Bottleneck Nodes Avoidance Data Gathering Tree for Wireless Sensor Networks

Md. Morshedul Islam

A thesis submitted to the Department of Computer Science and Engineering in partial fulfilment of the requirements for the

degree of

Master of Science in Engineering
(Computer Science and Engineering)

Department of Computer Science and Engineering

Bangladesh University of Engineering and Technology

Dhaka, Bangladesh

January 2012

The thesis "***Bottleneck nodes avoidance data gathering tree for wireless sensor networks***", submitted by Md. Morshedul Islam, Roll No. 0409052010F, Session: April 2009, to the Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Master of Science in Engineering (Computer Science and Engineering) and approved as to its style and contents. Examination held on 21st January, 2012.

Board of Examiners

1. ______________________________

Dr. M. Mahfuzul Islam
Associate Professor
Department of CSE, BUET, Dhaka -1000

Chairman
(Supervisor)

2. ______________________________

Dr. M. Kaykobad
Professor& Head
Department of CSE, BUET, Dhaka -1000

Member
(Ex-officio)

3. ______________________________

Dr. Md. Saidur Rahman
Professor
Department of CSE,BUET, Dhaka -1000

Member

4. ______________________________

Dr. Md. Mostofa Akbar
Professor
Department of CSE,BUET, Dhaka -1000

Member

5. ______________________________

Dr. Salekul Islam
Assistant Professor
Department of EEC, NSU, Dhaka -1229

Member
(External)

Declaration

I declare that this thesis is my own work and has not been submitted in any form for another degree or diploma at any university or other institute of tertiary education. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Md. Morshedul Islam   Date:

Dedicated to my parents for all their love and inspiration

Acknowledgements

First of all I want to thank my Allah Almighty and then I am thankful to my parents who supported me in this whole study and always pray for my success and good health.

I would like to express my sincere gratitude and profound indebtedness to my supervisor Dr. M. Mahfuzul Islam for his guidance, valuable suggestions, commendable support and endless patience towards the completion of this thesis. I feel very proud to have worked with him. Without the inspiring enthusiasm and encouragement of my supervisor, this work could not have been completed.

I thank all the staffs and graduate students at Bangladesh University of Engineering and Technology (BUET) and all the friends for their support and encouragement. I would also like to extend my thanks to my elder and younger brothers.

Finally, I wish to express my gratitude to Bangladesh University of Engineering and Technology (BUET) for providing an excellent environment for research and all the other facilities to complete the thesis successfully.

Abstract

Latest development of electronic and communication technologies enabled Wireless Sensor Networks (WSNs) deployed for data collection from an area of interest. The sensed data is then aggregated and transmitted to the base station (sink), where the collected data is further processed for end-user queries. With the increase of WSNs applications, there has been a growing demand for gathering information from environments for an expected network lifetime. Since the sensor nodes have limited power and it is impractical to replace the energy source of the nodes, an energy efficient data gathering technique need to be design to shrinks the use of radio waves propagation to ensure a desired lifetime. According to the co-operative modes of the sensor nodes, the most popular data gathering techniques in WSNs are cluster-based, chain-based and tree-based. Due to the minimum graph structure and for easy implementation, we consider tree topology based algorithms in our research work. However, most of the existing tree based schemes introduce lower energy bottleneck nodes as intermediate nodes making them overburden during tree construction, shortening the lifetime of the network. To solve the overburden nodes problem, other categories of algorithms perform a load balancing operation to decrease the load of bottleneck nodes. In this thesis, we propose two load balancing tree based data gathering schemes. A mathematical model for load calculation and nodes categorized are also proposed for both the schemes. The first algorithm "Energy aware bottleneck node avoidance data gathering tree (EBA-DG)" is proposed to construct a load balance tree to gather data from environment. Considering the transmission cost with node degree an energy-aware load balancing scheme named "Energy aware bottleneck nodes avoidance shortest-path data gathering tree (EBASP-DG)" is also proposed. In EBASP-DG, a new data structure called data gathering sequence (DGS) is also proposed for the benefits of constructing tree. Theoretical analysis and simulation results show that both EBA-DG and EBASP-DG can achieve better performance than the existing other schemes.

Abbreviations

API	Application program interface
BS	Base station
CEDA	Cell based energy density aware routing
CLMT	Centralize lifetime maximizing tree
CSMA	Carrier sense multiple access
DB-MDST	Delay bounded minimum degree spanning tree
DCML	Delay constrain and maximum lifetime data gathering
DG	Data gathering
DG-EL	Data gathering routing algorithm based on energy level
DGS	Data gathering sequence
DoS	Denial of service
DPM	Dynamic power management
EBA-DG	Energy aware bottleneck nodes avoidance data gathering
EMLN-DG	Energy aware maximum leaf nodes data gathering
EPBASP-DG	Energy-aware bottleneck nodes avoidance shortest-path DG
ERAPL	Energy-efficient routing algorithm to prolong lifetime
ETR	Energy aware tree routing
FDMA	Frequency division multiple access
FHT	Fewest hope tree
GAs	Generic algorithms

GSM	Global system for mobile communications
HEED	Hybrid energy efficient distributed protocol
IP	Internet protocol
ISM	Industrial, scientific and medical
LEACH	Low energy adaptive clustering hierarchy
MDST	Minimum degree spanning tree
MIMO	Multiple input multiple output
ML-DG	Maximum-lifetime data gathering tree
MLDGA	Tree structure based data gathering for maximum lifetime
OTP	Outgoing traffic proposition
PAN	Personal area network
PEDAP	Power efficient data gathering and aggregation
PEDAP-PA	Power aware PEDAP
PEGASIS	Power efficient gathering in sensor information system
PSAs	Personal digital assistants
RF	Radio frequency
SPIM	Sensor protocols for information via negotiation
SPT	Shortest path tree
TCP	Transport control protocol
TDMA	Time division multiple access
WAN	Wide area network
WLAN	Wireless local area network
WPAN	Wireless personal area network
WSN	Wireless sensor network

Contents

Acknowledgements vi

Abstract vii

Abbreviations viii

Abstract vii

1. Introduction 1

2. Wireless Sensor Network 6

2.1 Introduction 6

2.2 Objectives and Applications 8

2.3 Properties of Sensor Networks 9

2.4 Architecture 10

2.4.1 Network Architecture 11

2.4.2 Sensor Node Architecture 13

2.4.2.1 Microcontroller System 13

2.4.2.2 Transceiver System 14

2.4.2.3 Memory 15

2.4.2.4 Power Source 15

2.4.2.5 Sensors 16

2.4.2.6 Operating System 16

2.4.2.7 TinyDB 17

2.4.2.8 Types of Sensor Nodes and Specifications 17

2.5 Communications Standards and Protocols 18
2.5.1 Communication Standards 18
2.5.1.1 IEEE 802.15 19
2.5.1.2 IEEE 802.15.4 19
2.5.1.3 IEEE 802.15.5 19
2.5.1.4 IEEE 1451 20
2.5.2 Frequencies used in WSN 20
2.5.3 Layered Architecture and Protocols 21
2.5.3.1 Physical Layer 21
2.5.3.2 Link Layer 21
2.5.3.3 Network Layer 22
2.5.3.4 Transport Layer 23
2.5.3.5 Application Layer 24
2.6 Power Management 24
2.6.1 Power Problem of Sensor Nodes 24
2.6.2 Power Preserving Issues 25
2.6.2.1 Tree Construction 26
2.6.2.2 Query Processing 26
2.6.2.3 Energy Efficient Routing 27
2.6.2.4 Efficient Node Placement 27
2.6.2.5 Node Admission Control 27
2.7 Security of Wireless Sensor Networks 27
2.7.1 Obstacles to WSN Security 28
2.7.2 Attacks 29
2.7.3 Defensive Measures 30
2.8 Comparative Analysis 31

