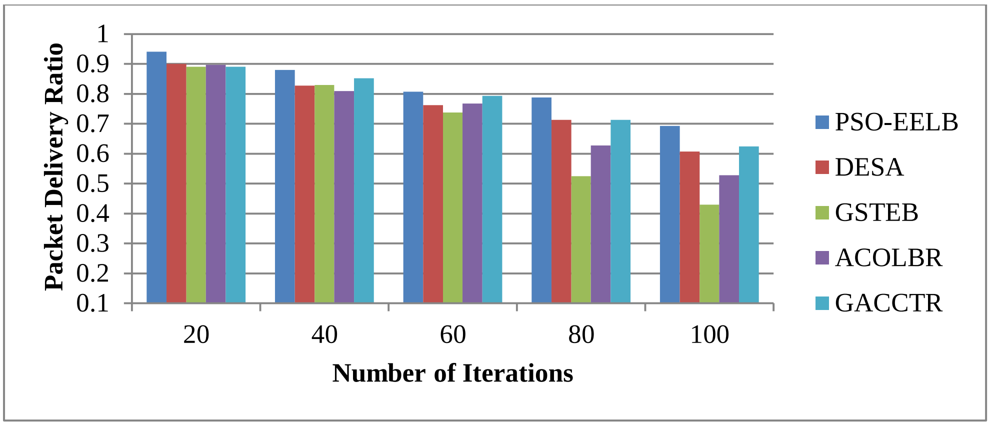


Figure 5.5 Packet Delivery Ratio Vs Number of Iterations

The minimum number of data packets is received in PSO-EELB at 100th iteration with value of PDR as 0.69. Average number data packets received in PSO-EELB are 7.18%, 20.35%, 13.14% and 6.06% more than ACOLBR, GACCTR, DESA and GSTEB respectively.

Test Case 5: Throughput

It is a ratio of total amount of data transferred successfully to the total amount of time required to transfer data. Throughput is calculated using Equation (3.26). The value of throughput has been calculated for PSO-EELB, GACCTR, GSTEB, ACOLBR and DESA with different number of nodes (20 to 100). Figure 5.6 shows the comparison of throughput of PSO-EELB, GACCTR, GSTEB, ACOLBR and DESA. It is clearly shown that PSO-EELB performs better than the selected existing techniques. The maximum value of throughput at 20 nodes i.e. PSO-EELB has 6.41%, 9.17%, 17.66% and 18.22% more throughput as compared to GSTEB, DESA, GACCTR and ACOLBR respectively. The rationale behind this performance is the higher PDR and energy efficiency of proposed PSO-

EELB as compared to the existing techniques which already have been analysed via results depicted in Figure 5.2 and Figure 5.5.

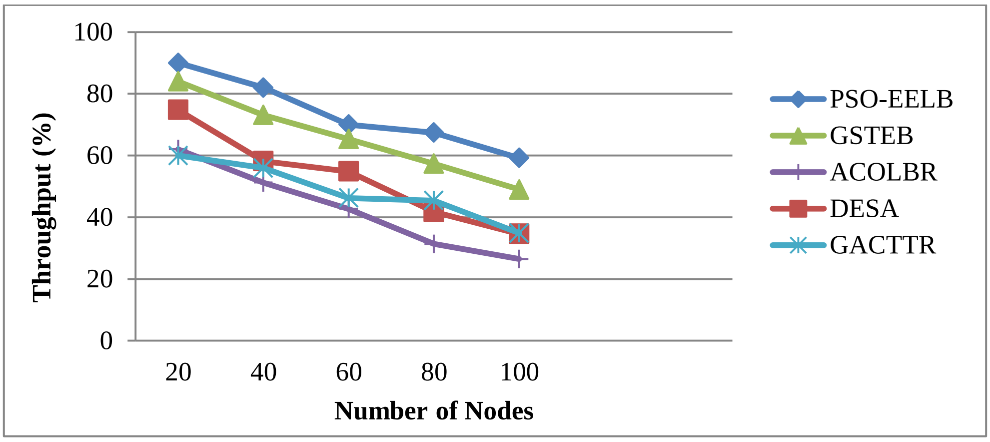


Figure 5.6 Throughput Vs. Number of Nodes

Test Case 6: Load Balancing factor

In order to depict that how well the proposed technique balances the load, the results are plotted between the standard deviation of load and varying (increasing) number of nodes. Figure 5.7 plots the standard deviation of load transmitted by GACCTR, GSTEB, DESA, ACOLBR and proposed PSO-EELB for varying number of nodes which clearly shows the better performance of proposed algorithm. Initially the load is equal and randomly initialized. The number of nodes are varying from 20-100.

Apart from these test cases, the proposed PSO-EELB algorithm also performs well in terms of algorithmic complexity as compared to GSTEB, GACCTR, DESA and ACOLBR. The overall run-time complexity of existing ACOLBR is $O(ncluster.(\log(nCluster)) + nCluster.n_c^2)$ *where* $nCluster <= n_c$ *in* every case. In this case, for intra-cluster routing operations, the complexity is $O(nCluster.n_c^2)$ and for inter-cluster routing operations is $O(ncluster.\log(nCluster))$, where n_c is number of nodes in the cluster and $ncluster$ is number of clusters.

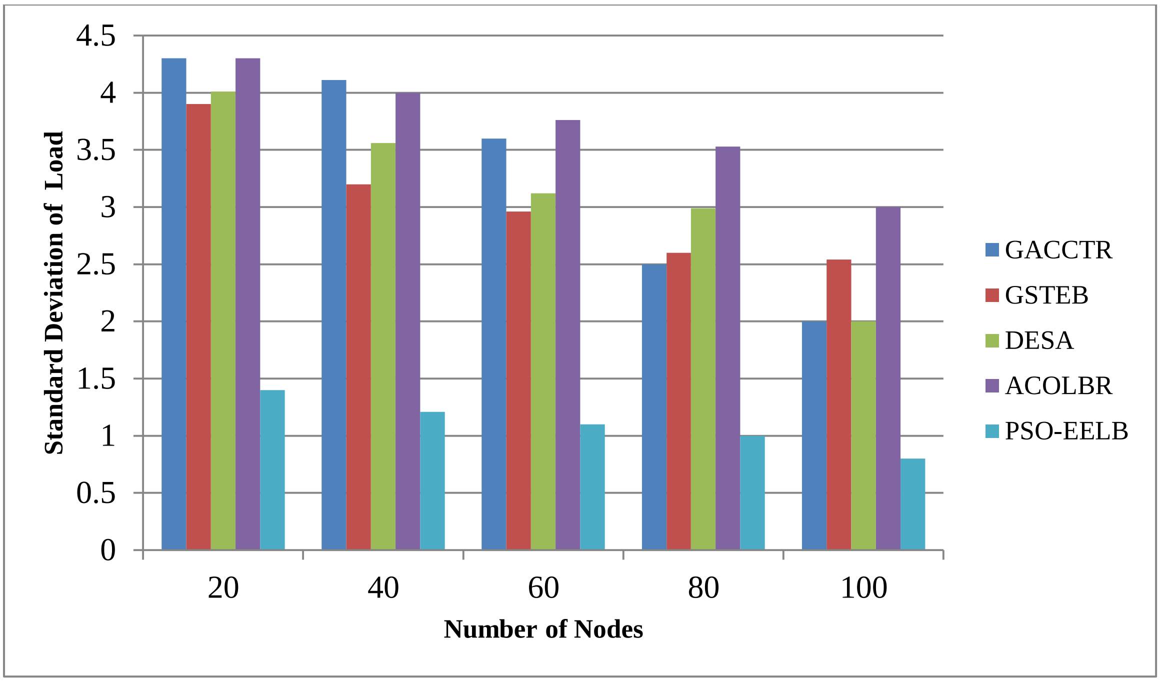


Figure 5.7 Comparison of Load Balancing for Various Techniques

In case of GACCTR, the overall complexity is $X(2n^2g + ln + l + n^2)$, g denotes number of generations in population and n represents number of nodes in grid and X is number of iterations. It comes out to be $O(n^2Xg)$. In DESA, the worst case complexity is $O(Xn^2)$, here X denotes the number of iterations of Simulated Annealing and $O(n^2)$ is the worst case complexity of Differential Evolution. In case of GSTEB, the overall complexity of algorithm is $(n_O^2ct + X.n_f.n_O.nCluster.m + n_Oct^2)$. Here, X is the number of iterations, n_f is the number of employed bees or food sources, n_O is the number of data objects in the dataset and m is the number of attributes and ct is the total number of categories for all attributes. The complexity of the proposed PSO-EELB algorithm is $O(n.X.n_p)$ for n_p number of paths having n number of nodes and X denotes number of iterations running on sink which proves that PSO-EELB performs better as compared to the other techniques.

5.3.2 Statistical Analysis

Coefficient of Variation ($Coff.\ of\ Var.$) is used to analyse the statistical significance of the results. It defines the data deviation as a proportion of its mean value and is calculated given in Equation (4.28). $Coff.\ of\ Var.$ of execution time has been studied for proposed load balancing technique (PSO-EELB) and existing techniques (GACCTR, GSTEB, ACOLBR and DESA) as shown in Figure 5.8.

The range of $Coff.\ of\ Var.$ (0.52% - 1.55%) in terms of execution time proves the stability of PSO-EELB as shown in Figure 5.8. Small value of $Cof.\ of\ Var.$ signifies PSO-EELB is more efficient in load balancing in the situations where the number of nodes are varying. Value of $Coff.\ of\ Var.$ decreases as the number of nodes is increasing. Statistical analysis illustrates that the PSO-EELB outperforms existing load balancing techniques for large number of nodes.

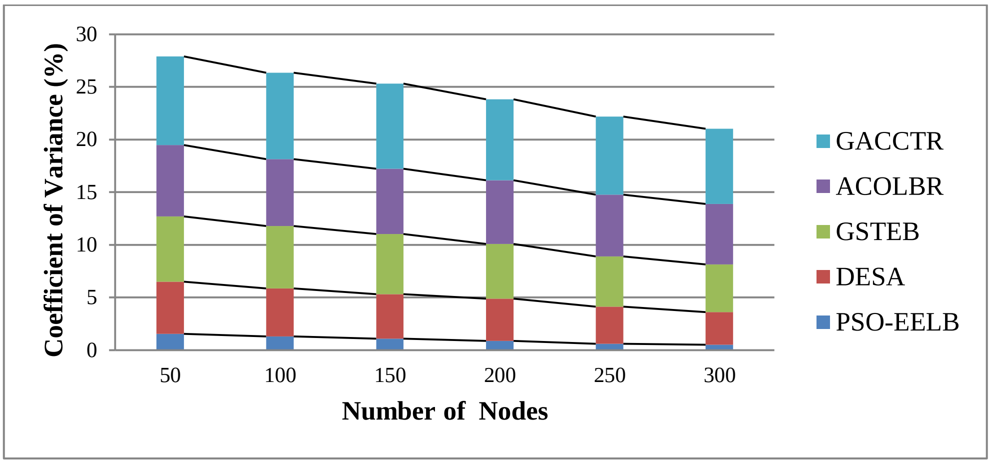


Figure 5.8 CoV for Execution Time of Various Load Balancing Techniques

5.4 Chapter Summary

A Particle Swarm Optimization (PSO) based Energy Efficient Load Balancing (PSO-EELB) technique for WSNs has been proposed. Based on deterministic PSO, multiple paths are selected and load balancing is performed by sending a packet (divided into sub-packets) using Erasure coding for data transfer at particular point of time. It is energy efficient and load balancing technique. It performs efficiently in terms of active number of sensor nodes and is more reliable as it performs routing over multiple paths (based on energy consumption) using Erasure coding along with clustering. It has optimal time complexity, i.e., $O\ (npX)$ (here n denotes number of nodes, p represents number of paths and X denotes number of iterations) in contrast to other exiting techniques. The results of simulation evaluations demonstrates that the PSO-EELB technique is effective in terms of energy consumption, throughput, network lifetime, number of active nodes, convergence rate and number of packets received as compared to existing load balancing techniques (GACCTR, GSTEB, ACOLBR and DESA) with different number of nodes and number of iterations.

Chapter 6

Multi-objective Load Balancing Clustering Technique for WSNs

The number of active nodes has a great impact on the network's energy efficiency. Minimizing the number of active CHs leads to minimization of the average energy consumed per node and in turn maximizes the network's energy efficiency. However, increasing the number of CHs and taking link quality measures into consideration resulted into more compact clusters and hence increased the PDR. Clustering protocols that ignore minimizing the number of un-clustered nodes lead to leaving those nodes unattended and hence deplete the energy quickly. A sleep scheduling mechanism should be employed to minimize the energy consumption of such nodes. Many of the prior clustering protocols assumed that the CHs can send their data to the BS directly by maximizing their transmission power. However, this solution is considered as an unrealistic assumption in many practical situations as maximizing the transmission range will result in a high level of energy consumption and will minimize the network's energy efficiency. In case of WSNs, there can be number of scenarios, in which multiple underlying objectives are conflicting in nature and there is a requirement to choose one of the trade-off solutions. Such scenarios are generally addressed by multi-objective optimization formulations. So, there is a need of an appropriate Multi-Objective Optimization technique to resolve such issues.

In this chapter, a Multi-objective Load Balancing Clustering (MLBC) approach is proposed for an energy efficient and reliable clustering technique. The proposed approach uses a variable number of CHs and its objective is to assign each network node to its respective CH. Each CH is assigned to its respective next hop. If the same path or CH is selected each time, then it will lead to load imbalance and network hole problem. In each iteration of the proposed technique, role of CH is shuffled and a path is selected based on the weight assigned in order to balance the load. The values of weights keep on changing depending upon the underlying parameters. The problem of clustering and load balancing in WSN is formulated as a multi-objective optimization problem, aiming at determining an energy efficient and reliable clustering technique. Two objective functions are considered to be achieved simultaneously for energy efficient clustering technique: minimize the average energy consumption in order to maximize the energy efficiency of the network and minimize the inter-cluster transmission cost in order to maximize the data delivery reliability. The load of data transmission is minimized and balanced by shuffling the nodes responsible according the weight assigned.

6.1 Preliminaries for MLBC technique

In this chapter, a clustering technique based on Multi-Objective Particle Swarm Optimization (MOPSO) is presented for WSNs. MOPSO is utilized to calculate the optimal cluster number in order to provide efficient solutions in terms of energy in multi-objective fashion. The evolutionary capability is utilized in MOPSO for the optimization of the count of clusters. By calculating the optimal number of clusters, packets can be routed from cluster head to sink in an efficient way. The multi-objective problem can be directly addressed for finding out the Pareto-optimal solutions in contrast to assigning a weight every parameter. Firstly the algorithm computes cluster head and after that its neighbours are calculated. All those nodes which lie in transmission range of a node become neighbours of a cluster head. A new metric i.e. weight is assigned to each neighbour node based on energy and distance for further selection process. The data aggregation is performed by CH according to the level of residual energy. Two objective functions are considered simultaneously: energy efficiency and reliability denoted by F_1 and F_2 respectively.

The preliminaries for the development of MLBC has been reviewed and summarized in this section. These preliminaries include the review on:

- Pareto-based Multi-objective Optimization
- Multi-Objective Particle Swarm Optimization

6.1.1 Pareto-based Multi-objective Optimization

A Multi-objective Optimization Problem (MOP) involves optimizing a number of objective functions which need to be maximized or minimized simultaneously and are usually conflicting in nature. A problem has a number of constraints, which must be satisfied by a solution. Due to multiple objectives in MOP, there is no unique solution capable of being described as an optimal solution. Therefore, the main objective is to find a number of optimal solutions. In multi-objective optimization, there is a multi-dimensional search space. Evolutionary Algorithms (EAs) are well suited to address problems of multi-objective optimization due to their evolving nature [273].

Assuming a minimization problem for convenience, a MOP with d_v decision variables and M number of objective functions can be expressed as follows: given an d_v-dimensional decision variable vector $x = \{x_1 \dots x_{d_v}\}$ in the solution space X find a vector x^* which yields the optimum value for a given set of M objective functions $z(x^*) = \{z_1(x^*),\dots, z_M(x^*)\}$ where $M \geq 2$. However, as the objective functions have conflicting nature, it is rare that the global optimum for all of the individual objective functions occurs simultaneously at a single point of search space. In such scenarios, the main focus is to find a set of trade-off solutions.

Pareto-dominance Principle

A feasible solution x is said to dominate another feasible solution y if and only if the following conditions are satisfied:

- Solution x is no worse than a solution y in all objectives.
- Solution x is strictly better than a solution y in at least one objective.

Formally speaking, x dominates y (denoted by ($x \succ y$)), if and only if:

$$z_i(x) \leq z_i(y), \forall i \in 1, \dots, M \tag{6.1}$$

$$z_i(x) < z_i(y), \exists i \in 1, \dots, M \tag{6.2}$$

If any of the conditions mentioned above in Equation (6.1) and Equation (6.2) is false, then solution x does not dominate the solution y. If solution x dominates solution y, then solution x is better than solution y. This concept results in a set of solutions rather than a single point of solution which are known as Pareto optimal solutions or *Pareto optimal set*. While comparing the solutions obtained with reference to defined objective functions, if not any of these solutions is more appropriate than the other solutions obtained so far, then it can be concluded that none of the obtained solutions dominate the others and are called non-dominated solutions. It cannot be concluded in such cases that which one of these solutions is better. The vector which stores the solutions present in the pareto-optimal set is termed as non-dominated vector.

Solution x^* is a Pareto optimal solution if there exists no feasible vector of decision variables $\in X$, which will decrease the value of some objective without simultaneously increasing the value of at least one other objective. There are no superior solutions to the problem than x^*, although there may be other equally good solutions. Hence, $x \in X$ is Pareto optimal if and only if, condition mentioned in Equation (6.3) is satisfied:

$$z(y) \prec z(x), \forall y \in X \tag{6.3}$$

The set of solutions that satisfy Equation (6.3) is known as the *Pareto optimal set*. These solutions are represented on a curve known as *Pareto-optimal front.* The curve formulated is shown in Figure 6.1 in which two objective functions (both are of minimization type) are considered.