2.8.1 Difference with Cellular System Networks 31
2.8.2 Difference with Wireless Local Area Networks 31
2.8.3 Difference with Wireless Ad-hoc Networks 32
2.9 Research Challenges 33
2.9.1 Collaborative information Gathering 34
2.9.2 Development of a New Transport Protocol 34
2.9.3 Development of a New Routing Protocol 34
2.9.4 Wireless Sensor Deployment 35
2.9.5 Security 35
2.9.6 Cross Layer Design 35
2.10 Motivation of Research on Tree Based Scheme 36
2.11 Summary and Comments 37
3. Data Gathering Tree 38
3.1 Introduction 38
3.2 Non Path-Cost Based Data Gathering Tree 41
3.2.1 Delay Bounded Minimum Degree Spanning Tree (DB-MDST) 41
3.2.2 Centralize Lifetime Maximizing Tree (CLMT) 42
3.2.3 Energy Aware Tree Routing (ETR) 43
3.2.4 Delay Constrain and Maximum Lifetime Data Gathering (DCML) 44
3.2.5 Energy Aware Maximum Leaf Nodes Data Gathering Tree (EMLN-DG) 46
3.3 Path-Cost Based Data Gathering Tree 47
3.3.1 Power Efficient Data Gathering and Aggregation (PEDAP) 48
3.3.2 Tree Structure Based Data Gathering for Maximum Lifetime (MLDGA) 49
3.3.3 Maximum-Lifetime Data Gathering Tree (ML-DG) 50
3.3.4 Data Gathering Routing Algorithm Based on Energy Level (DG-EL) 51
3.3.5 Energy-Efficient Routing Algorithm to Prolong Lifetime (ERAPL) 52

3.4 Summery and Comments........54

4. Simulation Setup........55

4.1 Introduction........56

4.2. Simulation Environment........56

4.2.1 SimJava........56

4.2.2. Simulation Parameters........58

4.3 Performance Parameters........58

4.4 Summery and Comments........60

5. Proposed Energy Aware Bottleneck Nodes Avoidance DG........61

5.1 Introduction........61

5.2 Motivation of the Proposed Scheme........63

5.3 Load Calculation of Sensor Nodes........64

5.4 Nodes Categorization Scheme........66

5.5 Energy-Aware Bottleneck-Nodes Avoidance DG (EBA-DG)........68

5.6 Simulation Results and Comparative Analysis........73

5.6.1 Energy Loss per Round........74

5.6.2 Average Lifetime of the Bottleneck Nodes........75

5.6.3 Lifetime of the Network........76

5.6.4 Height of the Tree........77

5.6.5 Throughput of the Network........78

5.7 Summary and Comments........79

6. Proposed Energy Aware Bottleneck Nodes Avoidance Shortest-Path DG..79

6.1 Introduction........79

6.2 Motivation of the Proposed Scheme........81

6.3 Data Gathering Sequence (DGS)........82

6.4 Energy-Aware Bottleneck-Nodes Avoidance Shortest-Path DG (EBASP-DG)................86
6.5 Simulation Results and Comparative Analysis..93
6.5.1 The Optimum value of α (alpha) in EBASP-DG..94
6.5.2 Energy Loss per Round..96
6.5.3 Average Lifetime of the Network 97
6.5.4 Average Lifetime of the Bottleneck Nodes..98
6.5.5 Average Lifetime of the Nodes 99
6. 5.6 Height of the Tree 100
6.5.7 Throughput of the Network...101
6.5.8 Total Path-cost of the Network 102
6.6 Summary and Comments...103
7. Conclusions and Future Works 104
Bibliography 107

List of Figures

Figure 2.1: Sensor network applications in different domain in our life..8

Figure 2.2: WSN architecture...12

Figure 2.3: Sensor nodes...13

Figure 2.4: The Mica2 mote uses AA batteries to the CPU/radio..15

Figure 3.1: Data gathering in WSN, a) without tree construction b) with tree construction............39

Figure 3.2: Types of data gathering algorithms 40

Figure3.3: DB-MST tree 42

Figure 3.4: Direct optimization 45

Figure 3.5: Indirect optimization..45

Figure 3.6: A network scenario, a) without DGS b) with DGS 53

Figure 4.1: A simulation layout 57

Figure 5.1: Network scenario with constant transmission-cost...68

Figure 5.2: Node Categorization 69

Figure 5.3: Tree construction (Iteration 1) 70

Figure 5.4: Tree construction (Iteration 2) 71

Figure 5.5: Final data gathering tree 73

Figure 5.6: Energy loss per round in EML-DG and EBA-DG..74

Figure 5.7: Average lifetime of the bottleneck nodes in EML-DG and EBA-DG...........................75

Figure 5.8: Lifetime of the network in EML-DG and EBA-DG...76

Figure 5.9: Height of the tree in EML-DG and EBA-DG..77

Figure 5.10: Throughput of EML-DG and EBA-DG..78

Figure 6.1: Network scenario with different path-cost...84

Figure 6.2: A DGS list 85

Figure 6.3: Network scenario with DGS...85

Figure 6.4: Node categorization without leaf node 86

Figure 6.5: Node categorization with leaf node 87

Figure 6.6: Tree construction (Iteration 1) 91

Figure 6.7: Tree construction (Iteration 2) 91

Figure 6.8: Tree construction (Iteration 3) 92

Figure 6.9: Tree construction (Iteration 4) 92

Figure 6.10: Final data gathering tree 93

Figure 6.11: Energy loss per round for different value of (α) alpha 94

Figure 6.12: Path-cost per round for different value of (α) alpha 95

Figure 6.13: Average round of bottleneck nodes for different value of (α) alpha..........................95

Figure 6.14: Energy loss per round in different schemes..96

Figure 6.15: Average lifetime of the network in different schemes..98

Figure 6.16: Average lifetime of the bottleneck nodes in different schemes.................................99

Figure 6.17: Average lifetime of the sensor nodes in different schemes 100

Figure 6.18: Height of the tree in different schemes...101

Figure 6.19: Throughput in different schemes..102

Figure 6.20: Total path-cost in different schemes...103

List of Tables

Table1: Several attacks in different layers of WSN and defensive methods………………….30

Table 2: Simulation parameters …………………………………………….………….....58

Chapter 1

Introduction

The massive growth in the use of Wireless Sensor Network (WSN) infrastructure has empowered researchers to explore and analyze several techniques used for communication. Sensor Networks have recently gained its popularity in several applications such as habitat sensing, seismic monitoring, battlefield surveillance, monitoring vehicle traffic, object tracking and many more. With the increase of WSNs applications, there has been a growing demand for gathering information from environments for an expected network lifetime. Once the sensor nodes deploy in an environment, they are generally inaccessible for the user to replace their resources such as batteries. The limitations are created due to the battery power of a sensor node therefore, made the issues of designing new architecture and protocols for WSNs, which are the most challenging parts [4]. Once a sensor node collects data from the environments, it sends the collected data to its neighbor nodes. The neighbor aggregates the collected data with its own data and propels it to the next neighbor nodes and the process continues until collected data reaches to the base station (sink), the controlling center of the network. The communication (send and receive data) in a WSN is considered as the most energy consuming activities, resulting, in shorter the network lifetime, not meeting the expectation as set during the deployment [4]. If network fails to continue due to the outage of battery power, the system is disrupted and network needs to be reinstalled by either rechargeable/replicable batteries or changing the sensor nodes. Both of the option is complicated, expensive and sometimes become impractical. Such a situation, therefore, demands for designing an energy efficient data gathering algorithm, shrinks the uses of radio waves propagation to ensure a desire lifetime of WSNs.

According to the co-operation mode of sensor nodes, existing data gathering algorithms can be divided into three distinct categories as: cluster-based, chain-based and tree-based [43]. In cluster-based protocols, some nodes are selected as cluster-head based by probability, and other nodes join to the closest cluster-head form a kind of Voronoi structural topology. The main objective of cluster based routing protocols is to efficiently maintain the energy consumption of sensor nodes by involving them in multi-hop communication within a particular cluster and performing data aggregation and fusion in order to decrease the number of transmitted messages to the sink. Typical cluster-based protocols include LEACH [24], HEED [86], CEDA [73] etc. The popularities of cluster based protocols are that they are easy to implement and the resources of cluster-head and its child nodes are easy to manage. Also die-out of one or more child nodes does not cause any data loss in the network. On the other hand, the main problems in cluster based algorithms are that the distribution of the cluster head is not symmetry; as a result some cluster head nodes become heavy loaded. Also the cluster head nodes perform most of the communication tasks (sending, receiving and aggregating data) and die-out earlier in the network causes information loss and decreases the lifetime of the network. The size of the cluster head (the number of child nodes) is also difficult to control in those categories of algorithms.

In chain-based protocols, all nodes in the network are organized as a chain. A higher energy node is selected as a chain head to communicate with the sink directly and all the other nodes will receive, aggregate and transmit their data towards the head through the chain. Here gathering data transmit from node to node, aggregates and eventually sent to the base station (sink) through the head of the network. Typical chain-based algorithms are PEGASIS [45], CCS [72], DRAEM [46] etc. Chain-based protocols enable the nodes to communicate with their closest neighbor by using low radio power which shrink loss of energy of sensor nodes and increase the lifetime of the network. Although the chain based protocols avoid the clustering overhead of cluster based protocols, still it requires dynamic topology adjustment since sensor’s energy is not tracked i.e. every sensor nodes need to be aware of the status of its neighbor so that they can know where to route that data. Such topology adjustment can introduce significant overhead especially for highly utilized networks. Also the long chain would cause larger delay in those chain based algorithms.

In tree-based protocols, all nodes in the network are organized as trees where the intermediate nodes collect data from the environment, aggregate it with the data received from its child nodes and then transmit the data to its parent. The leaf nodes are only sense data from the environment and route it to the parent node. The root (a highest energy node in the network) is responsible to collect all the data from the network and send it to the base station (sink), the command centre of the network. The most popular tree based protocols are Delay Constrain and Maximum Lifetime Data Gathering (DCML) [44], Energy Aware Maximum Leaf Nodes Data Gathering Tree (EMLN-DG)[51], Power Efficient Data Gathering and Aggregation (PEDAP)[25], Tree Structure Based Data Gathering for Maximum Lifetime (MLDGA) [88], Maximum-Lifetime Data Gathering Tree (ML-DG) [84], Data Gathering Routing Algorithm Based on Energy Level (DG-EL) [79], Energy-Efficient Routing Algorithm to Prolong Lifetime (ERAPL)[89], etc. Tree based schemes ensure less delay than the chain based schemes and reduce the overhead problem of cluster head in cluster based schemes. In tree based protocols the leaf nodes can reduce their energy consumption by periodically go to the sleeping mode. Also, the multi-hop communication reduces the usages of energy of the nodes which increases the lifetime of the network. For the simplicity and minimum graph structure [11], in this research work we have considered tree structure based protocols for our analysis.

The aim of this thesis is to contribute to develop a new tree based algorithm in WSN through a series of improvement in tree construction based on residual energy, reduce the number of bottleneck nodes by adjusting their load. This thesis, thus, focuses on the following major aims:

i. To efficiently calculate the load of the nodes by considering their node degree and the path- cost.
ii. To categorize all the nodes base on their residual energy (the remaining energy of the power source of sensor nodes) and load.
iii. To balance the energy loss among the nodes by constructing a load balancing tree based on the residual energy of the nodes. Higher energy node loss more energy per round than lower one.
iv. To avoid the disruption of data communications during the round time by restricting lower energy nodes to be intermediate nodes.
v. To construct a load balancing tree with limited path cost and delay.

To achieve the above mentioned aims, this thesis has devised tree based schemes by accommodating the abovementioned aims. The proposed schemes exploit, 1) ensure the lifetime of the network a predefine round-time 2) categorize the nodes based on their residual energy and load and construct a tree in such a way that the lower residual energy nodes will consume less energy per round-time than the higher one, 3) balance the load of the intermediate nodes based on their residual energy and transmission-cost, and 4) construct a tree with limited delay and lower path cost. These concepts and philosophies along with their implementation synergistically have made the following major noble contributions:

i. A tree based protocol which ensure the life time of the network as the predefine number of round-time by considering the residual energy of the nodes and their load.

ii. Categorize the nodes based on their residual energy and load; where higher energy nodes route most of the traffic load of the network. The load of the lower energy nodes is decreases by decreasing their node degree.

iii. Construct a data gathering sequence which increases the effectiveness of load calculation and enhance the probability to construct a tree with limited height and path-cost.

iv. Decrease the energy and packets loss by restricting the lower energy nodes to be intermediate nodes.

The remaining chapters of the thesis are organized as follows:

In chapter 2, a brief introduction of the WSN architecture is given to make familiar with this type of network. A details understanding about applications, characteristics, communication standards and protocols are also discussed. It also focuses the security issue and makes a comparative analysis with other types of existing wireless network. This chapter ends with the power management in WSN along with the research issue.

Chapter 3 provides a brief review of tree based data gathering techniques by dividing them into two distinct categories; non path-cost based and path-cost based data gathering schemes and then address their pros and cons. Finally, this chapter ends by giving a conclusion with some further direction.