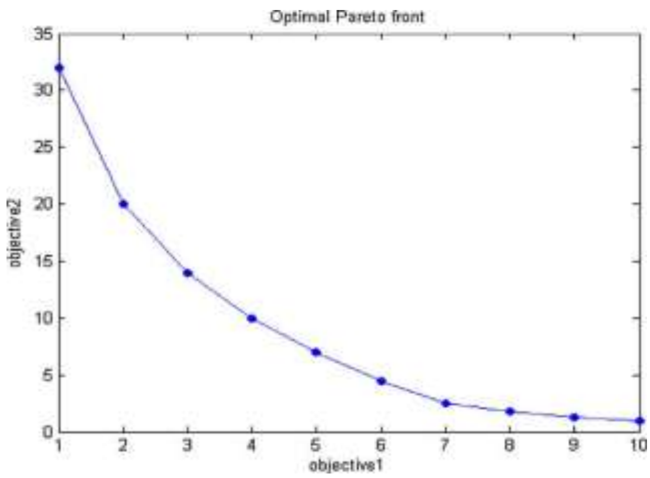


Figure 6.1 Optimal Pareto Front

There are several approaches which are utilized for addressing the MOPs, with Multi Objective Evolutionary Algorithms (MOEAs), posing all the desired characteristics for obtaining a set of non-dominated solutions, in a single run. These approaches work with two main goals:

Convergence: To find a set of Pareto-optimal solutions, and

Diversity: To find a set of diverse solutions in order to prevent premature convergence and achieve a well-distributed trade-off Pareto front.

Convergence leads the solutions towards the Pareto-optimal region and diversity guides the solutions towards the Pareto-optimal front. In multi-objective optimization, the main focus is to compute the Pareto-optimal solution set having solutions near true Pareto-optimal front and rest of these solutions are not considered as desired ones. Optimal solutions should be sundry. There are two spaces in multi-objective problems, one being decision variables space and the other objective space. Diversity can be determined in both the spaces. There is only a single search space in single-objective solution, i.e., the decision variable space. But, in case of multi-objective problems, two search spaces are involved instead of one.

In this research work, Multi-objective Particle-Swarm-Optimization (MOPSO) is considered as an optimization tool to solve the problem of clustering and load balancing in an energy efficient way in WSN. There are extensive applications of this evolutionary approach in different fields of WSNs. As per literature, this algorithm is utilized to fulfil the needs of practical optimization problems known till date and is popular because of its ease of hardware implementation. The multi-objective clustering where the nodes are grouped keeping in mind the two objectives: energy efficiency and reliability; is the main problem which needs to be addressed. As there will not be a single best solution for such scenarios, so a set of solutions will be generated in this case.

6.1.2 Multi-Objective Particle Swarm Optimization

Algorithms evolutionary in nature are highly utilized to calculate manifold solutions for problems having the nature of multi-objective optimization. These algorithms have an ability to find multiple solutions in contrast to finding piecemeal solution. Number of evolutionary algorithms is evolved based on distinct processes for generating the solutions. As for example, differential evolution, genetic algorithm, swarm intelligence and Artificial-Immune-System (AIS), etc.

In PSO, single solution which is complete in nature for a problem is known as a *particle*. A group of all such particles which hunts for an optimal solution is known as *swarm*. PSO has been adapted to address the optimization problems which are multi-objective in nature since past few years. A lot of swarms are utilized in such problems. There are number of real life applications which are dynamic in nature and in such cases, the underlying approach has to trace mobile optimum. PSO is considered as an efficient meta-heuristic approach which has been enhanced to address the multi-objective optimization problems. The number of dimensions in a vector is tantamount to the count of attributes of the problem. In the initial stage, particle velocities and positions are randomly calculated. Equations (5.21) and (5.22) of velocity and position for multi-objective problem turn out to be:

$$v_{i,d_v}^{k+1} = wv_{i,d_v}^{k} + c_1 r_1 (X_{i,d_v,pb} - X_{i,d_v}^{k}) + c_2 r_2 (X_{gb,d_v} - X_{k,d_v}^{i}) \tag{6.4}$$

$$X_{i,d_v}^{k+1} = X_{i,d_v}^{k} + v_{i,d_v}^{k+1} \tag{6.5}$$

where $d_v = 1,2,..,d_{vn}$; $i = 1,2,..,N_p$, and N_p is the size of the population and d_v denotes the number of dimensions.

A set of particles (solutions) are initialized randomly in PSO. The main objective is to reach optima by adapting the particle iteratively over the generations.

6.2 Proposed MLBC technique

This section gives a detailed description of the proposed MLBC approach which is utilized by BS to find the optimal set of CHs and the routing tree that connects them. BS adopts a Pareto-based multi-objective approach for determining the optimal set of CHs. In the proposed approach, a new individual encoding scheme that represents a novel clustering and routing problems in WSN is proposed. The inter-cluster communication involves only the CHs in a network. The proposed approach assigns each network node to its respective CH and each CH to its respective next hop, which will also be a CH. Figure 6.2 provides an overview of the proposed approach workflow.

6.2.1 Individual Initialization

The individuals are presented in such a way that each individual provides the set of CHs and the path from each CH towards BS. The dimension of an individual is assumed to be equivalent to the number of nodes deployed in the network (i.e., n). Let, $I_i = [R_{i,1}, R_{i,2}, R_{i,3} \ldots \ldots . . R_{i,n}]$ be the i^{th} individual of the population where $R_{i,d_v}, 1 \leq d_v \leq n$ maps the assignment of the sensor node N_{d_v} to a CH. Each individual is initialized arbitrarily based on a uniform distribution in the range (0,1).

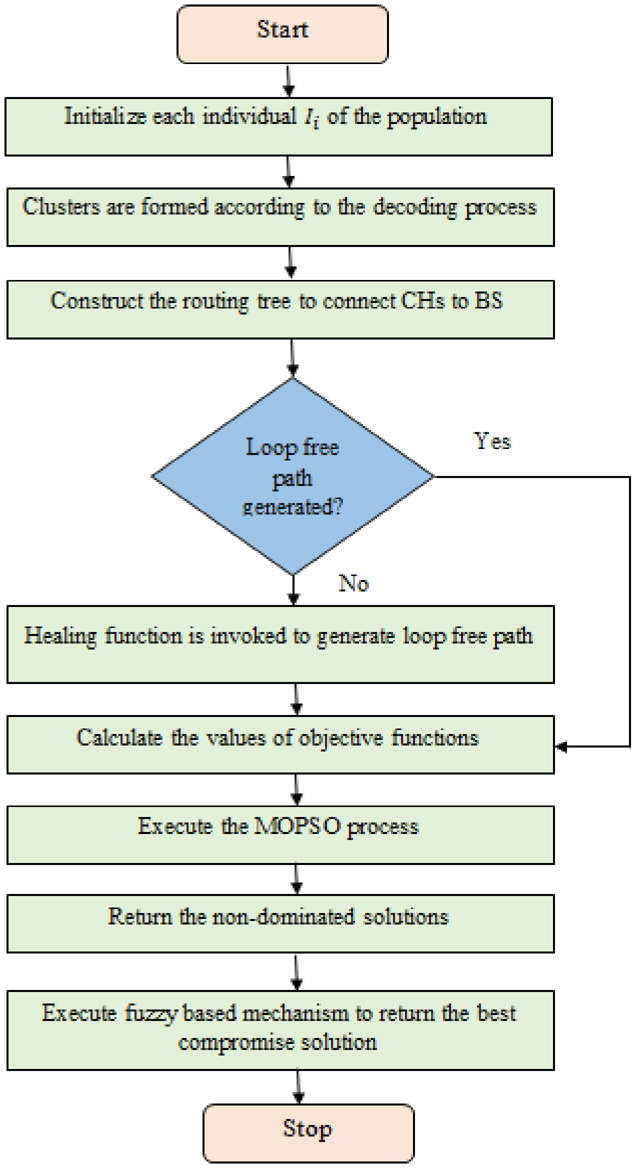


Figure 6.2 MOPSO based Proposed Technique

Let $NBR(N_{d_v})$ be the list of all N_{d_v} neighbours. Then, the CH of node K is encoded for initial clustering phase as given by Equation (6.6):

$$CH_K = \lceil (R_{i,d_v} \times |NBR(N_{d_v})|) \rceil \tag{6.6}$$

Consider a WSN with 18 sensor nodes, i.e., $S = \{N_0, N_1, \ldots, N_{17}\}$ where N_0 is the BS as shown in Figure 6.3. Therefore, the number of dimensions of an individual is equal to the number of nodes – 1 (BS), i.e., $n = 17$. The edge $u \rightarrow v$ indicates that node v is within communication range of node u hence node u can send data to node v but not necessarily vice versa.

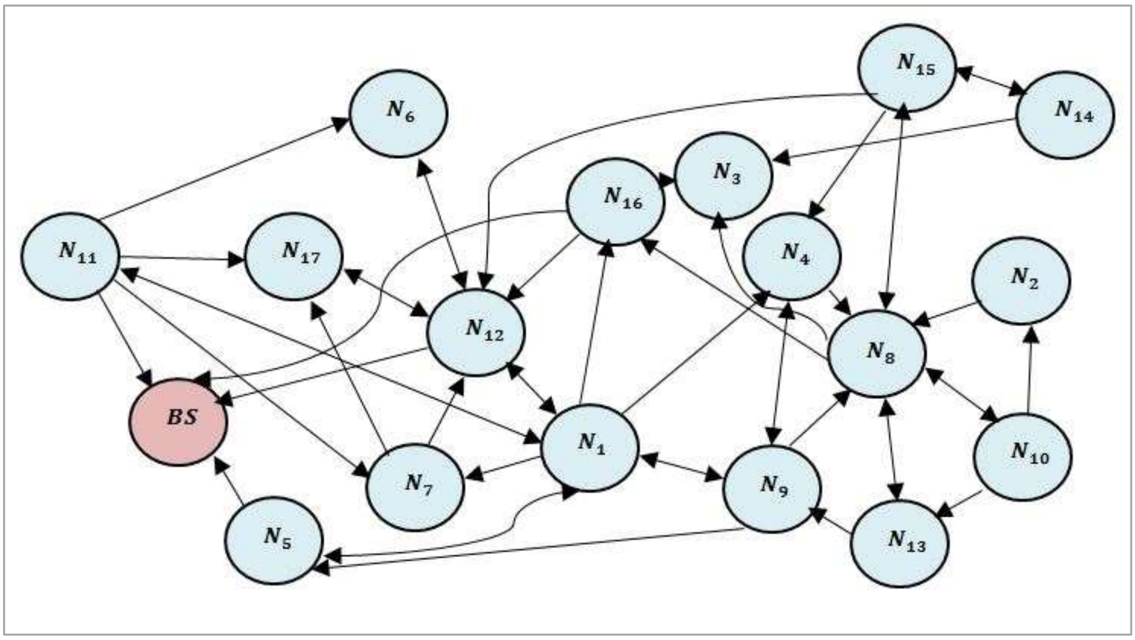


Figure 6.3 WSN Deployed with 18 Nodes

Now, for each $X_{i,d}$, $1 \leq d_v \leq 17$ of individual I_i, a random number is generated to initialize it. For instance, an individual I_i = [1.00, 0.97, 0.02, 0.34, 0.74, 0.67, 0.61, 0.74, 0.11, 0.29, 0.92, 0.33, 0.18, 0.60, 0.46, 0.47, 0.24] has been randomly generated as shown in the second column (i.e., X_{i,d_v}) of Table 6.1. This individual actually represents a candidate solution to both the clustering and routing problems as follows:

For instance, consider the generated random number for the first individual, 1.00, i.e., $X_{i,1}$ = 1.00 as shown in first column of Table 6.1. Hence, $\lceil (R_{i,d_v} \times |NBR(N_1)|) \rceil$ = 7, therefore the

7th neighbour from $NBR(N_1)$, i.e., N_{12} is selected as a CH for N_1 as shown in Table 6.1. In the same way, each sensor node is assigned to a CH using the randomly generated particle. Then, the CH candidates which result from decoding I_i is $V_i = \{N_{12}, N_{15}, N_8, N_0\}$. Table 6.1 summarizes the decoding process for individual I_i. The final assignment of each node to its next hop for individual I_i and the corresponding generated clusters are given in Table 6.1 and Figure 6.4 respectively.

Table 6.1 Individual Decoding Process to Assign a CH to Each Node

N_i	$NBR(N_i)$	$\lvert NBR(N_i)\rvert$	X_{i,d_v}	$\lceil(R_{i,d_v} \times \lvert NBR(N_i)\rvert)\rceil$	CH elected
N_1	$\{N_9, N_7, N_{16}, N_4, N_{11}, N_5, N_{12}\}$	7	1.00	7	N_{12}
N_2	$\{N_8\}$	1	0.97	1	N_8
N_3	{Null}	0	0.02	0	None
N_4	$\{N_8, N_9\}$	2	0.34	1	N_8
N_5	$\{N_1, N_0\}$	2	0.74	2	N_0
N_6	$\{N_{12}\}$	1	0.67	1	N_{12}
N_7	$\{N_5, N_{12}, N_{17}\}$	3	0.61	2	N_{12}
N_8	$\{N_3, N_2, N_{10}, N_{16}, N_{15}, N_{13}\}$	6	0.74	5	N_{15}
N_9	$\{N_8, N_1, N_4, N_5\}$	4	0.11	1	N_8
N_{10}	$\{N_8, N_2, N_{13}\}$	3	0.29	1	N_8
N_{11}	$\{N_6, N_{17}, N_7, N_0\}$	4	0.92	4	N_0
N_{12}	$\{N_1, N_0, N_{17}, N_6\}$	4	0.33	2	N_0
N_{13}	$\{N_8, N_9\}$	2	0.18	1	N_8
N_{14}	$\{N_3, N_{15}\}$	2	0.60	2	N_{15}
N_{15}	$\{N_{14}, N_8, N_4, N_{12}\}$	4	0.46	2	N_8
N_{16}	$\{N_3, N_{12}, N_0\}$	3	0.47	2	N_{12}
N_{17}	$\{N_{12}\}$	1	0.24	1	N_{12}

After cluster formation, the task of a CH is to aggregate the data received from the cluster members and report that data to the next hop. In the proposed MLBC technique, the level of aggregation varies according the residual energy of CH. If the residual energy is higher, then level of aggregation will be decreased as more data can be captured to provide more accuracy or vice versa. The cost of aggregation and level of data aggregation are expressed by Equation (4.29).

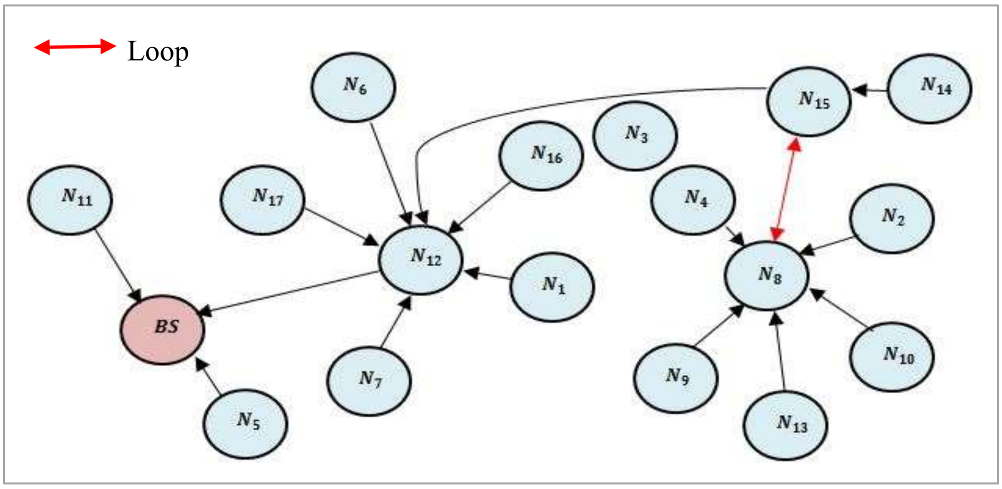


Figure 6.4 Generated Clusters for the Corresponding CHs

After the initial clustering, in the next iteration, each cluster head will be selected on the basis of the weight assigned. Two input parameters are considered for weight calculation: *initial energy* and *distance* as given by Equation (6.7):

$$w_i = \frac{1}{dis_i} \times e_i \tag{6.7}$$

Here e_i denotes the energy of i^{th} node and dis_i denotes the distance from current node to node i.

Then, in the next iterations, neighbor list of each node will be updated in the descending order on the basis of weight assigned. The weights are assigned to only those nodes which are having energy $e_i > e_{thresh}$ (average residual energy). If there are multiple nodes having similar weight (maximum weight), then selection will be performed randomly.

Let us consider the case of node N_8. The neighbours of $N_8 = \{N_3, N_2, N_{10}, N_{16}, N_{15}, N_{13}\}$ and number of neighbor nodes > 1. After first phase, the weight calculation is performed as shown in Table 6.2. N_{13} is having highest value of weight, so it will be selected as CH. In the next iteration, the weight of N_{13} can be 0.1821, which may not the highest weight in a network and then some other CH will be selected. Similarly, is the process for next hop selection. Hence, the roles of CHs and next hop are shuffled and results in load balancing. The path selection is done on the basis of the weight parameter as it is calculated in Equation (6.7).