Chapter 4 discusses about the simulator and simulation environments of the proposed schemes. The important simulation parameters are also listed here so that the realistic assumptions used in simulations can be viewed clearly. Then it gives a petite description on performance parameters that are used to compare the simulation results of the proposed and existing schemes.

Chapter 5 introduces by giving some motivation of this research. Then explains the proposed non path-cost based scheme, which includes some noble techniques for i) introducing a mathematical model to calculate the load of the nodes by considering their node degree and transmission-cost, ii) categorizing the nodes based on their residual energy and load and iii) proposing a new non path-cost based data gathering scheme by balancing the load of the overburden nodes. A comparative analysis on the simulation output of the proposed and existing non path-cost based scheme are also explain here.

Chapter 6 introduce a new path-cost based data structure named data gathering sequence (DGS) which solve the mutual transmission and loop transmission problem of WSNs. The DGS is also effective to measure the load of the nodes and explore the shortest-path of all nodes from the root. A new path-cost based scheme is then discuss which construct a load balancing energy-based shortest-path tree by using the DGS. This chapter ends with by making a comparison on the simulation results of both path-cost and non path-cost based schemes and making a concluding discussion.

Finally in Chapter 7, some conclusions are drawn with making reference to some possible future direction of the research.

Chapter 2

Wireless Sensor Network

Wireless Sensor Network (WSN) is a homogeneous or heterogeneous system formed mainly by energy constrained tiny sensors of same capability or different respectively facilitating large-scale real-time data gathering and processing environments. The gratitude and the accomplishment of WSNs in social and environmental vicinity are increasing day by day for the past few years. Comparing to other communication models in terms of easy configuration, free radio spectrum and low cost deployment, the WSN is a favorite choice to protect and monitor military, scientific, environmental, safety-critical, or domestic infrastructures and resources in both simple and complex environments.

Sensor nodes of a WSN are deployed to collect and analyze low-level data from an environment of interest. These data can be gathered for a various types of applications such as sensing temperature, pressure, sound etc. from a well or harsh environment. Based on the nature of applications, sensor nodes can be either mobile or immobile and can be placed randomly or deterministically based on the needs of applications and supports of environments. Some networks support new nodes to enter into it while the others do not. This chapter describes a brief overview and security of WSN, the success of which depends on the local cooperation among the nodes to transfer data to the base station for a desired time span.

2.1 Introduction

Wireless Sensor Networks (WSNs) have emerged as a new technology for monitoring inaccessible environments and events such as ocean and wildlife, earthquake, habitat in a jungle, weather forecasting and surveillance in military zones. This is an on-demand ad-hoc network formed by a collection of embedded sensors. Sensors can communicate with each other or with the base station using wireless media. The network using wireless sensor nodes is distinguished form the other classical networks due to strict limitation of energy consumption, the density of nodes during deployment and the simplicity of processing power.

The sensor node of a WSN has two activities: sensing the environment and sending information to the base station. Each node is typically equipped with a radio transceiver, a small microcontroller and an energy source, usually a battery. Based on the deployment for application, the size of sensor node differs by cost and power. The cost of a sensor node is usually ranging from few cents to a hundred dollar. The price of sensor node is also varied due to energy contents, memory capacity, computational speed and bandwidth used.

Due to the limited energy of battery used, most of the sensor nodes in a WSN cannot communicate directly with the base station. Moreover, even if possible, communicating directly to the base station can causes the sensor node to use up its energy very quickly, earlier before the expected life time. The sensor node can also not able to process the sensed data by itself due to limited processing power and saving the energy of the battery. The sensors nodes are therefore use multi-hop communications to send the sensed information, where the base station can use this data to extract desired information to fulfill the targets of deployment.

The security is one of the major drawbacks of WSNs. Since the sensors are tiny devices with containing limited energy and processing power, it is not possible to embed any security measures within the networks. The sensor nodes usually transmit raw information collected from the environment in plain text. High complexity encryption algorithms cannot be applied. Hence, the malicious users or devices get easy way to enter into the network for extracting or hacking information, transmitted among the sensors or from sensors to base station.

This chapter provides a brief overview of WSNs' architecture, power management, communications standards, protocols and security, followed by some interesting areas comprising of open research challenges faced while deploying the WSNs. Research problems for preserving energy using tree scheme is also discussed in this chapter.

The remaining of the chapter is organized as follows. Section 2.2 addresses the objectives behind deploying the WSNs and various applications where WSN plays vital role. The properties of WSNs are presented in section 2.3, while the section 2.4 provides a brief description of its architecture. The communication standards along with the protocols used in different communications layers of WSN is highlighted in section 2.5. Section 2.6 describes the concepts of power management, while the security issues are addressed in section 2.7. Section 2.8 compares WSN with other competitive classical wireless networks. The research challenges in the domain of WSN and in tree construction of sensor nodes are presented in section 2.9 and 2.10 respectively. Finally some concluding remarks are given is Section 2.11.

2.2 Objectives and Applications

The main objectives of WSN are to collect data from the environments for a given time instance, send it to the remote base station for processing and extracting knowledge. The extracted information then can be used for different types of decision making.

Figure 2.1: Sensor network applications in different domains in our life.

The sensing devices of WSN can sense different types of changes in the environment including acoustic, temperature, humidity, movement and light. These diverse sensing characteristics allow WSN to be used for a number of ways in different applications.

Some common applications are as follows: 1) environment monitoring such as a measurement of rainfall; 2) monitoring of impacts on bridges or buildings during the changes of earthquake vibration patterns; 3) surveillances such as for intruder and intruder and in battlefield; 4) monitoring ocean and wildlife where wired sensors would not be feasible; 5) monitoring of road and highway traffics; 6) health monitoring by using body sensor networks [48] in medical domain; and 7) characterizing the properties of different chemicals. Figure 2.1 summarize some of the applications of WSN in different domain of our practical life.

2.3 Properties of Sensor Networks

By networking large number of tiny sensor nodes, it is possible to obtain data about physical phenomena that was difficult or impossible to obtain in conventional ways. To achieve this target, the architecture and working principle of a WSN is designed in such a way that the following properties can be achieved:

- **Energy Constrained Network:** Since a large number of sensors are deployed in the critical environments where in most cases sensor nodes are invisible, especially when used for monitoring, sensor nodes are very tiny and have very limited energy. On the other hand WSN is deployed for a given time period which is usually large. Due to the presence of the environmental constraints, it is difficult or impossible to change the sensor nodes or batteries. Therefore, all steps of the working principle and protocols of WSN are designed with taking special care in every step to reduce energy consumption and optimize its effective use.

- **Sensor Data Types:** Sensor nodes collect different type of data periodically from the environment in the forms of temperature, pressure, sound and movement reading and so on.

- **In-Network Processing:** To save energy, the sensor nodes in a WSN form a tree structure, collected data from its child nodes, aggregate them and send them to the base stations.