Table 6.2 Weights Assigned to Neighbours of Node N_8

Neighbour Nodes	e_i	dis_i	w_i
N_2	1.31	14	0.0935
N_{10}	1.34	13	0.1030
N_{16}	1.96	11	0.1781
N_{13}	1.56	7	0.2228
N_{15}	1.1	8	0.1375
N_3	1.87	9	0.2077

A source node (CH) in order to sends its data to the destination (sink) follow certain rules: Let the distance limit for source and destination be d_0 as calculated in Equation (5.32). If dis_i (actual distance between source and the destination) is less than d_0, then data can be transmitted in a single hop from the source to the destination, in the form of direct communication. If $dis_i > d_0$, then the whole routing process presented above is utilized for data transmission to minimize energy consumption and to enhance the network life cycle.

The inter-cluster communication is used to transmit the data from the CHs to the BS. In the proposed approach, a multi-hop model is considered where the CHs form a network among these nodes, using a multi-hop route for data transmission towards the BS. The proposed individual encoding scheme is also utilized in the routing tree construction by assigning each CH to its next hop. However, the constructed routing tree is considered as invalid if any of the following conditions is violated:

•Loop-free routing tree.

•Each route from each CH should terminate at the BS.

Otherwise, the constructed routing tree is valid and can be used for the inter-cluster communication.

Let's consider the final nodes assignments in Figure 6.4 and the generated CHs set, $V_i = \{N_{12}, N_{15}, N_8, N_0\}$. It can be observed that CH N_{12} can send to the BS node directly. However, there is loop between CH N_8 and CH N_{15} as it is shown by red coloured edge in Figure 6.4. Hence the constructed routing tree is considered as invalid and is assigned a negative value to remove this solution for further consideration. Now, let's suppose that the generated random number for N_{15} is 0.96, i.e. $X_{i,15}$ = 0.96 in Table 6.1, then node N_{15} will be assigned to node N_{12} instead of N_8. In this case, the final nodes assignment to their next hop is given in Figure 6.5 in which there is no loop in the network. For the CH set $V_i = \{N_{12}, N_{15}, N_8, N_0\}$, Table 6.3 will result in the following routes from each CH, $N_{12} \rightarrow N_0$, $N_{15} \rightarrow N_{12} \rightarrow N_0$ and $N_8 \rightarrow N_{15} \rightarrow N_{12} \rightarrow N_0$.

Table 6.3 Assignment of Sensor nodes to their Respective Next Hop

Source Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Destination Node	12	8	-1	8	0	12	12	15	8	8	0	0	8	15	12	12	12

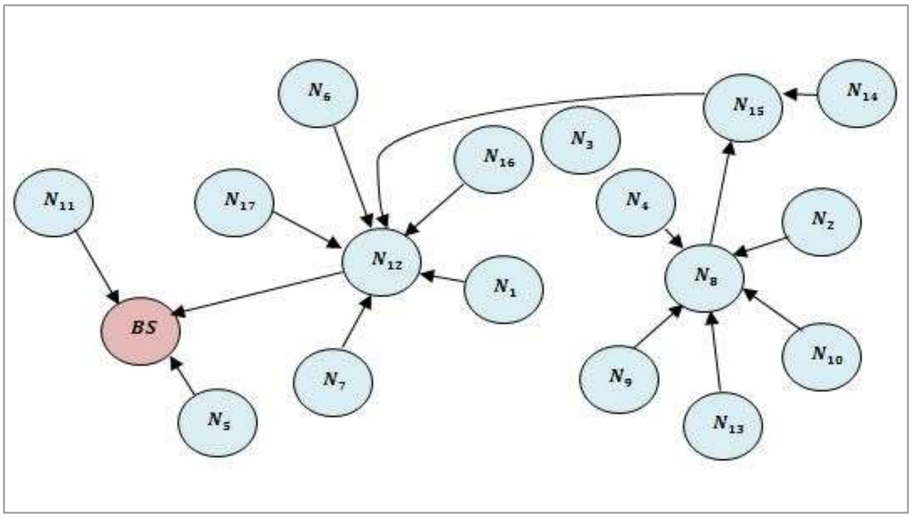


Figure 6.5 Assignment of Nodes to their Respective Next Hop Neighbour

This routing tree is considered valid since there is no loop among the CHs and each route from each CH terminates at the BS. The corresponding routing tree is illustrated in Figure

6.5. Throughout different experimentation, it is found that, regardless of the network density, the proposed approach results in a large number of non-valid routing trees due to existing loops. Hence, there is a need for a healing function to repair the constructed routing tree.

6.2.2 Healing Function for Loop Prevention

In order to construct a loop-free routing tree, the Dijkstra algorithm[274] is used to find the Shortest Path Tree (SPT) that connects the CHs to the BS. The network is presented as a weighted directed graph, $G = (V, E)$ where V represents the set of CHs in addition to the BS and E represents the set of edges between them. It is assumed that the best link quality between two nodes represents the shortest path. An edge e from node u to v has weight w_{uv}, as given by Equation (6.8):

$$w_e = w_{uv} = \begin{cases} PDR, & if\ v\ is\ neighbour\ of\ u \\ 0, & if\ u = v \\ NV, & otherwise \end{cases} \tag{6.8}$$

Here, Packet Delivery Ratio(PDR) is defined as $PDR = \frac{Number\ of\ packet\ received}{Total\ number\ of\ packets\ sent}$, which is a measure of link quality and its value lies in [0 1]. Here, NV is defined as a very high negative value. The link quality for the route from CH to the BS is calculated as given by Equation (6.9):

$$LQ_{CH\to BS} = \begin{cases} \sum_{e \in R} w_e, if\ BS\ is\ reachable\ from\ CH \\ NV, \quad otherwise \end{cases} \tag{6.9}$$

It should be noted that the link quality from the sensor u to sensor v is different from the link quality from the sensor v to sensor u. Therefore, at iteration t, the BS generates a dynamic Adjacency Matrix D_t as follows:

$$D_t = \begin{bmatrix} 0 & w_{12} & \dots & w_{1n} \\ w_{21} & 0 & \dots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \dots & 0 \end{bmatrix} \tag{6.10}$$

The BS uses the adjacency matrix D_i to find SPT that connects the BS and all the CH candidates. The Dijkstra algorithm is used to find the SPT that connects the CHs to the BS as shown in Algorithm 6.1.

Algorithm 6.1: Dijkstra algorithm to find the SPT

Input: The directed graph $G = (V, E)$ and the positive edge lengths $\{w_e: e \in E\}$ given by D_t. // Q: Set of unvisited vertices

Output: SPT and its associated cost

// For each $CH \in V$, $lq[CH]$ is the link quality for the route from CH to the BS as calculated in Equation (6.9). The SPT cost is calculated using Equation (6.13).

Begin

1. $lq[BS] = 0$
2. $prev[BS] = null$
3. **for each** $CH \in V$ **do**
4.   **if** $CH! = BS$ **then**
        $lq[CH] = \infty$
        $prev[CH] = null$
5.   **end**
6. add CH to Q
7. **end**
8. **while** Q is not empty **do**
9. u ← vertex in Q with maximum $PDR(u)$ value
10.   remove u from Q
11. **for each** neighbour v of u **do**
12. $alt \leftarrow lq[u] + w_{v \to u}$
13.   **if** $alt < lq[v]$ **then**
14. $lq[v] \leftarrow alt$
15. $prev[v] \leftarrow u$
16.   **end**
17. **end**
18. **end while**
19. **return** $prev[], \sum_{CH \in V} lq[CH]$

End

6.2.3 Individual Evaluations

The next step after initializing the individuals is to evaluate the generated clusters and the constructed routing tree according to some objective functions. This step helps to update the set of Pareto-optimal solutions and the Pareto front periodically. The main goal of the technique is to find the optimal set of CHs such that the following objectives are achieved concurrently:

- Minimize the average consumed energy in order to maximize the network lifetime.

- Minimize the inter-cluster transmission cost in order to maximize the data delivery reliability.

The objective functions are constructed to evaluate each candidate solution depending on the decision parameters described as follows:

Average Energy Consumed: In order to save more energy, fewer sensor nodes need to be active during each round. The main approach is to minimize the number of elected CHs denoted by K. Let vector V_i denotes the vector that represents the CHs generated from decoding individual I_i, after removing duplicate values. Then, the number of elected CHs is given by:

$$K_{I_i} = |V_i| \tag{6.11}$$

Furthermore, a sensor node with a higher level of energy is a better CH candidate to both aggregate the data and to act as a relay node towards another CH or BS. The energy efficiency can be increased in terms of CHs if the average CH residual energy can be maximized or average CH energy consumption can be reduced. The objective function F_1, is chosen as the reciprocal of the average remaining energy for the CH candidates and is given by Equation (6.12):

$$Min\ (F_1) = EE_{I_i} = \frac{|V_i|}{\sum_{i=0}^{|V_i|} E(CH_{I_i,k})} \tag{6.12}$$

$E(CH_{I_i,k})$ is the remaining energy of CH number k generated from decoding individual I_i.

Data Reliability: In order to increase the network throughput or to increase the data delivery reliability, there is a need to minimize the inter-cluster communication cost. The cost of the link between any two nodes is determined by link weights in the Adjacency Matrix D_t as given by Equation (6.9) and Equation (6.10).

The total cost of the constructed tree (inter-cluster communication cost), is defined as the sum of the costs of links between the CHs forming that tree. If any two CHs are not connected, the constructed tree is assigned a high penalty value to narrow down the search to optimal valid tree solutions only. Therefore, the total cost of the constructed tree is calculated as follows:

$$Min\ (F_2) = TC_i = \begin{cases} \sum_{k=1}^{K} \sum_{e=1}^{E} w_e, if\ all\ nodes\ in\ V\ are\ connected \\ N, otherwise \end{cases} \tag{6.13}$$

where K denotes number of CH candidates. E represents the number of edges in path number k. w_e is the weight of edge e given by Equation (6.8). The main objective of the proposed MLBC technique is to simultaneously minimize EE_i and TC_i for individual I_i. The process of MOPSO algorithm is described in Algorithm 6.2.

Algorithm 6.2: MOPSO algorithm

// Repository REP stores the CHs vectors representing non-dominated vectors.

// N_p represents population.

Begin

1. Initialize each node's position randomly.
2. Initialize each node's velocity and parameters required for MOPSO.
3. **for** each particle X **do**

    (a) **while** each node in the network is not visited **do**

    i. Choose a CH X_i randomly.

    ii. Store the neighbours of the CH.

(b) Exclude CH 'i' and its neighbours for next selection process of CH.

(c) **end while**

5. **end for**
6. Evaluate each particle of N_p.
7. Store the best CHs vectors.
8. Generate the Pareto front by utilizing non-dominated sorting.
9. Save vectors of CHs that showcase vectors non-dominated in nature in REP.
10. Calculate the global CHs vector from the REP.
11. **while** $MaxIter$ is not satisfied **do**

    (a) Calculate the velocity (VEL) of each CH vector.

    (b) Calculate the new CHs vectors of the solutions adding the velocity calculated in the last step.

    (c) Calculate each particle in N_p.

    (d) The contents of REP are updated.

    (e) When the current combination of CHs of the solution is better than the combination of CHs contained in its memory, the position of particle is updated.

    (f) $MaxIter = MaxIter + 1$.

12. **end while**

End

For each solution the objective functions are calculated using their respective equations. Non-domination sorting is used to find out the optimal Pareto front which results in global best CHs vector. The velocity is calculated for each individual based on personal best positions, current positions of the CHs of the current individual and the positions of global best CHs vector positions. The current velocity and position of each CH in the current vector are used for new CHs vector. The execution of this process will be terminated after the maximum number of iterations is reached.

6.2.4 Determining the Best Compromise Individual

After obtaining a set of Pareto optimal solutions provided by Multi- objective evolutionary approaches, a mechanism is needed to determine the best compromise solution. Due to the imprecision in the decision of the decision maker, it is assumed that there is fuzziness in the goal for each objective. This fuzziness is defined by membership functions that represent the degree of fuzziness of some fuzzy sets using values in the range [0, 1].

The fuzzy mechanism analyses the contribution of the solutions for the fulfilment of the objective and a fuzzy variable is assigned accordingly. It helps in finding a compromise solution if the solutions are very close to each other. A fuzzy based mechanism [275] is used to find out a compromise solution on the Pareto front. This mechanism has been successfully used in many different applications of MOEAs.

In the fuzzy-based mechanism, a membership value for i^{th} objective of j^{th} solution is calculated in the Pareto-front using the membership function as given by Equation (6.14):

$$u_j^i = \left\{ \begin{array}{c} 1 if\ F_i \le F_i^{\min} \\ \dfrac{F_i^{\max} - F_i}{F_i^{\max} - F_i^{\min}} if\ F_{\mathrm{i}}^{\min} < F_i < F_{\mathrm{i}}^{\max} \\ 0 if\ F_i \ge F_i^{\max} \end{array} \right\} \tag{6.14}$$

u_j^i indicates how well the j^{th} solution in the Pareto optimal set can satisfy the i^{th} objective. The sum of membership values for all objectives of the j^{th} solution suggests how well it satisfies all the objectives. Given S solutions in the Pareto-optimal set and M objective functions for each solution, the achievement of each non-dominated solution with respect to all non-dominated solutions can be calculated using Equation (6.15):

$$u^j = \frac{\sum_{i=1}^{M} u_j^i}{\sum_{j=1}^{N} \sum_{i=1}^{M} u_j^i} \tag{6.15}$$

The solution with the maximum value of u_j is a compromise solution which is required by the decision maker.

6.3 Experiment Setup and Results

The performance evaluation of the proposed MLBC technique is done on $100\,m \times 100\,m$ grid with static nodes. The transmission range of each node varies from 10 to 60 m. The number of nodes is varying between 20 and 100. The proposed MLBC technique is compared with existing IMOWCA[245], MOPSO-DE[251] and JPSO[249] for parameters: packet delivery ratio, total residual energy, network lifetime, number of active nodes, coverage and scalability. The results are obtained after performing 100 iterations of each technique. A radio energy model is utilized for the calculation of energy loss for communication discussed in section (3.1.2). Table 6.4 enlists the experimental settings and its values. The various MOPSO parameters along their values are shown in Table 6.5 for proposed MLBC approach.

Table 6.4 Simulation Parameters

Simulation Parameters	Values
Area Covered	$100 \times 100\,m$
Deployment Mode	Randomly deployed
Location of sink	Center of network field
Number of nodes	100
Packet Length (l)	2000 bits
$E_{initial}$	2 J
E_{elec}	50 nJ/bit
E_{mp}	100pJ/bit/m^2
E_{fs}	10pJ/bit/m^2

Table 6.5 MOPSO Parameters

Parameters	Values
c_1	2
c_2	2
Inertia Weight w	0.694
Population size	100
Number of iterations	100

Figure 6.6 shows the Pareto optimal front for the two objective functions defined Energy consumption (F_1) and Reliability (F_2) as given by Equation (6.12) and Equation (6.13). The values of these objective functions are normalized between (0,1) as shown in Figure 6.6.

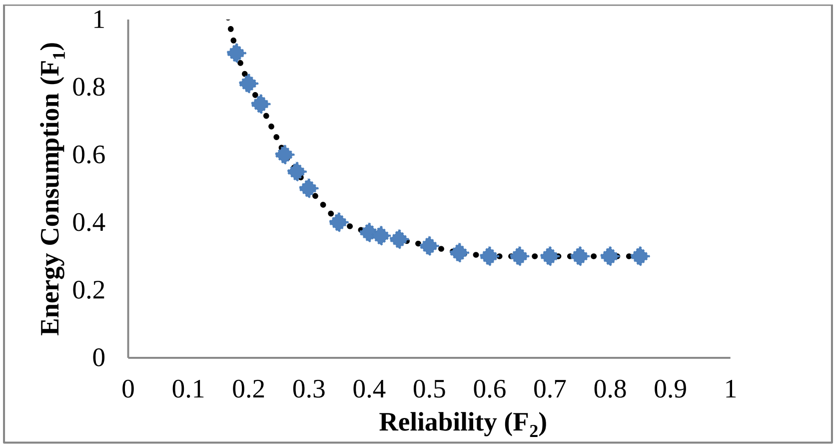


Figure 6.6 Pareto Optimal Front of Proposed MLBC

Test Case 1: Residual Energy vs. Number of Iterations

The value of residual energy has been calculated for proposed MLBC and existing JPSO, MOPSO-DE and IMOWCA techniques with increasing number of iterations. The value of total residual energy is decreasing, as the number of iterations increases (20 to 100). The reason behind this performance is incurred overhead in terms of communication during formation of topology.

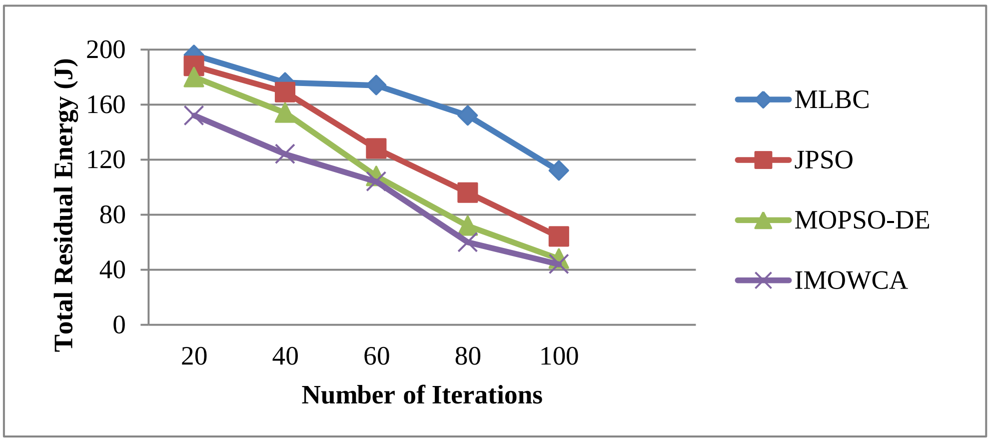


Figure 6.7 Residual Energy Vs Number of Iterations

The value of total residual energy in MLBC is more as compared to IMOWCA, MOPSO-DE and JPSO for varying number of iterations as depicted in Figure 6.7. The average residual energy in MLBC is 13.13%, 16.21% and 19.07% more as compared to JPSO, MOPSO-DE and IMOWCA respectively.