- **Deployment Environment:** WSNs are often deployed in dispersed and/or remote geographic locations including harsh, hostile or widely scattered environments. Gathering of data from such an environment is very challenging.

- **Self Organized Characteristics:** In most cases, the communications protocols among the sensor nodes are designed in such a way that they can organize themselves without any external control, because in many cases manual configuration is typically not feasible when nodes are failed and new nodes area joined in the network. The self organizing characteristics help the network to continue to operate properly and support reliable connectivity under any converse circumstances.

- **Communications Types:** Some WSN allows a sensor node to communicate with other nodes within the signal reachable limit, while others support to send data to distant nodes via a number of intermediate ones.

2.4 Architecture

WSN is a reconfigurable ad-hoc network that can operate without the need of a fixed infrastructure. Each individual sensor node is intended to do some activities necessary to form the interconnected network, while meeting strict requirements of size, cost and power consumption to achieve the target network lifetime. A core challenge is to map the overall system requirements based on the device capabilities, requirements and actions. To use the WSN effectively, architecture of the network and sensor nodes must be analyzed for designing various applications under several hardware capabilities.

2.4.1 Network Architecture

There are two types of WSN: the single hop and the multi hop [3]. In a single-hop network, a sensor node can send data only to those sensor nodes within the reachable limit of its radio signal. The multi-hop WSN, on the other hand, allows sensor nods to send data in a longer distance by taking helps of some intermediate modes. The number of neighboring nodes within the coverage area of a single node can be determined either by maintaining the list of them or taking instance decision during the communications based on received signal strength [4, 71].

The sensor nodes of a WSN collect raw data from the environment and perform some signal level computations like modulation, analog to digital conversion and so on in order to signify them for application specific sensing. Nodes are typically be deployed densely in an area having a specific geographic interest and transmit their data either directly (single-hop) or through the other sensor nodes (multi-hop) to the WSN gateway. The gateway is responsible for transmitting data from the sensor nodes to the remote base station providing wide area network (WAN) connectivity and data logging for both the local and remote users. The gateway coordinates the activity within the sensor nodes and provide them necessary limited storage and computational support.

Each sensor node has a specific coverage area, beyond of which it cannot sense any changes as the nodes are limited by battery power and restricted to achieve a given lifetime. Hence, high spatial resolution of sensed information can be achieved only through deploying the sensor node densely. The computational module of sensor node is a programmable unit that provides computation, storage, and bidirectional communication with other nodes in the system. It interfaces with analog and digital sensors on the sensor module, performs basic signal processing, and dispatches the data according to the application’s needs.

Connections in a WSN are created in ad-hoc basis and there is no predetermined plan of providing connectivity to a specific sensor node or from the gateway to base station. In some cases, connections may fail due to natural calamities, ruins up of battery power and wireless signal absorption and fluctuations. This connectivity problem can be addressed and make the network robust by using redundant connection path. Each of the paths has different characteristics w.r.t. the expected robustness, bandwidth, energy efficiency, cost, and manageability.

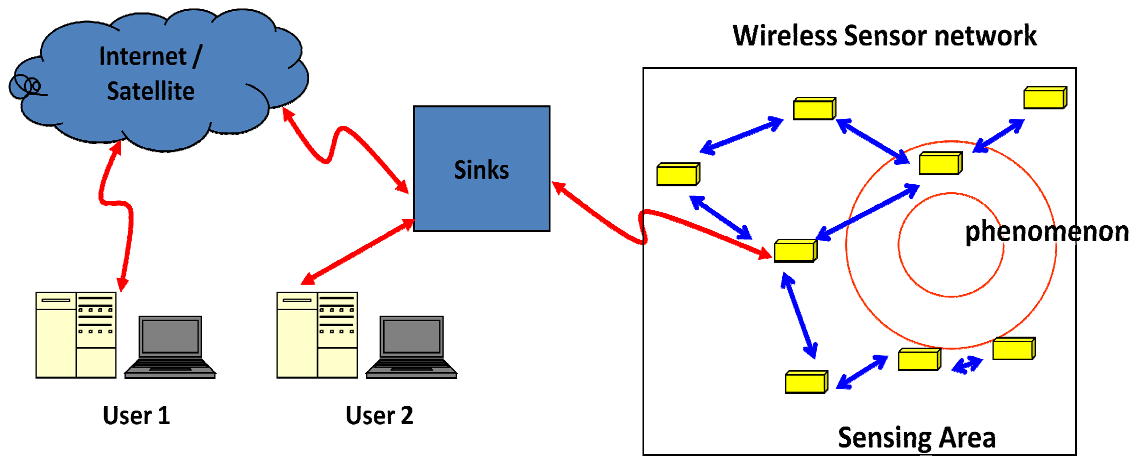


Figure 2.2: WSN architecture

Ultimately, data coming from sensors via gateway is to be propagated to the Internet as shown in Figure 2.2**.**The figure clearly shows data collected from the WSN is sent to the end user via the base station (Sink) and through the Internet. The propagated data from base station may be raw, filtered, or processed based on the architectures of the network. One of the main reasons of using base station is that it is not feasible to send data from sensor nodes or the gateway to the Internet as the process requires a large amount of power, computation and installation cost that cannot be supported. The sufficient power at base station and architecture allow it to participate in the WAN via using the optic fiber or point-to-point wireless communications.

The main task of data management and its processing is done at the base station as the sensor and the gateway cannot process them due to the limited power and computational capabilities. The base station uses different algorithms including association rule [6], fuzzy technique [57] etc. to extract the required knowledge form the sensor data stored in sensor data warehouse. The extracted knowledge is then transmitted to the base station through the public Internet in secured manner. In some cases, the base station contains raw data that can be pulled out to the user end on demand basis.

2.4.2 Sensor Node Architecture

The main task of a sensor node is to sense data from the environment and sends them to the gateway or the base station. In some cases, a sensor node is also capable of performing some pre-processing of data and communicating with other sensor nodes within its coverage range. To achieve these objectives, a sensor node is designed comprising of the following components: the microcontroller system, the transceiver system, the memory, the power source and one or more different types of sensors. The picture in the figure 2.3 show two typical sensor nodes produced by the crossbow company [60].The details of each part of the parts of a sensor node are given below.

Figure 2.3: Sensor nodes

2.4.2.1 Microcontroller System

The microcontroller system is the main computational and controlling part of a sensor node. This part processes data and controls the functionality of other components of a sensor node. The reason for choosing the microcontroller instead of similar devices like a general purpose desktop microprocessor, a digital signal processor, field programmable gate array or an application-specific integrated circuit is that the microcontroller has the flexibility to be connected to other devices more easily, consumes less power and one part of this device can go to sleep state while the other part is active. The general purpose microprocessor consumes more power. Digital signal processors are appropriate for broadband wireless communication and its processing task is complicated. The tiny sensors cannot therefore accommodate it despite of its high power signal processing capabilities. Such a demand also makes the field programmable gate arrays or application specific integrated circuit infeasible. The field programmable gate array can be reprogrammed and reconfigured according to requirements, but it takes time and power. Application specific integrated circuits, on the other hand, are specialized processors designed for a given application. Unlike the microcontroller, this device is bulky in size and performs its functionalities in hardware level.