Test Case 2: Number of active nodes Vs Number of iterations

The number of active nodes values has been evaluated for MLBC, IMOWCA, MOPSO-DE and JPSO with the increasing number of iterations (1 to 100). A node is considered as active node if its present residual energy is more than zero and there must be at least one CH within its range. As the number of iterations is increasing, the number of active nodes decreases due to energy dissipation. The value of number of active nodes is maximum for 93 at 20 iterations. Number of active node in MLBC is 16.28%, 11.79% and 9.13% more as compared to IMOWCA, JPSO and MOPSO-DE respectively as shown in Figure 6.8.

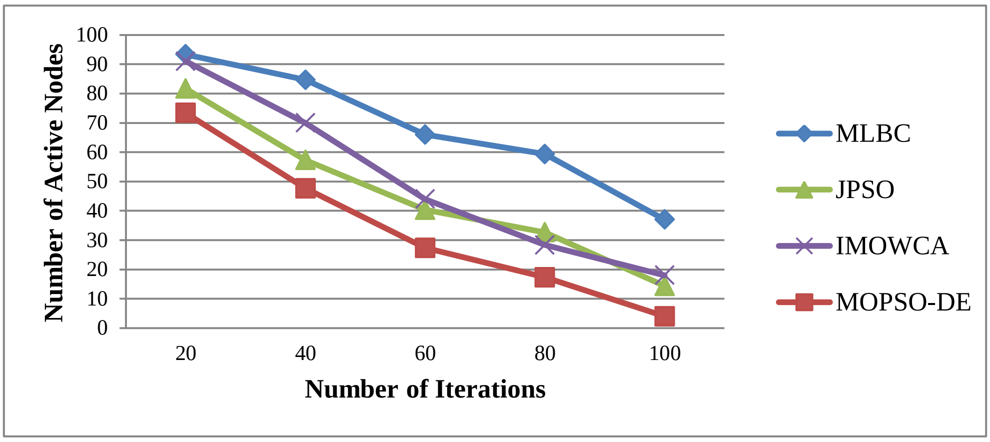


Figure 6.8 Number of Active Nodes Vs Number of Iterations

Test Case 3: Packet Delivery Ratio Vs Number of Rounds

The value of Packet Delivery Ratio (PDR) is calculated with respect to number of rounds for proposed MLBC technique and existing IMOWCA, MOPSO-DE and JPSO techniques. As shown in Figure 6.9, data packets receiving rate is decreasing, with the increase in number of iterations. Initially, maximum value of PDR is 0.885 after 20 iterations in case of MLBC, but IMOWCA (0.712) and MOPSO-DE (0.72) are receiving approximate similar amount of data packets. The maximum value of PDR for JPSO is 0.775 at 20th iteration. At 20th round, MLBC receives 17.21%, 18.13% and 18.26% more data packets than IMOWCA, MOPSO-DE and JPSO respectively. The minimum number of data packets received in MLBC is at 100th iteration with PDR=0.625. The average number of data packets received in MLBC is 12.76%, 13.91% and 20.95% more than IMOWCA, JPSO and MOPSO-DE respectively.

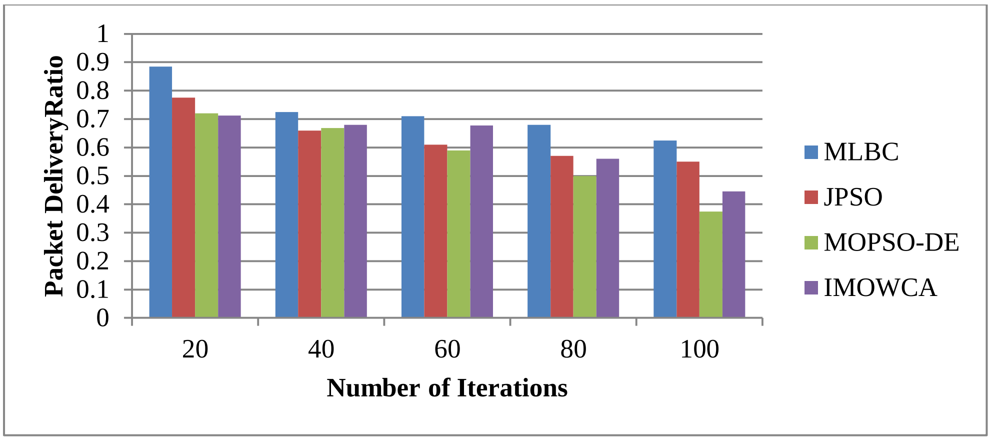


Figure 6.9 Packet Delivery Ratio Vs Number of Iterations

Test Case 4: Network Lifetime vs. Number of Iterations

The value of network lifetime has been calculated for proposed MLBC and existing IMOWCA, MOPSO-DE and JPSO with different number of iterations. With the increase in the number of iterations (20 to 100), the value of network lifetime is decreasing. The rationale behind this performance is the communication overhead involved while topology formation. The value of network lifetime in MLBC is more as compared to existing techniques IMOWCA, MOPSO and JPSO for varying number of iterations as shown in Figure 6.10. The maximum value of network lifetime is 130 seconds at 20th iteration. The average network lifetime in MLBC is 11.93%, 12.91% and 18.57% more as compared to JPSO, MOPSO-DE and IMOWCA respectively.

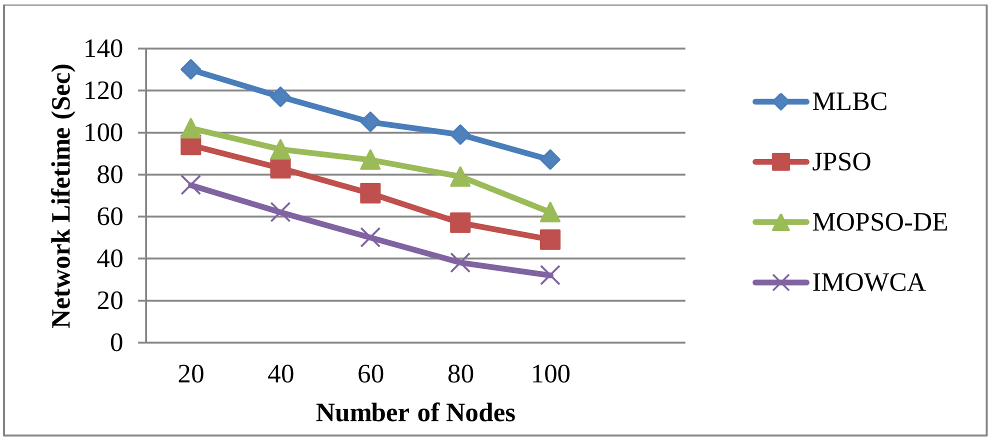


Figure 6.10 Network Lifetime Vs Number of Iterations

Apart from the above discussed parameters, the main objective of all the MOEAs is to find solutions close to the Pareto optimal front as much as possible and solutions should be diverse as much as possible in the non-dominated front. Since the true Pareto-optimal front for the proposed application is unknown, for performance analysis, coverage and spacing of the Pareto front are considered. Coverage measures the convergence of the Pareto front and the distribution of solutions along the Pareto front is measured by spacing.

Test Case 5: Coverage of the Pareto front

Let A and B are assumed to be Pareto-optimal sets. The relative spread of solutions between two non-dominated sets is measured by this metric. The function Cov maps the ordered pair (A, B) to the interval $[0, 1]$ and is given by Equation (6.16):

$$Cov(A, B) = \frac{|\{b \in B | \exists\, a \in A: a \succeq b\}|}{|B|} \tag{6.16}$$

where $|B|$ denotes the number of solutions in set B and $a \succcurlyeq b$ represents that solution b is weakly dominated by solution a. The value $Cov(A, B) = 1$ implies that A weakly dominates all decision vectors in B. In contrary, $Cov(A, B) = 0$ implies that none of the points in B are weakly dominated by A. If $Cov(A, B) > Cov(B, A)$ then the set A has better solutions than the set B.

Test Case 6: Spacing

Spacing metric measures the distribution of the solutions over the non-dominated front. Spacing between solutions is computed as:

$$Spacing = \sqrt{\frac{1}{NS-1}\sum_{i=1}^{NS}\left(\frac{d_i}{d}\right)^2} \tag{6.17}$$

where, $d_i = min_j \sum_{m=1}^{M} \left|F_m^i - F_m^j\right| \; for\ j = 1 \ldots .NS\ and\ i \neq j$ and NS denotes the number of solutions in the non-dominated set and the number of objective functions is represented by M and d is the mean of all the d_i. The nearer the value of Spacing to zero, the more uniformly distributed the solutions found over the Pareto optimal front.

Table 6.6 Results of Cov Metric for MLBC and IMOWCA

Parameters	**Clustering Techniques**	
	MLBC	**IMOWCA**
$Best$	1	0.137
$Worst$	0.92	0
$Average$	<u>0.9474</u>	0.0625
$Median$	0.962	0.0986
SD	0.0039	0.0669

Table 6.7 Results of Cov Metric for MLBC and MOPSO-DE

Parameters	**Clustering Techniques**	
	MLBC	**MOPSO-DE**
$Best$	0.9461	0.3631
$Worst$	0.7129	0.0248
$Average$	<u>0.9011</u>	0.2247
$Median$	0.8172	0.0182
SD	0.0056	0.0098

Table 6.8 Results of Cov Metric for MLBC and JPSO

Parameters	**Clustering Techniques**	
	MLBC	**JPSO**
$Best$	0.9886	0.3126
$Worst$	0.7835	0.0192
$Average$	<u>0.9133</u>	0.2013
$Median$	0.8361	0.0133
SD	0.0067	0.0182

Table 6.9 Results of Spacing Metric for Proposed and Existing Techniques

Parameters	Clustering Techniques			
	MLBC	IMOWCA	MOPSO-DE	JPSO
$Best$	0.2182	0.3915	0.5432	0.1254
$Worst$	0.3876	0.6891	0.5182	0.7312
$Average$	0.3113	0.5169	0.4538	0.5421
$Median$	0.3253	0.4997	0.5122	0.3214
SD	0.0687	0.1144	0.1212	0.1867

The quality of the Pareto-optimal solutions acquired with the proposed MLBC and existing techniques is measured by two qualitative metrics Cov and S. The best average results for each technique with respect to each metric are shown as underlined in all the tables. The best, average, worst, Standard Deviation (SD) and median of the performance metrics are listed in Tables 6.6 – 6.9.

The value $Cov = 0.9474$ indicates that 94.74% of the Pareto-optimal solutions acquired with MLBC weakly dominates the solutions obtained with IMOWCA as shown in Table 6.6. Similarly, the value $Cov = 0.0625$ shows that only 6.25% of the solutions obtained with MLBC are weakly dominated by the solutions obtained by IMOWCA. Further, the SD of MLBC with respect to Cov is significantly lower than existing techniques, indicates that the performance of MLBC is more stable statistically. The distributions of the solutions extracted with existing IMOWCA, MOPSO-DE and JPSO and proposed MLBC technique are calculated with respect to non-uniformity metric S. A lower value of S indicates a uniform or better spread of solutions. The results obtained for metric S shows that the spread of the solutions obtained with MLBC have more uniformity as compared to the existing techniques.

6.4 Chapter Summary

A Multi-Objective Load Balancing Clustering (MLBC) has been proposed, in which MOPSO algorithm is utilized for providing optimal solutions. In the underlying mechanism, all the nodes transmit the data regarding location and energy to the CHs. Two objective functions are defined based on energy efficiency and reliability in terms of data delivery. The load balancing is performed by shuffling the roles of next hop node and CH in each iteration. The whole process is executed at BS in order to save energy. Furthermore, a performance comparison between the proposed approach and other well-known existing multi-objective clustering techniques is conducted. The performance evaluation shows that the proposed MLBC technique is better than existing techniques in terms of active number of nodes, packet delivery ratio, network lifetime and energy consumption.

Chapter 7

Conclusions and Future Directions

7.1 Conclusions

In WSNs, it is essential to use a routing protocol that is energy efficient, scalable and robust in terms of reliable packet delivery. In this thesis, set of data aggregation and load balancing multipath techniques have been developed to address the energy efficiency, data delivery reliability and load balancing issues in WSNs. Data aggregation and load balancing in multipath routing are two well-known problems, which have been addressed in this research work. The main contributions of this research work are:

- A novel data aggregation technique, DAHDA has been presented. This technique provides required level of accuracy along with the data aggregation.
- CEC data aggregation technique has been proposed for heterogeneous networks based on hexagonal clustering model. An efficient MAC protocol has been utilized for better energy efficiency.
- A novel load balancing technique, PSO-EELB has been presented. Discrete PSO is utilized for clustering based multipath routing technique.
- The multi-objective load balancing (MLBC) has been proposed to address the multiple objectives simultaneously. MOPSO is utilized to consider reliability and energy efficiency simultaneously.

The key findings of this research work in reference to Chapters are discussed below:

The thesis commences with the introductory section of Wireless Sensor Networks in **Chapter 1** that provides an overview of Wireless Sensor Networks, architecture of sensor node and protocol stack of WSNs. The various approaches and parameters of data aggregation and load balancing have been discussed. In **Chapter 2,** the emphasis is on the multipath routing techniques. The various techniques of load balancing and data aggregation have been identified, discussed and compared and further research issues in data aggregation and load balancing have been identified.

In **Chapter 3**, a novel Dynamic Adaptive Hierarchical Data Aggregation (DAHDA) has been proposed for Clustered WSNs that utilises the LEACH-C algorithm. The proposed DAHDA has been modified as Extended DAHDA (EDAHDA) and Modified EDAHDA on the basis of level of underlying functionalities. The EDAHDA has been developed assuming that not all sensors readings are important in every round as the readings might be similar for nearby sensors. To achieve this key feature, sensors weights are designed to decide which sensors should be assigned as cluster heads and which should be assigned as sending sensors. Modified EDAHDA has been presented to handle sudden changes in the underlying data, which can happen according to the occurrence of events in the sensed field. In this, the sensors in the event area are more likely to be chosen to transmit their data to CHs.

The NS2 based simulated environment has been used to evaluate the performance. Simulation results show that EDAHDA extends the network lifetime and performs well when compared with existing techniques in both evolving as well as non-uniform WSNs while maintaining the accuracy of the data. The algorithm performs well in terms of residual energy, number of alive nodes, data accuracy, sudden burst detection, sensor distribution, lifetime of last node, first node and average node life for uniform, non-uniform and evolving networks. The results also demonstrate the adaptivity feature of the algorithm by being able to capture the events of interest.

The EDAHDA extends the lifetime of first node, last node and average lifetime compared to LEACH for both non-uniform and evolving networks. The performance of the Modified EDAHDA with adaptivity feature has been evaluated by capturing sudden bursts or events in the underlying values of the data. The higher detection ratio is obtained for the Modified EDAHDA than the detection ratio resulted with EDAHDA for the duration of the different events, which demonstrate that the events are captured more accurately in Modified EDAHDA.

In **Chapter 4**, Cross-layer Energy Based Clustering (CEC) technique based on dynamic data aggregation has been proposed for the extension of the network lifetime and minimization of energy consumption in heterogeneous networks. The level of data aggregation varies according to the energy level. In case of low residual energy, level of data aggregation is increased to provide the energy efficient data transmission with acceptable level of accuracy. Additionally, a weight parameter (α) is used to make the decision for the importance to be given to the two parameters namely: probability of becoming the cluster and cluster head's consumption of energy. The $\alpha = 0.5$ signifies that decision is based on equal importance for both residual energy and distance; $\alpha = 0$ signifies the decision based on accounting residual energy; while $\alpha = 1$, signifies the consideration of distance only in decision making. The simulation results demonstrate that CEC technique is effective in terms of energy consumption, network lifetime, number of active nodes and packet delivery ratio. The effectiveness of the presented technique is compared to existing techniques (LCM, EEPCA and SOEECP) with varying number of nodes and number of iterations. Experimental results show that the CEC $(\alpha = 0.5)$ outperforms the other two variants CEC $(\alpha = 1)$ and CEC $(\alpha = 0)$, as equal weightage is assigned to residual energy and distance while making decision. It is observed that CEC with $\alpha = 0$ makes least consumption of energy when compared to other techniques and variants.

While considering the sink in center, the network's lifetime ratio is computed for LCM, EEPCA, SOEECP and different values of α in CEC. The CEC performs better with respect to lifetime of the network when α is set as 0. The CEC with α as 0.5 collects greater count of packets of data than other protocols before the BS gets disconnected from WSN. Experimental results depicts that CEC maintains better data precision as compared to the existing techniques LCM, EEPCA and SOEECP. Further, Statistical analysis demonstrates the CEC outperforms existing techniques for large numbers of nodes.

In **Chapter 5**, a Particle Swarm Optimization (PSO) based Energy Efficient Load Balancing (PSO-EELB) technique for WSNs has been proposed. In proposed technique, number of routing paths is identified and energy consumption of different nodes and paths is calculated. The fitness functions are defined in terms of residual energy, node density and distance. The results of simulation evaluations demonstrate that the PSO-EELB technique is effective in terms of energy consumption, throughput, network lifetime, number of active nodes, convergence rate, execution time and number of packets received as compared to existing load balancing techniques (GACCTR, GSTEB, ACOLBR and DESA) with different number of nodes and number of iterations. For a hundred node network, the average energy consumption in PSO-EELB is 6.34%, 9.721%, 10.54%, 12.64% and 16.66% lower as compared to ACOLBR, GSTEB, GACCTR and DESA respectively. The average network lifetime in PSO-EELB is 12.63%, 13.71%, 15.12% and 18.75% higher as compared to DESA, ACOLBR, GSTEB and GACCTR respectively. With the increase in the number of iterations, the number of active nodes is decreasing due to energy dissipation in communication and computation operations. The minimum number of data packets is received in PSO-EELB at 100th iteration with value of PDR as 0.69. It has optimal time complexity, i.e., O (npX) (here n denotes number of nodes, p represents number of paths and X denotes number of iterations) in contrast to other exiting techniques. Further, Statistical analysis demonstrates the PSO-EELB outperforms existing load balancing techniques for large numbers of nodes.