2.4.2.2 Transceiver System

Sensor node uses the free radio spectrum known as industrial, scientific, medical (ISM) band which is easy to access. This radio spectrum is dedicated and kept free for research purposes. The communication in ISM band can be performed through the radio frequency (RF), optical communication i.e., laser and infrared. Laser requires less energy, but needs line of sight for communication. It is also sensitive to atmospheric conditions. Infrared is limited to its broadcasting capacity. RF based communication is the most relevant transmission media that fits to most of the WSN applications. WSN uses the communication frequencies of 900 MHz and 2.4GHz. The functionality of both transmitter and receiver are combined into a single device know as transceivers which is embedded in sensor nodes. There are four operational states of transceivers: transmits, receive, idle and sleep. Power consumption in idle mode is almost equal to power consumption in receive mode. Thus the transceiver remains in sleep mode when it is not transmitting or receiving. However, precaution is required before switching to sleep mode as significant amount of power is consumed for switching from the sleep mode to the transmit mode.

2.4.2.3 Memory

The memory of a sensor node is embedded within its chip called the on-chip memory. Some sensor nodes can accepts flash memory, which provides high storage capacity in a low cost. Memory requirement in a sensor node highly depends on the application for which it used. There are two types of memory used in a sensor node: a) user memory used for storing application related or personal data; and b) program memory used for programming the device.

2.4.2.4 Power Source

Power is consumed in a sensor node for sensing information from the environments, communication and data processing. Among them, data communication (sending and receiving) requires the maximum amount of energy**.** Power in a WSN stored either in batteries or in capacitors, of which using batteries are more popular. Two types of batteries used, are the chargeable and the non-rechargeable batteries. Batteries can also be classified according to electrochemical materials used for electrode such as NiCd (Nickel-Cadmium), NiZn (Nickel-Zinc), Nimh (Nickel metal hydride) and Lithium-Ion. Some of the recently developed sensors used batteries which are able to renew their energy from solar source or vibration.

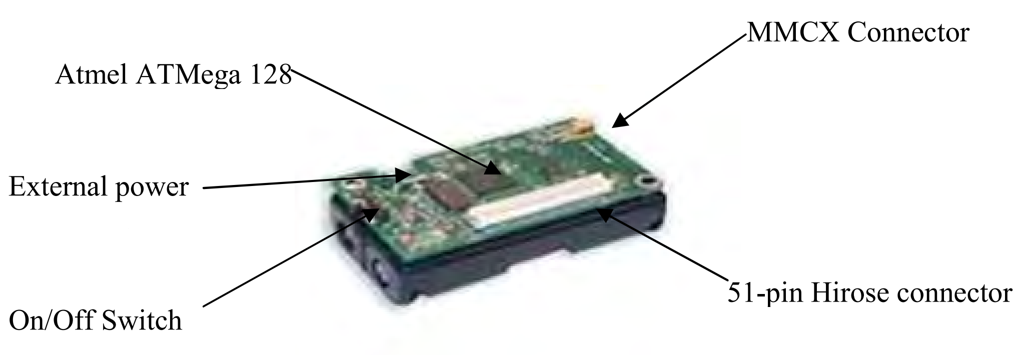


Figure 2.4: The Mica2 mote uses AA batteries to the CPU/radio

Since the sensors are tiny nodes, it is required to use tiny batteries having limited power, which is the main attention of designing a sensor node. For effective use of power, two major power saving policies used which are the dynamic power management (DPM) and the dynamic voltage scaling (DVS). DPM takes care of shutting down parts of sensor node which are not currently being used or active. DVS scheme varies the power levels depending on the non-deterministic workload. By varying the voltage along with the frequency, it is possible to obtain a significant reduction in power consumption. Figure 2.4 shows an example of sensor node which uses AA batteries as the power source.

2.4.2.5 Sensors

Sensors are hardware device that produce measureable response to a change in a physical condition like temperature, humidity and pressure. Sensors can sense or measure physical data of the area to be monitored. The continual analog signal is sensed by the sensor and is digitized using analog-to-digital converter. The digital information is then sent to microcontrollers for further processing.

There are several types of sensors which are embedded in a sensor board such as MTS 400/420 sensor board [13] includes temperature, humidity, barometric pressure, 2 axis accelerometer and ambient light sensor.

2.4.2.6 Operating System

Operating System for WSN is typically less complex than general-purpose operating system because of the special requirements and resource contains in hardware platforms. For example, as WSN applications are usually not interactive, the operating system does not require any support for user interfaces. Similarly, the resource constraints in WSNs make virtual memory implementation unnecessary.

Since the WSN hardware is not different from traditional embedded system such as microwave and calculator, it is therefore possible to use embedded operating systems such as eCos [14] or uC/OS [74] for WSN. TinyOS is the first operating system specifically designed for WSNs [76]. Unlike the other operating systems, TinyOS is based on an event-driven programming model instead of multithreading. TinyOS programs are compressed of event handlers and tasks management components. When an external event occurs, such as an incoming data packet or a sensor reading, TinyOS calls the appropriate event handler to handle the event. Event handlers can scheduled for some task in the kernel of a predefined time period. Both the TinyOS system and programs are written in a special programming language called nesC [75] which is an extension to the C programming language.

2.4.2.7 TinyDB

TinyDB [49] is a query processing system for extracting information from a network of sensors. Unlike existing solutions for data processing in TinyOS, TinyDB does not require to write embedded C code for sensor. Instead, TinyDB provides a simple, structured query language (SQL) like interface to specify the data to be extracted, along with additional parameters such as the rate at which data should be refreshed or collected. After specifying a query, TinyDB collects data from nodes in the environment, filters it, aggregates it together, and routes it out to a PC. TinyDB does this by applying power-efficient in network processing algorithms [65].

2.4.2.8 Types of Sensor Nodes and Specifications

There are different types of sensor-nodes (motes) exists in the market. The earlier node the "WeC" mote came in the market in the year 1998, which has the program memory 8KB, RAM 0.5KB, data rate 10Kbps and wake up time 1000 µs. The device does not have any built-in sensor and connect external one with it if necessary. In 1999, the "Rene" mote was manufactured with same specifications except the expansion facility. The variants of René came with higher capabilities in terms of program memory (16KB), Ram (1KB) and quick response time (36 µs) in 2000 and but still have lack of the integrated sensors. In the same year, the "Dot" mote came with the embedded sensors. The MICA motes and its variations having 128KB flash memory and 4 KB of data memory came to market in the year 2001 and 2002 respectively. The "Telos" mote came in the year 2004 with the 10-pic extension interface, for the first time, to perform communication through USB. Given the small memory sizes, writing software for motes is challenging. Ideally, programmers should be relieved from optimizing code at assembly level to keep code footprint small, while the high-level support and software services are not free. Being able to mix and match only necessary software components to support a particular application are essential to achieving a small footprint.