In **Chapter 6**, Multi-objective Load Balancing Clustering (MLBC) technique has been proposed. The evolutionary approaches can be applied successfully to multi-objective optimization problems. Multi-Objective Particle Swarm Optimization (MOPSO), which is an evolutionary optimization approach is utilized for considering multiple objectives at a time. The objective functions are evaluated and MOPSO algorithm is executed. After, the completion of the execution, a set of non-dominating solutions is obtained called Pareto set. The main aim of the proposed technique is to balance the load via clustering technique by addressing multiple objectives at a time.

The clustering problem in a WSN consists of multiple conflicting objectives. Furthermore, it can be viewed as a problem that is divided into two sub-problems: finding the optimal set of CHs and finding the inter-cluster communication tree that connects these CHs to the BS. Pareto-optimization approaches can be adapted to find a joint solution to both the clustering and routing problems in WSNs. Two objective functions are considered simultaneously: Energy Efficiency and Reliability. Energy efficiency is measured in terms of residual energy of cluster head and reliability is measured in terms of packet delivery ratio.

The experimental results on 100 node random network show that the proposed MLBC technique outperforms the existing techniques such as JPSO, MOPSO-DE and MOOCTC in terms of residual energy, packet delivery ratio, network lifetime and number of active nodes. The average network lifetime in MLBC is 11.93%, 12.91% and 18.57% more as compared to JPSO, MOPSO-DE and MOOCTC respectively. Initially, maximum value of PDR is 0.885

after 20 iterations in case of MLBC. The existing techniques have PDR values as 0.712, 0.72, and 0.775 for MOOCTC MOPSO-DE and JPSO respectively at 20th iteration. Number of active node in MLBC is found to be 16.28%, 11.79% and 9.13% more as compared to MOOCTC, JPSO and MOPSO-DE respectively. MLBC receives 17.21%, 18.13% and 18.26% more data packets than MOOCTC, MOPSO-DE and JPSO respectively. Experimental results have showed that using a dedicated routing tree results in higher network throughput and hence enhance the network's data delivery reliability. Moreover, limiting the inter-cluster communication to the CHs results in fewer active nodes and this minimizes the average consumed energy per node and hence enhances the network's energy efficiency.

7.2 Future Directions

The contribution of this thesis has lead to new research areas in WSNs, which are required to be addressed through further research. The following future research directions have been suggested for the research community:

- Dynamic Adaptive Hierarchical Data Aggregation (DAHDA) technique can be further improved by optimizing the number of operations in cluster formation phase. Binary Differential Evolution (BDE) is utilized for cluster formation along with density based clustering, which can be further optimized in order to reduce the energy consumption in computation tasks.
- Cross-layer Energy-based Clustering (CEC) technique can be enhanced by analysing the type of traffic in the underlying traffic. Based on the traffic analysis, multiple aggregation schemes and MAC protocols can be utilized to make the proposed scheme more adaptive and versatile.
- PSO based Energy Efficient Load Balancing (PSO-EELB) can be extended by utilizing the concept of weight based data aggregation scheme using multiple sinks for further reducing the energy consumption.
- Multi-objective Load Balancing Clustering (MLBC) can be extended by considering additional objective functions to add more versatility to the approach. Detecting the faulty nodes and replacing such nodes by mobile backup nodes can incorporate fault tolerance.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, “Wireless sensor networks: a survey,” *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[2] K. Gupta and V. Sikka, “Design Issues and Challenges in Wireless Sensor Networks,” *International Journal of Computer Applications,* vol. 112, no. 4, pp. 26–52, 2015.

[3] I. F. Akyildiz and I. H. Kasimoglu, “Wireless sensor and actor networks: Research Challenges,” *Ad Hoc Networks,* vol. 2, no. 4, pp. 351–367, 2004.

[4] T. Arampatzis, J. Lygeros, and S. Manesis, “A Survey of Applications of Wireless Sensors and Wireless Sensor Networks,” *IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control*, Barcelona, Spain, pp. 719–724, 2005.

[5] M. Dong, K. Ota, H. Li, S. Du, H. Zhu, and S. Guo, “RENDEZVOUS: towards fast event detecting in wireless sensor and actor networks,” *Computing,* vol. 96, no. 10, pp. 995–1010, 2013.

[6] C. Y. Chong and S. P. Kumar, “Sensor networks: Evolution, opportunities, and challenges,” *Proceedings of the IEEE,* vol. 91, no. 8, pp. 1247–1256, 2003.

[7] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, “Wireless sensor networks: A survey on recent developments and potential synergies,” *Journal of Supercomputing,* vol. 68, no. 1, pp. 1–48, 2014.

[8] M. P. Durisic, Z. Tafa, G. Dimic, and V. Milutinovic, “A survey of military applications of wireless sensor networks,” *Mediterranean Conference on Embedded Computing (MECO)*, Montenegro, pp. 196–199, 2012.

[9] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, “Wireless Sensor Networks for Habitat Monitoring,” *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications,* New York, USA, pp. 88–97, 2002.

[10] M. Oliveira and J. J. Rodrigues, “Wireless Sensor Networks: a Survey on Environmental Monitoring,” *Journal of Communications,* vol. 6, no. 2, pp. 143–151, 2011.

[11] A. Darwish and A. E. Hassanien, “Wearable and implantable wireless sensor network solutions for healthcare monitoring,” *Sensors,* vol. 11, no. 6, pp. 5561–5595, 2011.

[12] A. Milenković, C. Otto, and E. Jovanov, “Wireless sensor networks for personal health monitoring: Issues and an implementation,” *Computer Communications*, vol. 29, no. 13–14, pp. 2521–2533, 2006.

[13] D.-M. Han and J.-H. Lim, “Smart home energy management system using IEEE 802.15.4 and ZigBee,” *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1403–1410, 2010.

[14] B. P. L. Lau, M. S. Hasala, V. S. Kadaba, B. Thirunavukarasu, C. Yuen, B. Yuen, R. Nayak,"Extracting point of interest and classifying environment for low sampling crowd sensing smartphone sensor data" *IEEE International Conference on Pervasive Computing and Communication Workshops* , Kona, Hawaii, pp. 1-6, 13-17 March 2017.

[15] V. C. Gungor, B. Lu, and G. P. Hancke, “Opportunities and Challenges of Wireless Sensor Networks in Smart Grid,” *IEEE Transactions on Industrial Electronics*, vol. 57, no. 10, pp. 3557–3564, 2010.

[16] M. Erol-Kantarci and H. T. Mouftah, “Wireless Sensor Networks for Cost-Efficient Residential Energy Management in the Smart Grid,” *IEEE Transactions on Smart Grid*, vol. 2, no. 2, pp. 314–325, 2011.

[17] A. D. Wood, J. A. Stankovic, G. Virone, L. Selavo, Z. He, Q. Cao, T. Doan, Y. Wu, L. Fang, and R. Stoleru, “Context-aware wireless sensor networks for assisted living and residential monitoring,” *IEEE Network*, vol. 22, no. 4, pp. 26–33, 2008.

[18] A. Wheeler, “Commercial applications of wireless sensor networks using ZigBee,” *IEEE Communications Magazine*, vol. 45, no. 4, pp. 70–77, 2007.

[19] N. Al-Karaki and A. E. Kamal, “Routing Techniques in Wireless Sensor Networks,” *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6-28, 2012.

[20] L. J. G. Villalba, A. L. S. Orozco, A. T. Cabrera, and C. J. B. Abbas, “Routing protocols in wireless sensor networks,” *Sensors*, vol. 9, no. 11, pp. 8399–8421, 2009.

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

[22] A. A. Anasane and R. A. Satao, “A Survey on Various Multipath Routing Protocols in Wireless Sensor Networks,” *Procedia Computer Science*, vol. 79, no. 20, pp. 610–615, 2016.

[23] K. Sha, J. Gehlot, and R. Greve, “Multipath routing techniques in wireless sensor networks: A survey,” *Wireless Personal Communications*, vol. 70, no. 2, pp. 807–829, 2013.

[24] N. Vijayarani and A. Senthilkumar, “Multipath Routing Protocols in Wireless Sensor Networks: A Retrospective Review,” *Indian Journal of Science and Technology*, vol. 10, no. 17, pp. 1–9, 2017

[25] M. Radi, B. Dezfouli, K. A. Bakar, and M. Lee, “Multipath routing in wireless sensor networks: Survey and research challenges,” *Sensors*, vol. 12, no. 1, pp. 650–685, 2012.

[26] M. Masdari and M. Tanabi, “Multipath Routing protocols in Wireless Sensor Networks: A Survey and Analysis,” *International Journal of Future Generation Communication and Networking*, vol. 6, no. 6, pp. 181–192, 2013.

[27] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, “Impact of network density on data aggregation in wireless sensor networks,” *22nd International Conference on Distributed Computing Systems*, Vienna, Austria, pp. 17–18, 2002.\

[28] E. Cayirci, “Data aggregation and dilution by modulus addressing in wireless sensor networks,” *IEEE Communications Letters*, vol. 7, no. 8, pp. 355–357, 2003.

[29] R. R. Rout and S. K. Ghosh, “Adaptive data aggregation and energy efficiency using network coding in a clustered wireless sensor network: An analytical approach,” *Computer Communication*, vol. 40, pp. 65–75, 2014.

[30] R. Banerjee, “Cluster Based Routing Algorithm with Evenly Load Distribution for Large Scale Networks,” *International Conference on Computer Communication and Informatics,*, Coimbatore, India, pp. 1–6, 2014.

[31] P. N. Renjith and E. Baburaj, “An analysis on data aggregation in Wireless Sensor Networks,” *2012 International Conference on Radar, Communication and Computing (ICRCC)*, Madras, India, pp. 62–71, 2012.

[32] S. G. S. Yadav and A. Chitra, “Reviewing the process of data fusion in wireless sensor network : a brief survey ,” *International Journal of Wireless and Mobile Computing*, vol. 8, no. 2, pp. 130–140, 2015.

[33] R. I. Da Silva, D. F. Macedo, and J. M. S. Nogueira, “Spatial query processing in wireless sensor networks - A survey,” *Information Fusion*, vol. 15, no. 1, pp. 32–43, 2014.

[34] J. E. B. Maia, A. Brayner, and F. Rodrigues, “A framework for processing complex queries in wireless sensor networks,”*ACM SIGAPP Applied Computing Review*, vol. 13, no. 2, pp. 30–41, 2013.

[35] T. K. Mishra and R. K. Paul, “A Survey on Load Balancing Techniques for Wireless Sensor Networks,” *International Journal of Advanced Research in Computer and Communication Engineering* ,vol. 6, no. 2, pp. 342–348, 2017.

[36] R. Kacimi, R. Dhaou, and A. L. Beylot, “Load balancing techniques for lifetime maximizing in wireless sensor networks,” *Ad Hoc Networks*, vol. 11, no. 8, pp. 2172–2186, 2013.

[37] D. Wajgi and N. V Thakur, “Load Balancing Algorithms in Wireless Sensor Network : A Survey,” *IRACST – International Journal of Computer Networks and Wireless Communications*, vol. 2, no. 8, pp. 2250–3501, 2012.

[38] I. Dietrich and F. Dressler, “On the lifetime of wireless sensor networks,” *ACM Transactions on Sensor Networks,* vol. 5, no. 1, pp. 1–39, 2009.\

[39] N. Saeed, M. Murad, M. Nawaz, and M. Irum, “Survey on Single Path and Multipath Energy Efficient Routing Protocols for Wireless Sensor Networks,” Journal of Computer and Communications., vol. 5, no. 5, pp. 1–11, 2017.

[40] P. Gopi, “Multipath Routing in Wireless Sensor Networks: A Survey and Analysis,” *IOSR Journal of Computer Engineering*, vol. 16, no. 4, pp. 27–34, 2014.

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

[42] M. Maimour, “Maximally radio-disjoint multipath routing for wireless multimedia sensor networks,” *Proceedings of the 4th ACM workshop on Wireless multimedia networking and performance modeling,* NY, USA, pp. 26–31, 2008.

[43] Z. Wang, E. Bulut, and B. K. Szymanski, "Energy Efficient Collision Aware Multipath Routing for Wireless Sensor Networks," in *IEEE International Conference on Communications*, 2009, pp. 1–5.

[44] M. Radi, B. Dezfouli, S. A. Razak, and K. A. Bakar, "LIEMRO: A Low-Interference Energy-efficient Multipath Routing protocol for improving QoS in event-based Wireless Sensor Networks," *Proceedings - 4th International Conference on Sensor Technologies and Applications, SENSORCOMM 2010*, Venice, Italy, pp. 551–557, 2010.

[45] P. Hurni and T. Braun, "Energy-efficient multi-path routing in wireless sensor networks," *Ad-hoc, Mobile and Wireless Networks*, Sophia-Antipolis, France, pp. 72–85, 2008.

[46] R. Vidhyapriya and P. Vanathi, "Energy efficient adaptive multipath routing for wireless sensor networks," *IAENG International Journal of Computer Science*, vol. 34, no.1, pp. 1-9,, August, 2007.

[47] S. Sharma, P. Agarwal, and S. K. Jena, "EAMRP: energy aware multipath routing protocol for wireless sensor networks," *International Journal of Information and Communication Technology*, vol. 8, no. 2, pp. 235-248, 2016.

[48] S. K. Gupta, P. Kuila, P. K. Jana, and I. S. Member, "Energy Efficient Multipath Routing for Wireless Sensor Networks : A Genetic Algorithm Approach," *International Conference on Advances in Computing, Communications and Informatics*, Jaipur, India, pp. 1735–1740, 2016.

[49] R. Neelisetti and A. Lim, "Delay-constrained high throughput protocol for multi-path transmission over wireless multimedia sensor networks," *International Symposium on a World of Wireless Mobile and Multimedia Networks*, CA, USA, pp. 1–8, 2008.

[50] X. Huang and Y. Fang, "Multiconstrained QoS multipath routing in wireless sensor networks," *Wireless Networks*, vol. 14, no. 4, pp. 465–478, 2008.

[51] A. Bagula and K. Mazandu, "Energy constrained multipath routing in wireless sensor networks," *Ubiquitous Intelligence Computing*, Springer, Berlin, Heidelberg, pp. 453–467, 2008.

[52] W. Lou and Y. Kwon, "H-SPREAD: A hybrid multipath scheme for secure and reliable data collection in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 4, pp. 1320–1330, 2006.

[53] E. Felemban, Chang-Gun Lee, and E. Ekici, "MMSPEED: multipath Multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 738–754, 2006.

[54] W. Lou, "An Efficient N-to-1 Multipath Routing Protocol in Wireless Sensor Networks," in *2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2005)*, 2005, pp. 665–672.

[55] X. Sun, H. Chen, X. Wu, X. Yin, and W. Song, "Opportunistic communications based on distributed width-controllable braided multipath routing in wireless sensor networks," *Ad Hoc Networks*, vol. 36, pp. 349–367, 2016.

[56] S. Xiao, B. Li, and X. Yuan, "Maximizing precision for energy-efficient data aggregation in wireless sensor networks with lossy links," *Ad Hoc Networks*, vol. 26, pp. 103–113, 2015.

[57] S. De, C. Qiao, and H. Wu, “Meshed multipath routing with selective forwarding: An efficient strategy in wireless sensor networks,” *Computer Networks*, vol. 43, no. 4, pp. 481–497, 2003.

[58] X. Hong, M. Gerla, H. Wang, and L. Clare, “Load balanced, energy-aware communications for Mars sensor networks,” *IEEE Aerospace Conference Proceedings*, MT, USA, pp. 1109–1115, 2002.

[59] J. Jiang, G. Han, H. Guo, L. Shu, and J. J. P. C. Rodrigues, “Geographic multipath routing based on geospatial division in duty-cycled underwater wireless sensor networks,” *Journal of Network and Computer Applications*, vol. 59, pp. 4–13, 2016.

[60] K.-Y. Shin, J. Song, J. Kim, M. Yu, and P. S. Mah, “REAR: Reliable Energy Aware Routing Protocol for Wireless Sensor Networks,” *9th International Conference on Advanced Communication Technology*, Kobe, Japan, pp. 525–530, 2007.

[61] A. Hadjidj, A. Bouabdallah, and Y. Challal, “HDMRP: An Efficient Fault-Tolerant Multipath Routing Protocol for Heterogeneous Wireless Sensor Networks,” *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, vol. 74 , pp. 469–482, 2012.

[62] C. I. G. E. H. Silva, “Directed diffusion for wireless sensor networking,” IEEE/ACM Transactions on Networking, vol. 11, no. 1, pp. 2–16, 2003.

[63] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, “Highly-resilient, energy-efficient multipath routing in wireless sensor networks,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 4, pp. 11–25, 2001.

[64] W. Tang, X. Ma, J. Huang, and J. Wei, “Toward Improved RPL : A Congestion Avoidance Multipath Routing Protocol with Time Factor for Wireless Sensor Networks,” *Journal of Sensors*, vol. 2016, pp. 1-11, 2016.

[65] A. Premita, “ A Review on Power Efficient Energy- Aware Routing Protocol for Wireless Sensor Networks”, *International Journal of Engineering Research & Technology (IJERT)*, vol. 1, no. 4, pp. 1–8, 2012.

[66] O. León, J. Hernández-Serrano, and M. Soriano, “An energy-efficient multipath routing protocol for wireless sensor networks,” *International Journal of Communication Systems*, vol. 23, no. 5, pp. 633–652, 2010.

[67] P. Li, H. Zhao, J. Wang, Z. Liu, and Y. Chun-yang, “A Reliable Multi-path Routing Protocol in Wireless Sensor Network Design and Implementation,” *2009 Ninth International Conference on Hybrid Intelligent Systems*, Shenyang, China, pp. 271–274, 2009.

[68] G. Treplan, L. Tran-Thanh, and J. Levendovszky, “Energy efficient reliable cooperative multipath routing in wireless sensor networks,” *World Academy of Science, Engineering and Technology*, vol. 4, no. 8, pp. 1136–1141, 2010.

[69] Y. Wang, H. Mao, C. Tsai, and C. Chuang, “HMRP : Hierarchy-Based Multipath Routing Protocol for,” *International Conference on Embedded and Ubiquitous Computing*, Nagasaki, Japan, pp. 452–459, 2005.

[70] R. X. R. Xiu-li, L. H. L. Hong-wei, and W. Y. W. Yu, “Multipath Routing Based on Ant Colony System in Wireless Sensor Networks,” *2008 International Conference on Computer Science and Software Engineering,* Hubei, China, pp. 202–205, 2008.

[71] J. Yang, W. Zhao, M. Xu, and B. Xu, “A Multipath Routing Protocol Based on Clustering and Ant Colony Optimization for Wireless Sensor Networks,” *International Journal of Computer Network and Information Security Network,* Vol. 1, pp. 49–59, 2009.

[72] S. Han, Z. Zhong, H. Li, G. Chen, E. Chan, and A. K. Mok, “Coding-Aware Multi-path Routing in Multi-Hop Wireless Networks,” 2*008 IEEE International Performance, Computing and Communications Conference*, Texas,USA, pp. 93–100, 2008.

[73] B. Yahya and J. Ben-othman, “REER : Robust and Energy Efficient Multipath Routing Protocol for Wireless Sensor Networks,” *GLOBECOM'09 Proceedings of the 28th IEEE conference on Global telecommunications,* Hawaii, USA, pp. 5482-5488, 2009.

[74] P. Kuila and P. K. Jana, “Approximation schemes for load balanced clustering in wireless sensor networks,” *The Journal of Supercomputing,* vol. 68, no. 1, pp. 87–105, 2014.

[75] G. Gupta and M. Younis, “Load-balanced clustering of wireless sensor networks,” *IEEE International Conference on Communications*, Anchorage, AK, USA, pp. 1848–1852, 2003.

[76] H. Zhang, L. Li, X. F. Yan, and X. Li, “A load-balancing clustering algorithm of WSN for data gathering,” *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce,* Dengleng, China, pp. 915–918, 2011.

[77] N. Kim, J. Heo, H. S. Kim, W. H. Kwon, “Reconfiguration of clusterheads for load balancing in wireless sensor networks,” *3rd International Symposium on Intelligent Information Technology Application, IITA,* Shanghai, China, pp. 7–10, 2009.

[78] L. Endre, “Novel Load Balancing Scheduling Algorithms for Wireless Sensor Networks,” *Fourth International Conference on Communication Theory, Reliability, and Quality of Service*, Budapest, Hungary, pp. 54–59, 2011.

[79] S. Ozdemir, “Secure Load Balancing via Hierarchical Data Aggregation in Heterogeneous Sensor Networks,” *Journal of Information Science and Engineering,* vol. 25, pp. 1691–1705, 2009.

[80] Y. Zhang, Z. Zheng, and Y. Jin, “Load-balanced Algorithm in Wireless Sensor Networks Based on Pruning Mechanism,” *IEEE International Conference on Communication Software and Networks*, Macau, China, pp. 604-606, 2009.

[81] M. Mahdavi, M. Ismail, and K. Jumari, “Load balancing in energy efficient connected coverage wireless sensor network,” *International Conference on Electrical Engineering and Informatics*, Selangor, Malaysia, pp. 448–452, 2009.

[82] C. M. Cianci, V. Trifa, and A. Martinoli, “Threshold-based algorithms for power-aware load balancing in sensor networks,” *IEEE Swarm Intelligence Symposium, SIS,* Pasadena, CA, pp. 359–366, 2005.

[83] P. Kuila, S. K. Gupta, and P. K. Jana, “A novel evolutionary approach for load balanced clustering problem for wireless sensor networks,” *Swarm and Evolutionary Computation*, vol. 12, pp. 48–56, 2013.

[84] P. T. Sivasankar and M. RamaKrishnan, "Vitality Aware Cluster Head Election to Alleviate the Wireless Sensor Network for Long Time", *International Journal of Advanced Computer Science and Applications,*Vol. 6, No. 12, pp. 36-41, 2015.

[85] Y. Deng, Y. Hu, and A. Assumptions, "A Load Balance Clustering Algorithm for Heterogeneous Wireless Sensor Networks," *2010 International Conference on E-Product E-Service and E-Entertainment (ICEEE)*, Henan, China, pp. 4–7, 2010.

[86] M. Zeynali and M. A. Jamali, "Fuzzy based approach for load balanced distributing database on sensor network," *Communications in Computer and Information Science*, vol. 63, no. 2, pp. 215–220, 2009.

[87] M. Perillo, Z. Cheng, and W. Heinzelman, "On the Problem of Unbalanced Load Distribution in Wireless Sensor Networks," *IEEE GlobeCom Work*, Dallas,TX, USA, pp. 74–79, 2004.

[88] H. Kim, G. De Veciana, X. Yang, and M. Venkatachalam, "Distributed-optimal user association and cell load balancing in wireless networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 1, pp. 177–190, 2012.

[89] J. He, S. Ji, Y. Pan, and Z. Cai, "Approximation algorithms for load-balanced virtual backbone construction in wireless sensor networks," *Theoretical Computer Science*, vol. 50, No.7, pp. 2–16, 2013.

[90] C. Petrioli, M. Nati, P. Casari, M. Zorzi, and S. Basagni, "ALBA-R: Load-balancing geographic routing around connectivity holes in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 3, pp. 529–539, 2014.

[91] C. P. Chen, S. C. Mukhopadhyay, C. L. Chuang, M. Y. Liu, and J. A. Jiang, "Efficient coverage and connectivity preservation with load balance for wireless sensor networks," *IEEE Sensors Journal*, vol. 15, no. 1, pp. 48–62, 2015.

[92] B. Baranidharan and B. Santhi, "DUCF: Distributed load balancing Unequal Clustering in wireless sensor networks using Fuzzy approach," *Applied Soft Computing*, vol. 40, pp. 495–506, 2016.

[93] K. Van Nguyen, P. Le Nguyen, Q. H. Vu, and T. Van Do, "An energy efficient and load balanced distributed routing scheme for wireless sensor networks with holes," *Journal of Systems and Software*, vol. 123, pp. 92–105, 2017.

[94] W. Zuo, H. Xie, Y. Lin, H. Hu, and Z. Cai, "A Load Balancing Optimization Algorithm for Context-Aware Wireless Sensor Networks Based on Fuzzy Neural Networks," *American Journal of Neural Networks and Applications*, vol. 2, no. 2, pp. 6-16, 2017.

[95] X. Liu and H. Xin, "Variable Weight Based Clustering Approach for Load Balancing in Wireless Sensor Networks," *4th International Conference on Geo-Informatics in Resource Management and Sustainable Ecosystem, GRMSE 2016*, Hong Kong, China, pp. 80-90,2016.

[96] A. R. M. Kamal and M. A. Hamid, "Supervisory routing control for dynamic load balancing in low data rate wireless sensor networks," *Wireless Networks*, vol. 23, no. 4, pp. 1–15, 2016.

[97] H. Y. Kim, "An energy-efficient load balancing scheme to extend lifetime in wireless sensor networks," *Cluster Computing*, vol. 19, no. 1, pp. 279–283, 2016.

[98] C. Gherbi, Z. Aliouat, and M. Benmohammed, "An adaptive clustering approach to dynamic load balancing and energy efficiency in wireless sensor networks," *Energy*, vol. 114, pp. 647–662, 2016.

[99] N. Baroutis and M. Younis, "Load-conscious maximization of base-station location privacy in wireless sensor networks," *Computer Networks*, vol. 124, pp. 126–139, 2017.

[100] M. Saleem, G. a. Di Caro, and M. Farooq, "Swarm intelligence based routing protocol for wireless sensor networks: Survey and future directions," *Information Sciences*, vol. 181, no. 20, pp. 4597–4624, 2011.

[101] P. Kuila and P. K. Jana, "Energy Efficient Load-Balanced Clustering Algorithm for Wireless Sensor Networks," *2nd International Conference on Communication, Computing & Security [ICCCS-2012]*, Rourkela, India, vol. 6, pp. 771–777, 2012.

[102] M. Azharuddin, P. Kuila, P. K. Jana, and S. Thampi, "Energy efficient fault tolerant clustering and routing algorithms for wireless sensor networks," *Computers and Electrical Engineering*, vol. 41, no. 3, pp. 177–190, 2014.

[103] P. Kuila and P. K. Jana, "A novel differential evolution based clustering algorithm for wireless sensor networks," *Applied Soft Computing Journal*, vol. 25, pp. 414–425, 2014.

[104] P. Kuila and P. K. Jana, "Energy efficient clustering and routing algorithms for wireless sensor networks: Particle swarm optimization approach," *Engineering Applications of Artificial Intelligence*, vol. 33, pp. 127–140, 2014.

[105] A. M. Zungeru, L.-M. Ang, and K. P. Seng, "Classical and swarm intelligence based routing protocols for wireless sensor networks: A survey and comparison," *Journal of Network and Computer Applications*, vol. 35, no. 5, pp. 1508–1536, 2012.

[106] H. Ali, W. Shahzad, and F. A. Khan, "Energy-efficient clustering in mobile ad-hoc networks using multi-objective particle swarm optimization," *Applied Soft Computing Journal*, vol. 12, no. 7, pp. 1913–1928, 2012.

[107] Y. Yao, Q. Cao, and A. V. Vasilakos, "EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for wireless sensor networks," *IEEE/ACM Transaction on Networking*, vol. 23, no. 4, pp. 182–190, 2013.

[108] N. B. R. Bouabdallah, Fatma; Bouabdallah, "Load-balanced routing scheme for energy-efficient wireless sensor networks," *Global Telecommunications Conference*, New Orleans, USA, pp. 1-6, 2008.

[109] S. Potthuri, T. Shankar and A. Rajesh, "Lifetime Improvement in Wireless Sensor Networks using Hybrid Differential Evolution and Simulated Annealing (DESA)," *Ain Shams Engineering Journal*, pp. 1-9, 2016.

[110] R. Kumar and N.A. Kumar, "Load Balancing Clustering in WSN using MABC," *Research Journal of Information Technology*, vol. 6, no. 4, pp. 389–398, 2014.

[111] Y. Yue, J. Li, H. Fan, and Q. Qin, "Optimization-Based Artificial Bee Colony Algorithm for Data Collection in Large-Scale Mobile Wireless Sensor Networks," *Journal of Sensors*, vol. 2016, no. 1, 2016.

[112] A. Raha, M. K. Naskar, and A. Paul, "A Genetic Algorithm Inspired Load Balancing Protocol for Congestion Control in Wireless Sensor Networks using Trust Based Routing Framework (GACCTR)," *I. J. Computer Network and Information Security,* no. 9, pp. 9–20, 2013.

[113] A. Mohajerani and D. Gharavian, "An ant colony optimization based routing algorithm for extending network lifetime in wireless sensor networks," *Wireless Networks*, vol. 22, no. 8, pp. 2637–2647, 2016.

[114] R. Wang, "An Ant Colony Optimization-based Load Balancing Routing Algorithm for Wireless Multimedia Sensor Networks," *2010 IEEE 12th International Conference on Communication Technology*, Nanjing, China, pp. 584–587, 2010.

[115] J. Yang, Y. Lin, W. Xiong, and B. Xu, "Ant Colony-Based Multi-Path Routing Algorithm for Wireless Sensor Networks," 2*009 International Workshop on Intelligent Systems and Applications*, Wuhan, China, pp. 1–4, 2009.

[116] A. M. Abd-Elmoniem, H. M. Ibrahim, M. H. Mohamed, and A.-R. Hedar, "Ant Colony and Load Balancing Optimizations for AODV Routing Protcol," *International Journal of Sensor Networks and Data Communications,* vol. 1, pp. 1–14, 2012.

[117] J. He, S. Ji, M. Yan, Y. Pan, and Y. Li, "Genetic-algorithm-based construction of load-balanced CDSs in wireless sensor networks," *Proceedings - IEEE Military Communications Conference MILCOM,* Baltimore, MD, USA, pp. 667–672, 2011.

[118] A. R. Revathi and B. Santhi, "Efficient Clustering for Wireless Sensor Networks using Evolutionary Computing," *Indian Journal of Science and Technology,* vol. 8, no.14, pp. 1–6, 2015.

[119] R. Kumar and D. Kumar, "Multi-objective fractional artificial bee colony algorithm to energy aware routing protocol in wireless sensor network," *Wireless Networks*, vol. 22, no. 5, pp. 1461–1474, 2015.

[120] S. Kaur and R. C. Gangwar, "Hybrid GSTEB routing protocol using clustering and artificial bee colony optimization," 2*015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, Noida, India pp. 661–666, 2015.

[121] P. S. Mann and S. Singh, "Improved metaheuristic based energy-efficient clustering protocol for wireless sensor networks," *Engineering Applications of Artificial Intelligence*, vol. 57, no. 2, pp. 142–152, 2017.

[122] R. Rajagopalan and P. K. Varshney, "Data-aggregation techniques in sensor networks: A survey," *IEEE Communications Surveys and Tutorials*, vol. 8, no. 4, pp. 48–63, 2006.

[123] P. Jesus, C. Baquero, and P. S. Almeida, "A Survey of Distributed Data Aggregation Algorithms,"*IEEE Communications Surveys and Tutorials,* vol. 17, no. 1, pp. 381–404, 2015.

[124] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks," *18th International Parallel and Distributed Processing Symposium,* NM, USA, pp. 32-40, 2004.

[125] T. He, B. M. Blum, J. a Stankovic, and T. Abdelzaher, “AIDA: Adaptive application-independent data aggregation in wireless sensor networks,” *ACM Transactions on Embedded Computing Systems*, vol. 3, no. 2, pp. 426-457, 2004.

[126] H. Chen, H. Mineno, and T. Mizuno, “Adaptive data aggregation scheme in clustered wireless sensor networks,” *Computer Communication.*, vol. 31, no. 15, pp. 3579–3585, 2008.

[127] D. Virmani, T. Sharma, and R. Sharma, “Adaptive Energy Aware Data Aggregation Tree for Wireless Sensor Networks,” *International Journal of Hybrid Information Technology*, Vol. 6, No.1, 2013.

[128] S. M. Kumar and N. Rajkumar, “SCT Based Adaptive Data Aggregation for Wireless Sensor Networks,” *Wireless Personal Communincations*, vol. 75, no. 4, pp. 2121–2133, 2013.

[129] W. Wu, J. Cao, H. Wu, and J. Li, “Robust and Dynamic Data Aggregation in Wireless Sensor Networks: A Cross-Layer Approach,” *Computer Networks*, vol. 57, pp. 3929-3940, 2012.

[130] I. Yoon, H. Kim, and D. Noh, “Adaptive Data Aggregation and Compression to Improve Energy Utilization in Solar-Powered Wireless Sensor Networks,” *Sensors*, vol. 17, no. 6, p. 12-26, 2017.

[131] C. Wei, J. Yang, and Y. Gao, “Cluster-based routing protocols in wireless sensor networks: A survey,” *Proceedings of 2011 International Conference on Computer Science and Network Technology*, Harbin, China, pp. 1659–1663, 2011.

[132] M. Sood and S. Kanwar, “Clustering in MANET and VANET: A survey,” *2014 Int. Conf. Circuits, Syst. Commun. Inf. Technol. Appl. CSCITA 2014*, pp. 375–380, 2014.

[133] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” *33rd Annual Hawaii International Conference on System Sciences*, Washington, USA, pp. 3005–3014, 2000.

[134] M. Tripathi, M.S. Gaur, V. Laxmi, R. B. Battula, “Energy Efficient Leach-C Protocol for Wireless Sensor Network", *Third International Conference on Computational Intelligence and Information Technology*, Mumbai, India, pp. 403–405.

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

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

[137] C. Li, M. Ye, G. Chen, and J. Wu, “An energy-efficient unequal clustering mechanism for wireless sensor networks,” 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS 2005 *MASS 2005*, Washington, DC, USA, vol. 2005, pp. 597–604, 2005.

[138] Y. . Qiang, B. . Pei, W. . Wei, and Y. . Li, “An efficient cluster head selection approach for collaborative data processing in wireless sensor networks,” *International Journal of Distributed S-ensor Networks,* vol. 11, no. 6, 2015.

[139] S. Younis, Ossama ;Fahmy, “HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks,” *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.

[140] A. Chaudhuri, S. Neogy, and B. B. Bhaumik, “Multi-hop Unequal Clustering Routing for Wireless Sensor Networks,” *Emerging Research in Computing, Information, Communication and Applications,* Bangalore, India, pp. 6–9, 2013.

[141] S. Banerjee and S. Khuller, “A Clustering Scheme for Hierarchical Control in Wireless Networks,” *IEEE INFOCOM*, Anchorage,USA, pp. 1028–1037, 2001.