2.5 Communications Standards and Protocols

There are a number of communication standards and protocols designed by IEEE and other authorities. The standards should be followed while designing the communication architecture, policies and protocols. Due to using tiny devices with limited battery power in WSN, the communications standards and protocol used in nodes are differ significantly from the existing ones. The details of communications standards and protocols used in WSN are described below.

2.5.1 Communication Standards

IEEE is one of the leading standards-making organizations in the world providing standards in a wide range of industries including: power and energy, biomedical and healthcare, information technology, telecommunications, transportation, nanotechnology, information assurance, and many more. In 2005, IEEE had closed 900 active standards, with 5000 standards under development. One of the more notable IEEE standards is the IEEE 802 local area network (LAN)/ metropolitan area network (MAN) group of standards which includes the IEEE 802.3 Ethernet standard and the IEEE 802.11 Wireless Networking standard. There are several communication standards devised by IEEE, which may fit for WSN [32]. Some of the notable standards used in WSN are mentioned below.

2.5.1.1 IEEE 802.15

The 802.15 wireless personal area networks (WPAN) effort focuses on the development of consensus standards for short distance wireless network. These WPANs address wireless networking of portable and mobile computing devices such as personal computers (PCs), personal digital assistants (PDAs), peripherals, cell phones, pagers, and consumer electronics; allowing these devices to communicate and interoperate with one another. The goal is to publish standards, recommended practices, or guides that have board market applicability and deal effectively with the issues of coexistence and interoperability with other wired and wireless networking solutions. Sensor nodes can also be deployed to from WSN as personal area networks.

2.5.1.2 IEEE 802.15.4

The IEEE 802.15.4 was chartered to investigate a low data rate solution with multi-month to multi-year battery life and very low complexity. It is operating in an unlicensed, international frequency band. Potential applications are sensors, interactive toys, smart badges, remote controls, and home automation.

2.5.1.3 IEEE 802.15.5

The IEEE 802.15.5 is chartered to determine the necessary mechanism that must be present in the physical and medium access control (MAC) layers of WPANs to enable mesh networking. A mesh network is a personal area network (PAN) that employs one of the two connection arrangements, full mesh topology or partial mesh topology. In the fell mesh topology, each node is connected directly to each of the others. In the partial mesh topology, some nodes are connected to all the others, but some of the nodes are connected only to those other nodes with which they exchange the most of the data.

2.5.1.4 IEEE 1451

IEEE 1451 [27] is a set of smart transducer interface standards developed by the IEEE instrumentation and measurement society's sensor technology technical committee. The standard described a set of open, common, network-independent communication interfaces for connecting transducers (sensors or actuators) to microprocessors, instrumentation systems, and control/field networks. One of the key elements of these standards is the definition of transducers electronic data sheet (TEDS) for each transducer. The TEDS, in essence, is a memory device attached to the transducers, which stores transducer identification, calibration, correction data and manufacturer-related information. The goal of the IEEE 1451 family of standards is to allow the access of transducer data through a common set of interfaces whether the transducers are connected to systems or networks via a wired or wireless means.

2.5.2 Frequencies used in WSN

Low-power, short-range and low-data-rate WSNs use the unlicensed free RF band [12]. Network designers need to first determine which sub-band is best for their application: 900 MHz or 2.4 GHz. Both the frequency bands reside in the ISM band and among the unlicensed frequency bands of 900 to 928 MHz, 2.4 to 2.489GHz. Almost all of the transceivers of WSN applications use the 900-928 MHZ and 2.4 -2.483GHz frequency bands. The 900 MHz band has long broadcast range because of its relatively longer wavelength. It covers large area and for that reason power consumption is not high. This makes battery life higher. The 2.4 GHz band has small broadcast range because of its relatively lower wavelength. Before going to use any spectrum, at first it needs to check whether the frequency band is licensed or unlicensed. For example, in Europe 900 to 928 MHz band is not used because it is part of the global system for mobile communications (GSM) network for cell-phone communication and, thus, it is unavailable. In Bangladesh, this frequency band is also occupied by telecommunication companies.

2.5.3 Layered Architecture and Protocols

Layering is a common abstraction in network design. Layering provides design modularity for network protocols that facilitates standardization and implementation. The Internet has driven the actual implementation of layering, which is built around the transport control protocol (TCP) for the transport layer and the internet protocol (IP) for routing at the network layer. There are five layers in TCP/IP protocols which are physical layer, link layer, network layer, transport layer and application layer. WSNs use these five layers model but its architecture does not use the IP protocol for routing, since routing through a WSN is different than that in the internet. The addressing and sub-netting mechanism in the IP protocol is not well suited and all the mechanism used in TCP is not necessary to transport the data in WSN.

2.5.3.1 Physical Layer

The physical layer deals primarily with transmitting bits over a point to point wireless link. The design tradeoffs associated with the physical layer includes modulation, coding, diversity, adaptive techniques, multiple input multiple output (MIMO), multi carrier modulation and spread spectrum. Among these issues, the choice of a good modulation scheme is critical for reliable communication in a WSN. Binary and M-ray modulation schemes discussed in [66] are energy efficient and hence most suitable to use in WSNs.

2.5.3.2 Link Layer

The data link layer is responsible for the multiplexing of data streams, data frames detection, medium access and error control. It ensures reliable point-to-point and point-to-multipoint connections in a communication in a communication network. In self organizing medium access control for sensor networks (SMACS) and eavesdrop and register (EAR) model [68]. Sensor nodes are mostly stationary and there exist a number of higher energy mobile nodes. Network startup and link layer organization for the sensor nodes is done by SMACS, combining neighborhood discovery and channel assignment phases to form a connected network. A communication link consists of a pair of time slots operating at a randomly chosen but fixed frequency. This is a feasible option in WSNs, since the available bandwidth is much higher than the maximum data rate for sensor nodes. The scheme preserves power by random wake up during setup of the network and after time slot allocation to the neighboring nodes by turning radio off while idle. The EAR protocols attempts to offer continuous services to the mobile nodes under both mobile and stationary conditions. Here, the mobile nodes assume full control of the connection process and also decide when to drop connections, thereby minimizing messaging overhead.

The carrier sense multiple access (CSMA) based MAC protocol [82] is able to process highly correlated flows of data. This scheme uses constant listening periods for energy efficiency and introduces random delays for robustness. In order to achieve fairness [50], an adaptive rate control scheme is used.

In [66], hybrid time division multiple access-frequency division multiple access (hybrid TDMA-FDMA) is shown to be more energy efficient that time division multiple access (TDMA) or frequency division multiple access (FDMA). This work emphasizes that energy efficient protocol for WSNs cannot be designed unless physical layer and hardware issues are taken into account.