[142] G. Smaragdakis, I. Matta, and a. Bestavros, “SEP: A stable election protocol for clustered heterogeneous wireless sensor networks,” *Second International Workshop on Sensor and Actor Network Protocols and Applications (SANPA 2004),* Boston, Massachusetts, pp. 1–11, 2004.

[143] L. Qing, Q. Zhu, and M. Wang, “Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks,” *Compuer Communincation*, vol. 29, no. 12, pp. 2230–2237, 2006.

[144] B. Elbhiri, S. Rachid, S. El Fkihi, and D. Aboutajdine, “Developed Distributed Energy-Efficient Clustering (DDEEC) for heterogeneous wireless sensor networks,” *2010 5th International Symposium on I/V Communications and Mobile Networks,* Rabat, Morocco, pp. 1–4, 2010.

[145] N. Javaid, T. N. Qureshi, A. H. Khan, A. Iqbal, E. Akhtar, and M. Ishfaq, “EDDEEC: Enhanced developed distributed energy-efficient clustering for heterogeneouswireless sensor networks,” *Procedia Computer Science*, vol. 19, pp. 914–919, 2013.

[146] K. T. Kim and H. Y. Youn, “A Stochastic and Optimized Energy Efficient Clustering Protocol for Wireless Sensor Networks,” *International Journal of Distributed Sensor Networks,* vol. 10, no. 3, 2014.

[147] W.-S. Jung, K.-W. Lim, Y.-B. Ko, and S.-J. Park, “A Hybrid Approach for Clustering-Based Data Aggregation in Wireless Sensor Networks,” *Third International Conference on Digital Society,* Cancun, Mexico, pp. 112–117, 2009.

[148] S. S. Wang and Z. P. Chen, “LCM: A link-aware clustering mechanism for energy-efficient routing in wireless sensor networks,” *IEEE Sensors Journal,* vol. 13, no. 2, pp. 728–736, 2013.

[149] F. Hannes and I. Stojmenović, “Geographic and energy-aware routing in Wireless Sensor Networks,” *International Journal of Ad Hoc and Ubiquitous Computing,* vol. 4, no. 2, pp. 61-70, 2009.

[150] O. Zytoune, Y. Fakhri, D. Aboutajdine, “Lifetime maximisation algorithm in Wireless Sensor Network", I*nternational Journal of Ad Hoc and Ubiquitous Computing*, vol. 6, no. 3, pp. 140–149, 2010.

[151] K. Lee and H. Lee, "A self-organized and smart-adaptive clustering and routing approach for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 8, no. 1,pp. 1-14, 2012.

[152] J. I. Kong, J. W. Kim, and D. S. Eom, "Energy-aware distributed clustering algorithm for improving network performance in WSNs," *International Journal of Distributed Sensor Networks*, vol. 10, no. 3,pp. 1-10, 2014.

[153] Y. Liao, H. Qi, and W. Li, "Load-balanced clustering algorithm with distributed self-organization for wireless sensor networks," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1498–1506, 2013.

[154] J. Zheng, S. Member, P. Wang, and S. Member, "Distributed Data Aggregation Using Slepian–Wolf Coding in Cluster-Based Wireless Sensor Networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2564–2574, 2010.

[155] W.-S. Jung, K.-W. Lim, Y.-B. Ko, and S.-J. Park, "Efficient clustering-based data aggregation techniques for wireless sensor networks," *Wireless Networks*, vol. 17, no. 5, pp. 1387–1400, 2011.

[156] K. Maraiya, K. Kant, and N. Gupta, "Efficient Cluster Head Selection Scheme for Data Aggregation in Wireless Sensor Network," *International Journal of Computer Applications*,vol. 23, no. 9, pp. 10–18, 2011.

[157] D. Mantri, N. R. Prasad, R. Prasad, and S. Ohmori, "Two Tier Cluster based Data Aggregation (TTCDA) in Wireless Sensor Network," *2012 IEEE International Conference on Advanced Networks and Telecommunications Systems*, Bangalore, India, pp. 117–122, 2012.

[158] J. Yuea, W. Zhang, W. Xiao, D. Tang, and J. Tang, "Energy Efficient and Balanced Cluster-Based Data Aggregation Algorithm for Wireless Sensor Networks," *Procedia Engineering*, vol. 29, pp. 2009–2015, 2012.

[159] D. Mantri, N. R. Prasad and R. Prasad, "Grouping of clusters for efficient data aggregation (GCEDA) in wireless sensor network," *3rd IEEE International Advance Computing Conference, IACC*, Ghaziabad, India, pp. 132–137, 2013.

[160] X. Xu, R. Ansari, A. Khokhar, and A. Vasilakos, "Hierarchical Data Aggregation Using Compressive Sensing (HDACS) in WSNs," *ACM Transactions on Sensor Networks*, vol. 11, no. 3, pp. 1769–1773, 2013.

[161] D. S. Mantri, N. R. Prasad, and R. Prasad, "Bandwidth efficient cluster-based data aggregation for Wireless Sensor Network," *Computers and Electrical Engineering*, vol. 41, pp. 256–264, 2015.

[162] S. Chatterjea, "A dynamic data aggregation scheme for wireless sensor networks," *14th Workshop on Circuits, Systems and Signal Processing, ProRISC*, Veldhoven, Netherlands, pp. 11-18, 2003.

[163] K. Miranda and V. Ramos, "Improving data aggregation in Wireless Sensor Networks with time series estimation," *IEEE Latin America Transactions*, vol. 14, no. 5, pp. 2425–2432, 2016.

[164] D. Mantri, N. R. Prasad, and R. Prasad, “MHBCDA Mobility and Heterogeneity aware Bandwidth Efficient Cluster based Data Aggregation for Wireless Sensor Network,” *3rd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE)*, NJ, USA, pp. 1-5, 2013.

[165] F. Khorasani and H. Naji, “Energy Efficient Data Aggregation in Wireless Sensor Networks Using Neural Networks,” *International Journal of Sensor Networks*, vol. 24, no. 1, pp. 1-18, 2015.

[166] S. Ozdemir and Y. Xiao, “Integrity protecting hierarchical concealed data aggregation for wireless sensor networks,” *Computer Networks*, vol. 55, no. 8, pp. 1735–1746, 2011.

[167] C. M. Chen, Y. H. Lin, Y. C. Lin, and H. M. Sun, “RCDA: Recoverable concealed data aggregation for data integrity in wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 4, pp. 727–734, 2012.

[168] Y. H. Lin, S. Y. Chang, and H. M. Sun, “CDAMA: Concealed data aggregation scheme for multiple applications in wireless sensor networks,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 7, pp. 1471–1483, 2013.

[169] C. Zhang, C. Li, and Y. Zhao, “A Balance Privacy-Preserving Data Aggregation Model in Wireless Sensor Networks,” *International Journal of Distributed Sensor Networks*, vol. 11, no. 6, 2015.

[170] S. Sicari, L. A. Grieco, G. Boggia, and A. Coen-Porisini, “DyDAP: A dynamic data aggregation scheme for privacy aware wireless sensor networks,” *Journal of Systems and Software*, vol. 85, no. 1, pp. 152–166, 2012.

[171] Y. Liu, C. Liu, and Q.-A. Zeng, “Improved trust management based on the strength of ties for secure data aggregation in wireless sensor networks,” *Telecommunication Systems*, vol. 62, no. 2, pp. 319-325, 2015.

[172] S. Gopikrishnan and P. Priakanth, “HSDA: hybrid communication for secure data aggregation in wireless sensor network,” *Wireless Networks*, vol. 22, no. 3, pp. 1061–1078, 2016.

[173] X. Zhao, J. Zhu, X. Liang, S. Jiang, and Q. Chen, “Lightweight and integrity-protecting oriented data aggregation scheme for wireless sensor networks,” *IET Information Security*, vol. 11, no. 2, pp. 82–88, 2017.

[174] H. Çam, S. Özdemir, P. Nair, D. Muthuavinashiappan, and H. Ozgur Sanli, “Energy-efficient secure pattern based data aggregation for wireless sensor networks,” *Computer Communication*, vol. 29, no. 4, pp. 446–455, 2006.

[175] H. Xu, L. Huang, Y. Zhang, H. Huang, S. Jiang, and G. Liu, “Energy-efficient cooperative data aggregation for wireless sensor networks,” *Journal of Parallel and Distributed Computing*, vol. 70, no. 9, pp. 953–961, 2010.

[176] Y. Li, L. Guo, and S. K. Prasad, “An energy-efficient distributed algorithm for minimum-latency aggregation scheduling in wireless sensor networks,” *International Conference on Distributed Computing Systems*, Genova, Italy, pp. 827–836, 2010.

[177] S. Misra and P. Dias Thomasinous, "A simple, least-time, and energy-efficient routing protocol with one-level data aggregation for wireless sensor networks," *Journal of Systems and Software*, vol. 83, no. 5, pp. 852–860, 2010.

[178] C. Chen, K. Lee, J. Park, and S. J. Baek, "Minimum Cost Data Aggregation for Wireless Sensor Networks Computing Functions of Sensed Data," *Journal of Sensors,* vol. 2015, 2015.

[179] I. R. Chen, A. P. Speer, and M. Eltoweissy, "Adaptive fault-tolerant QoS control algorithms for maximizing system lifetime of query-based wireless sensor networks," I*EEE Transactions on Dependable and Secure Computing*, vol. 8, no. 2, pp. 161–176, 2011.

[180] L. Xiang, J. Luo, and A. Vasilakos, "Compressed data aggregation for energy efficient wireless sensor networks," *8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks,* San Jose, CA, USA, pp. 46–54, 2011.

[181] H. Li, K. Lin, and K. Li, "Energy-efficient and high-accuracy secure data aggregation in wireless sensor networks," *Computer Communication,* vol. 34, no. 4, pp. 591–597, 2011.

[182] T.-W. Kuo, K. C.-J. Lin, and M.-J. Tsai, "On the Construction of Data Aggregation Tree with Minimum Energy Cost in Wireless Sensor Networks: NP-Completeness and Approximation Algorithms", *IEEE Transactions on Computers,* vol. 65, no. 10, pp. 1–1, 2015.

[183] C.-M. Chao and T.-Y. Hsiao, "Design of structure-free and energy-balanced data aggregation in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 37, pp. 229–239, 2014.

[184] C. Liu, Y. Liu, and Z. Zhang, "Improved Reliable Trust-Based and Energy-Efficient Data Aggregation for Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, vol.9, no. 5, 2013.

[185] S. B. Pourpeighambar, M. Aminian, and M. Sabaei, "Energy efficient data aggregation of moving object in Wireless Sensor Networks," *2011 Australasian Telecommunication Networks and Applications Conference,* Melbourne, Australia, pp. 1–8, 2011.

[186] S.-Y. Tsai, S.-I. Sou, and M.-H. Tsai, "Reducing Energy Consumption by Data Aggregation in M2M Networks," *Wireless Personal Communication*, vol. 74, no. 4, pp. 1231–1244, 2014.\

[187] M. Bala Krishna and M. N. Doja, "Multi-Objective Meta-Heuristic Approach for Energy-Efficient Secure Data Aggregation in Wireless Sensor Networks," *Wireless Personal Communication*, vol. 81, no. 1, pp. 1–16, 2015.

[188] B. Ramachandran, Gowri Sankar;Daniels, Wilfried ;Proença, José; Michiels, Sam;Joosen, Wouter ;Hughes, Danny ; Porter, "Hitch Hiker: A Remote Binding Model with Priority Based Data Aggregation for Wireless Sensor Networks," *Proceedings of the 18th International ACM SIGSOFT Symposium on Component-Based Software Engineering*, Montreal, QC, Canada, pp. 43–48, 2015.

[189] H. Lee, H. Hwang, T. Le Duc, M. H. Shon, H. Choo, and D. S. Kim, "Restructuring binomial trees for delay-aware and energy-efficient data aggregation in wireless sensor

networks," *9th International Conference on Ubiquitous Information Management and Communication - IMCOM '15*, New York, USA, pp. 1–4, 2015.

[190] M. G. Kim, Y. T. Han, and H. S. Park, "Energy-aware hybrid data aggregation mechanism considering the energy hole problem in asynchronous MAC-based WSNs," *IEEE Communication Letters*, vol. 15, no. 11, pp. 1169–1171, 2011.

[191] J. Zhang, Q. Wu, F. Ren, T. He, and C. Lin, "Effective Data Aggregation Supported by Dynamic Routing in Wireless Sensor Networks," *2010 IEEE International Conference on Communications*, Cape Town, South Africa, pp. 1–6, 2010.

[192] S. Abbasi-Daresari and J. Abouei, "Toward cluster-based weighted compressive data aggregation in wireless sensor networks," *Ad Hoc Networks*, vol. 36, pp. 368–385, 2016.

[193] S. Kandukuri, J. Lebreton, R. Lorion, N. Murad, and J. Daniel Lan-Sun-Luk, "Energy-efficient data aggregation techniques for exploiting spatio-temporal correlations in wireless sensor networks," *Wireless Telecommunication Symposium*, vol. 2016–May, 2016.

[194] C. Tang and N. Yang, "CoDA: Collaborative Data Aggregation in Emerging Sensor Networks Using Bio-Level Voronoi Diagrams," *Sensors*, vol. 16, pp. 12-35, 2016.

[195] H. Li, C. Wu, Q.-S. Hua, and F. C. M. Lau, "Latency-minimizing data aggregation in wireless sensor networks under physical interference model," *Ad Hoc Networks*, vol. 12, pp. 52–68, 2011.

[196] H. Liu, Z. Liu, D. Li, X. Lu, and H. Du, "Approximation algorithms for minimum latency data aggregation in wireless sensor networks with directional antenna," *Theoretical Computer Science*, vol. 497, pp. 139–153, 2013.

[197] W. Li, M. Bandai, and T. Watanabe, "Tradeoffs among delay, energy and accuracy of partial data aggregation in wireless sensor networks," *International Conference on Advanced Information Networking and Applications*, Perth, Australia, pp. 917–924, 2010.

[198] T.-S. P. Yum, "Optimal Routing and Data Aggregation for Maximizing Lifetime of Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 4, pp. 892–903, 2008.

[199] W. Wang, V. Srinivasan, and K.-C. Chua, "Extending the lifetime of wireless sensor networks through mobile relays," *IEEE/ACM Transactions on Networking*, vol. 16, no. 5, pp. 1108–1120, 2008

[200] M. Shan, G. Chen, D. Luo, X. Zhu, and X. Wu, "Building Maximum Lifetime Shortest Path Data Aggregation Trees in Wireless Sensor Networks," *ACM Transactions on Sensor Networks*, vol. 11, no. 1, pp. 11-24, 2014.

[201] P. Azad and V. Sharma, "Pareto-optimal clustering scheme using data aggregation for wireless sensor networks," *International Journal of Electronics*, vol. 102, no. 7, pp. 37–41, 2015.

[202] M. Asemani and M. Esnaashari, "Learning automata based energy efficient data aggregation in wireless sensor networks," *Wireless Networks*, vol. 21, no. 6, pp. 2035–2053, 2015.

[203] A. Awang and S. Agarwal, "Data Aggregation Using Dynamic Selection of Aggregation Points Based on RSSI for Wireless Sensor Networks," *Wireless Personal Communications*, vol. 80, no. 2, pp. 611–633, 2014.

[204] N.-T. Nguyen, B.-H. Liu, V.-T. Pham, and Y.-S. Luo, "On maximizing the lifetime for data aggregation in wireless sensor networks using virtual data aggregation trees," *Computer Networks*, vol. 105, pp. 99–110, 2016.

[205] R. Misra and C. Mandal, "Ant-aggregation: ant colony algorithm for optimal data aggregation in wireless sensor networks," *IFIP International Conference on Wireless and Optical Communications Networks*, Bangalore, India, pp. 1–5, 2006.

[206] W.H. Liao, Y. Kao and C. Fan, "An Ant Colony Algorithm for Data Aggregation in Wireless Sensor Networks," *International Conference on Sensor Technologies and Applications*, Valencia, Spain, pp. 101–106, 2007.

[207] C. Lin, G. Wu, F. Xia, M. Li, L. Yao, and Z. Pei, "Energy efficient ant colony algorithms for data aggregation in wireless sensor networks," *Journal of Computer and System Sciences,* vol. 78, no. 6, pp. 1686–1702, 2012.

[208] J.-H. Ho, H.-C. Shih, B.-Y. Liao, and S.-C. Chu, "A ladder diffusion algorithm using ant colony optimization for wireless sensor networks," *Information Sciences*, vol. 192, pp. 204–212, 2012.

[209] Y. Lu, I. S. Comsa, P. Kuonen, and B. Hirsbrunner, "Dynamic Data Aggregation Protocol based on Multiple Objective Tree in Wireless Sensor Networks," *IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP),* Singapore, pp. 7–9, 2015.

[210] B. Paul and E. Gopinathan, "Hybrid Data Aggregation Technique in Wireless Sensor Network through Classification of Fruitful Messages," *Fourth International Conference on Advances in Computing and Communications,* Cochin, India, pp. 157-160, 2014.

[211] S. Khan, K. K. Loo, Z.U. Din, "Cross layer design for routing and security in multi-hop wireless networks," *Journal of Information Assurance and Security,* vol. 4, pp. 170–173, 2009.

[212] A. El-Hoiydi and J. D. Decotignie, "Low power downlink MAC protocols for infrastructure wireless sensor networks," *Mobile Networks and Applications,* vol. 10, no. 5, pp. 675–690, 2005.

[213] M. Buettner, G. V Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks", *4th international conference on Embedded networked sensor systems,* New York, USA, pp. 307–320, 2006.

[214] Lakshmanan M and Noor Mohammed V, "An Adaptive Energy Efficient Mac Protocol for Wireless Sensor Networks," *International Journal on Recent and Innovation Trends in Computing and Communication,* vol. 3, no. 3, pp. 1026–1029, 2010.

[215] H. Gong, M. Liu, L. Yu, and X. Wang, "An Event Driven TDMA Protocol for Wireless Sensor Networks," *WRI International Conference on Communications and Mobile Computing,* Yunnan, China, pp. 132–136, 2009.

[216] D. Wu, G. Y. Wang, and X. L. Li, "Distributed TDMA scheduling protocol based on conflict-free for wireless sensor networks," *International Conference on Intelligent Computing and Integrated Systems,* Guilin, China, pp. 876–879, 2010.

[217] Z. Chen, S. Yang, L. Li, and Z. Xie, "A clustering approximation mechanism based on data spatial correlation in wireless sensor networks," *2010 Wireless Telecommunications Symposium*, Tampa, USA, pp. 1–7, 2010

[218] M. Bhardwaj, T. Garnett, and A. P. Chandrakasan, "Upper bounds on the lifetime of sensor networks," *IEEE International Conference on Communications*, Helsinki, Finland, pp. 785–790, 2001.

[219] C. Hu, Fei; Cao, Xiaojun; May, "Optimized scheduling for data aggregation in wireless sensor networks," *IEEE International Conference On Networking, Sensing and Control*, NV, USA, pp. 21-26, 2005.

[220] B. Yu, J. Li, and Y. Li, "Distributed Data Aggregation Scheduling in Wireless Sensor Networks," *IEEE INFOCOM*, Rio de Janeiro, Brazil, pp. 2159–2167, 2009.

[221] M. Bagaa, A. Derhab, N. Lasla, A. Ouadjaout, and N. Badache, "Semi-structured and unstructured data aggregation scheduling in wireless sensor networks," *2012 IEEE INFOCOM*, Orlando, USA, pp. 2671–2675, 2012.\

[222] D. Li, Q. Zhu, H. Du, and J. Li, "An improved distributed data aggregation scheduling in wireless sensor networks," *Journal of Combinatorial Optimization*, vol. 27, pp. 221–240, 2012.

[223] A. Jhumka, M. Bradbury, and S. Saginbekov, "Efficient fault-tolerant collision-free data aggregation scheduling for wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 74, no. 1, pp. 1789–1801, 2014.

[224] C. J. C. Joo, J.-G. Choi and N. B. Shroff, "Delay Performance of Scheduling with Data Aggregation in Wireless Sensor Networks," *IEEE INFOCOM*, San Diego,USA, pp. 1-9, 2010.

[225] M. Bagaa, M. Younis, D. Djenouri, A. Derhab, and N. Badache, "Distributed Low-Latency Data Aggregation Scheduling in Wireless Sensor Networks," *ACM Transactions on Sensor Networks*, vol. 11, no. 3, pp. 1–36, 2015.

[226] S. Kwon, J. H. Ko, J. Kim, and C. Kim, "Dynamic timeout for data aggregation in wireless sensor networks," *Computer Networks*, vol. 55, no. 3, pp. 650–664, 2011.

[227] H. O. Tan, I. Korpeoglu, and I. Stojmenovi, "Computing Localized Power-Efficient Data Aggregation Trees for Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 3, pp. 489–500, 2011.

[228] B. Hakoura, M. G. Rabbat, "Data Aggregation in Wireless Sensor Networks : A comparison of collection tree protocols and gossip algorithms", *IEEE Canadian Conference on Electrical and Computer Engineering*, Montreal, Canada, pp. 2–5, 2012.

[229] J. Lin, N. Xiong, a.V. Vasilakos, G. Chen, and W. Guo, "Evolutionary game-based data aggregation model for wireless sensor networks," *IET Communications*, vol. 5, no. 12, pp.. 1691-1697, 2011.

[230] L. A. Villas, D. L. Guidoni, R. B. Araujo, A. Boukerche, and A. a F. Loureiro, "A Scalable and Dynamic Data Aggregation Aware Routing Protocol for Wireless Sensor Networks Categories and Subject Descriptors," *13th ACM international conference on Modeling,*

analysis, and simulation of wireless and mobile system, New York, USA, pp. 110–117, 2010.

[231] I. Atoui, A. Ahmad, M. Medlej, A. Makhoul, S. Tawbe, and A. Hijazi, "Tree-based data aggregation approach in wireless sensor network using fitting functions," *2016* Sixth International Conference on Digital Information Processing and Communications, Beirut, Lebanon, pp. 146–150, 2016.

[232] G. Wei, Y. Ling, B. Guo, B. Xiao, and A. V. Vasilakos, "Prediction-based data aggregation in wireless sensor networks: Combining grey model and Kalman Filter," *Computer Communication*, vol. 34, no. 6, pp. 793–802, 2011.

[233] H. Jiang, S. Jin, and C. Wang, "Prediction or Not? An Energy-Efficient Framework for Clustering-based Data Collection in Wireless Sensor Networks," *IEEE Transactions on Communications,* vol. 39, no. 12, pp. 1721–1725, 1991.

[234] L. Meng, H. Zhang, and Y. Zou, "A Data Aggregation Transfer Protocol Based on Clustering and Data Prediction in Wireless Sensor Networks," *7th International Conference on Wireless Communications, Networking and Mobile Computing*, Wuhan, China, pp. 1–5, 2011.

[235] T. Liu, J. Peng, J. Yang, and C. Wang, "Energy efficient prediction clustering algorithm for multilevel heterogeneous wireless sensor networks," *Energy*, vol. 13, no. 3, pp. 71–75, 2011.

[236] H. Yousefi, M. H. Yeganeh, N. Alinaghipour, and A. Movaghar, "Structure-free real-time data aggregation in wireless sensor networks," *Computer Communication.*, vol. 35, no. 9, pp. 1132–1140, 2012.

[237] S. Dietzel, B. Bako, E. Schoch, and F. Kargl, "A fuzzy logic based approach for structure-free aggregation in vehicular ad-hoc networks," *Sixth ACM international workshop on Vehicular Internetworking,* New York, USA, pp. 79–88, 2009.

[238] M. S. Haghighi, Y. Xiang, V. Varadharajan, and B. Quinn, "A stochastic time-domain model for burst data aggregation in IEEE 802.15.4 wireless sensor networks," *IEEE Transactions on Computers,* vol. 64, no. 3, pp. 627–639, 2015.

[239] F. Ren, J. Zhang, Y. Wu, T. He, and C. Chen, "Attribute-Aware Data Aggregation Using Potential-Based Dynamic Routing in Wireless Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems,* vol. 24, no. 5, pp. 881–892, 2013.

[240] T. D. Engouang, Y. Liu, and Z. Zhang, "GABs : A Game-based Secure and Energy Efficient Data Aggregation for Wireless Sensor Networks," *International Journal of Distributed Sensor Network,* Vol. 11, No.1, pp. 1–31

[241] M. Iqbal, M. Naeem, A. Anpalagan, A. Ahmed, and M. Azam, "Wireless Sensor Network Optimization: Multi-Objective Paradigm," *Sensors*, vol. 15, pp. 17572–17620, 2015.

[242] B. A. Attea, E. A. Khalil, and A. Cosar, "Multi-objective evolutionary routing protocol for efficient coverage in mobile sensor networks," *Soft Computing*, vol. 19, no. 10, pp. 2983–2995, 2015.

[243] Z. Fei, B. Li, S. Yang, C. Xing, H. Chen, and L. Hanzo, "A Survey of Multi-Objective Optimization in Wireless Sensor Networks: Metrics, Algorithms and Open Problems," , vol. 19, no. 1, pp. 1–38, 2016.

[244] S. Jin, M. Zhou, and A. S. Wu, "Sensor network optimization using a genetic algorithm," 7th *World Multiconference on Systemics, Cybernetics and Informatics,* Florida, USA, pp. 1–6, 2003.

[245] H. Ouchitachen, A. Hair, and N. Idrissi, "Improved multi-objective weighted clustering algorithm in Wireless Sensor Network," Egyptian Informatics Journal, vol. 18, no. 1, pp. 45–54, 2017.

[246] S. Hussain, A.W. Matin, O. Islam, "Genetic Algorithm for Hierarchical Wireless Sensor Networks," *Journal of Networks,* vol. 2, no. 5, pp. 87–97, 2013.

[247] S. Peiravi, Ali ; Mashhadi, Habib Rajabi; Hamed Javadi, "An optimal energy- efficient clustering method in wireless sensor networks using multi- objective genetic algorithm," *Int. J. Commun. Syst.*, vol. 26, no. 1, pp. 114–126, 2011.

[248] S. Özdemir, B. A. Attea, and Ö. A. Khalil, "Multi-objective evolutionary algorithm based on decomposition for energy efficient coverage in wireless sensor networks," *Wireless Personal. Communication*, vol. 71, no. 1, pp. 195–215, 2013.

[249] Y. Lu, J. Chen, I. Comsa, P. Kuonen, and B. Hirsbrunner, "Construction of data aggregation tree for multi-objectives in wireless sensor networks through jump particle swarm optimization," *Procedia Computer Science,* vol. 35, no. 3, pp. 73–82, 2014.

[250] S. M. Jameii, K. Faez, and M. Dehghan, "Multiobjective optimization for topology and coverage control in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol.11, no. 1, pp. 1-11, 2015.

[251] D. R. Prasad, P. V Naganjaneyulu, and K. S. Prasad, "Energy Efficient Clustering in Multi-hop Wireless Sensor Networks Using Differential Evolutionary MOPSO," *Brazilian Archives of Biology and Technology*, vol. 59, no. 12, pp. 1–15, 2016.

[252] A. Sanderson and R. Graves, "Multi-Objective Routing in Wireless Sensor Networks with a Differential Evolution Algorithm," *2006 IEEE Int. Conf. Networking, Sens. Control*, pp. 880–885, 2006.

[253] A. Konstantinidis, K. Yang, Q. Zhang, and D. Zeinalipour-Yazti, "A multi-objective evolutionary algorithm for the deployment and power assignment problem in wireless sensor networks," *Computer Networks*, vol. 54, no. 6, pp. 960–976, 2010.

[254] A. Willig, "Wireless sensor networks: Concept, challenges and approaches," *Elektrotechnik und Informationstechnik*, vol. 123, no. 6, pp. 224–231, 2006.

[255] S. Sirsikar and S. Anavatti, "Issues of data aggregation methods in Wireless Sensor Network: A survey," *Procedia Computer Science.*, vol. 49, no. 1, pp. 194–201, 2015

[256] J. Wang, X. Yang, M. Wu, J. Kim, and I. Engineering, "An Energy-efficient Competitive Clustering Algorithm for Wireless Sensor Networks using Mobile Sink," *International Journal of Grid and Distributed Computing,* vol. 5, no. 4, pp. 79–92, 2012.

[257] X. Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg; Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," *Second International*

Conference on Knowledge Discovery and Data Mining, Portland, Oregon, pp. 226–231,1996.

[258] B. Manzoor, N. Javaid,O. Rehman,M. Akbar, Q. Nadeem, A. Iqbal,M. Ishfaq, "Q-LEACH: A new routing protocol for WSNs," *Procedia Computer Science*, vol. 19, pp. 926–931, 2013.

[259] R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[260] M. Daszykowski, B. Walczak, and D. L. Massart, "Looking for natural patterns in data. Part 1. Density-based approach," *Chemometrics and Intelligent Laboratory Systems*, vol. 56, no. 2, pp. 83–92, 2001.

[261] M. A. Al-Betar, A. T. Khader, Z. W. Geem, I. A. Doush, and M. A. Awadallah, "An analysis of selection methods in memory consideration for harmony search," *Applied Mathematics and Computation*, vol. 219, no. 22, pp. 10753–10767, 2013.

[262] "The SensorScope Grand-St-Bernard Deployment,." [Online]. Available: http://lcav.epfl.ch/cms/lang/en/pid/86035. [Accessed: 30-Jul-2017].

[263] M. C. M. Thein and T. Thein, "An energy efficient cluster-head selection for wireless sensor networks," *1st International Conference on Intelligent Systems, Modelling and Simulation*, Liverpool, UK, pp. 287–291, 2010.

[264] Y. M. E. Al-Refai, Hasan; Al-Awneh, Ali; Batiha, Khaldoun; Ali, Amer Abu; Rahman, "Efficient Routing LEACH (ER-LEACH) Enhanced on LEACH Protocol In Wireless Sensor Networks," *International Journal of Academic Research*, vol. 3, no. 3, pp. 42–48, 2011.

[265] A. Papadopoulos, A. Navarra, J. A. McCann, and C. M. Pinotti, "VIBE: An energy efficient routing protocol for dense and mobile sensor networks," *Journal of Network and Computer Applications*, vol. 35, no. 4, pp. 1177–1190, 2012.

[266] M. R. Senouci, A. Mellouk, H. Senouci, and A. Aissani, "Performance evaluation of network lifetime spatial-temporal distribution for WSN routing protocols," *Journal of Network and Computer Applications*, vol. 35, no. 4, pp. 1317–1328, 2012.

[267] J. Guan, D. Sun, A. Wang, and Y. Liu, "Energy Balanced Routing Algorithm Based on Mobile Sink for Wireless Sensor Networks" *Lecture Notes in Electrical Engineering*, vol 421. Springer, Singapore, pp. 603–613, 2012.

[268] M. S. Srouji, T. Bonny, J. Henkel, "High-speed Enoding / Decoding Technique for Reliable Data Transmission in Wireless Sensor Networks," *Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, Singapore, pp. 329–336, 2014.

[269] A. Serani, M. Diez, C. Leotardi, D. Peri, G. Fasano, U. Iemma, and E. F. Campana, "On the Use of Synchronous and Asynchronous Single-Objective Deterministic Particle Swarm Optimization in Ship Design Problems," *An International Conference on Engineering and Applied Sciences Optimization* , Kos Island, Greece , June 2014.

[270] Y. Shi and R. C. Eberhart, “Parameter Selection in Particle Swarm Optimization,” *Evolutionary Programming VII SE - 57*, vol. 14, no. 47, pp. 591–600, 1998.

[271] R. C. Eberhart and Y. Shi, “Comparing inertia weights and constriction factors in particle swarm optimization,” *Proceedings of the 2000 Congress on Evolutionary Computation*, CA, USA, pp. 84-88, 2000.

[272] T.-T. Wong, W.-S. Luk, and P.-A. Heng, “Sampling with Hammersley and Halton Points,” *Journal of Graphics Tools*, vol. 2, no. 2, pp. 9–24, 1997.

[273] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, “Multiobjective evolutionary algorithms: A survey of the state of the art,” Swarm and Evolutionary Computation, vol. 1, no. 1, pp. 32–49, 2011.

[274] O. Yilmaz, S. Demirci, Y. Kaymak, S. Ergun, and A. Yildirim, “Shortest hop multipath algorithm for wireless sensor networks,” Computers and Mathematics with Applications, vol. 63, no. 1, pp. 48–59, 2012.

[275] P. M. Pradhan and G. Panda, “Pareto optimization of cognitive radio parameters using multiobjective evolutionary algorithms and fuzzy decision making,” Swarm and Evolutionary Computation, vol. 7, pp. 7–20, 2012.

List of Publications

Published

International Journal (SCI/SCIE Indexed)

1. **Sukhchandan Randhawa** and Sushma Jain, “An Intelligent PSO based energy efficient load balancing multipath technique in wireless sensor networks”, *The Turkish Journal of Electrical Engineering & Computer Sciences* , vol. 25, no. 4, 2017, [Impact Factor = 0.578] , pp.3113-3131.

2. **Sukhchandan Randhawa** and Sushma Jain, “Data Aggregation In Wireless Sensor Networks: Previous Research, Current Status and Future Directions”, *Wireless Personal Communication* [Springer], July 2017, [Impact factor = 0.951] DOI :10.1007/s11277-017-4674-5.

3. **Sukhchandan Randhawa** and Sushma Jain, “An Intelligent PSO based energy efficient load balancing technique in wireless sensor networks”, *Sadhana* [Impact Factor = 0.465] (accepted).

4. **Sukhchandan Randhawa** and Sushma Jain, “DAHDA: Dynamic Adaptive Hierarchical Data Aggregation for Clustered Wireless Sensor Networks”, *Wireless Personal Communication* [Springer], August 2017, [Impact factor = 0.951] DOI: 10.1007/s11277-017-4843-6.

International Journal (Non- SCI)

1. **Sukhchandan Randhawa** and Sushma Jain, “Performance Analysis of LEACH with Machine Learning Algorithms in Wireless Sensor Network”, “International *Journal of Computer Application* ”, vol. 147, no. 2, 2016.

2. **Sukhchandan Randhawa** and Sushma Jain, “A Systematic Review on Energy Aware QoS Routing in Wireless Sensor Networks” *International Journal of Energy Information and Communications*, MECS, Vol.6, No. 5, pp.1-14, 2015.

Book Chapter

1. **Sukhchandan Randhawa** and Sushma Jain, “Multi-Objective Data Aggregation for Clustered Wireless Sensor Networks” *International Conference on Computing, Analytics and Networking* (ICCAN-2017) and AISC Series of Springer(Book), Bhubaneswar, Odisha, India, December 2017. [accepted](Scopus Indexed)

2. **Sukhchandan Randhawa** and Sushma Jain, “Multipath Routing in Wireless Sensor Networks: A Taxonomy and Future Research Directions” *International Conference on Recent Advancements in Computer, Communication and Computational Sciences (RAACE2017)* and Springer Book Series, Ajmer, India, November, 2017. [accepted](Scopus Indexed)

Communicated

1. **Sukhchandan Randhawa** and Sushma Jain, “Cross Layer Energy Efficient Clustering Technique for Heterogeneous Wireless Sensor Networks”, *Computer Networks Journal*, Elsevier, 2017, [Impact factor = 2.516].

2. **Sukhchandan Randhawa** and Sushma Jain, “Multi-Objective Load Balancing Clustering in Wireless Sensor Networks”, *Swarm and Evolutionary Computation*, Elsevier, 2017, [Impact factor = 3.893].