2.5.3.3 Network Layer

The network layer is responsible for establishing and maintaining end to end connections in the network. Tree construction is one of the important tasks of network layer to organize the nodes in the network to transmit information through neighbors rather than sending information directly. The other functions of network layer in WSN are neighbor discovery, routing and dynamic resource allocation. Neighbor discovery [71] is the process by which a node discovers its neighbors when it first enters the network. Routing, the process of determining how packets are directed through the network from their source to the destination, is another key function of the network layer. In WSN, routing is performed through intermediate nodes by relaying. Dynamic resource allocation dictates how network resources such as power and bandwidth are allocated throughout the network, although in general resource allocation occurs in multiple layers of protocol stack. Small minimum energy communication network (SMRCS) [63] computers an energy-efficient sub-network or sub-graph, namely the minimum energy communication network (MECN). A new algorithm called small MECN (SMECN) is proposed in [42] to construct more energy efficient sub-network. This scheme creates a sub-graph of the WSN that contains the minimum energy path. Flooding, an old and the simplest routing protocol broadcasting data to all neighbor nodes, can also be used in WSN, but suffers from serious deficiencies such as implosion, overlap and resource blindness [22]. The Gossiping scheme [19] sends data to one of the randomly selected neighbors. This scheme avoids implosion problem but message propagation takes longer time. In sensor protocols for information via negotiation (SPIM) scheme [22], whenever a node has available data, it broadcasts a description of the data and sends it only to the sensor nodes that express interest. Low energy adaptive clustering hierarchy (LEACH) [21] forms a two level clustering hierarchy, where cluster members send data to the cluster head which then in turn sends it to the base station. Energy dissipation is evenly spread by dissolving clusters at regular intervals and randomly choosing the cluster heads. In the directed diffusion scheme [33], base station sends out an interest of accepting data for processing which propagates in the network and sets up gradients for data to flow from source to base station. Ad-hoc on demand routing protocol (AODC) [30, 56] dynamic source routing (DSR) [28, 35] or a secure on demand routing protocol (ARIADNE) [29] can also be used for routing in WSN.

2.5.3.4 Transport Layer

The transport layer provides the end to end functions of error recovery, retransmission and reordering and flow control. The transport layer provides an extra measure of protection by monitoring for corrupt or lost packets on the end to end route and requesting a retransmission from the original source node if a packet is determined to be lost or corrupted. The transport layer serves to order packets transmitted over an end to end route before passing them to the application layer.

2.5.3.5 Application Layer

The application layer generates the data to be sent and processes the corresponding data received over the network. This layer provides compression of the application data with error correction and concealment. The compression techniques may be lossless or loss for different types of WSN applications [61] and energy efficient.

2.6 Power Management

In most of the application scenarios of WSN, replacement of sensor nodes or batteries might be impossible or infeasible for the cost. Hence, it is required to make the sensor node alive as much as can. Moreover, each node plays the dual role of data originator and data router in WSN. The failure of a few nodes can cause significant topological changes and might require rerouting of packets and reorganization of the network. Hence, power conservation and power management to prolong the battery lifetime of sensor node is getting additional importance. For these reasons, researchers are currently focusing on the design of power-aware protocols and algorithms for WSNs [1].

2.6.1 Power Problem of Sensor Nodes

The main task of a sensor node in a sensor field is to detect events, perform quick local data processing and then transmit the data. Power consumption can hence be divided into three domains: sensing, communication, and data processing. The energy level of the batteries is very small (0.5Ah 1.2 V) relative with its power consumption as shown in Figure 2.5.

Thus, it is important to consider the impact of energy constrains in the design of WSN. Sensor nodes with rechargeable batteries must preserve energy for the time period before recharging. In many applications, sensor devices cannot be recharged as they are deployed in embedding inside the walls or dropped into a remote region. Such nodes must operate for years solely based on the battery energy provided during the initial phase. The μ-AMPS and Picoradio projects are aimed at developing radios for these applications that can operate on the power less than 100 microwatts and exploit energy-harvesting to prolong lifetime [10, 62].

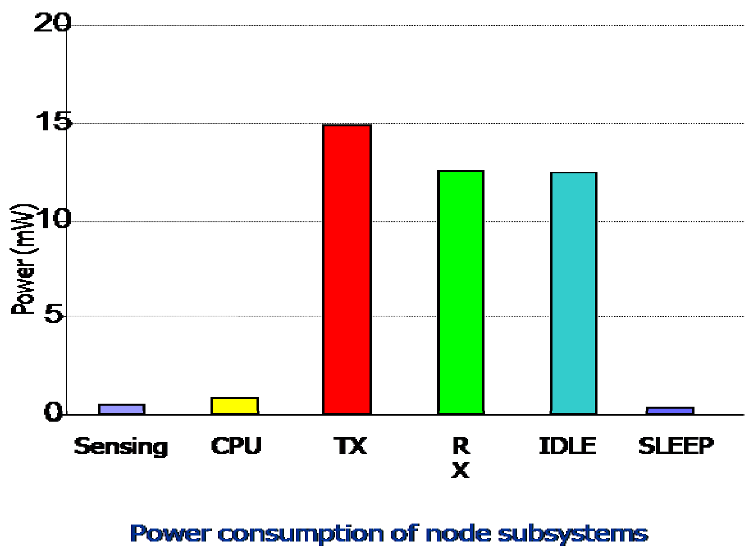


Figure 2.5: Typical energy consumption for different activities of sensor nodes

2.6.2 Power Preserving Issues

Energy constraints of WSN impact the hardware operations, transmitting power, and the signal processing. The required transmit energy-per-bit error rate (BER) target in a noisy channel is required to be minimized. Transmit power is not only factor in power consumption. The signal processing associated with packet transmission and reception, and even hardware operation in a standby mode, consume non negligible power as well [1, 20]. This entails energy tradeoffs between transmit power and BER across communication layers. At the physical layers, many communication techniques that reduce transmit power requires a significant amount of signal processing. It is widely assumed that the energy required for this processing is small and continues to decrease with ongoing improvements in hardware technology [38]. However, the results in [1, 20] suggest that these energy costs are still significant. Therefore, energy-constrained systems developed for energy-efficient processing techniques need to minimize power requirements across all levels of the protocol stack and to minimize message passing for network control is well. Sleep mode for nodes must be similarly optimized, since this mode requires standby energy for other protocol layers due to associated access and routing. The hardware and operating system design for the node should also be in such a way that the energy consumption is reduced. Some of these optimization techniques are described in [1, 9]. In summary it can be stated that energy constrains impact all layers of the protocol stacks, and hence make cross-layer design important for meeting the performance requirements [16, 17 and 69].

2.6.2.1 Tree Construction

Tree construction is the grouping of sensor nodes with hierarchical structure to gather data from the environment in WNS. Here sensor nodes collected data from the environment and send it to its upper level node call parent. The parent node collect data coming from its child nodes and aggregate with its own data. Tree structure plays a vital role in organizing sensor network topology and offers tremendous benefits for WSNs to preserve power. Network designers must carefully examine the formation of tree in the network while designing a particular application. Depending on the application, certain requirements like number of nodes, network density etc. may play an important role for tree construction.

2.6.2.2 Query Processing

One particular interesting approach is to regard the WSN as an entire database and to interact with it via database queries. The challenges are in finding energy-efficient ways of executing such queries and of defining proper query languages that can express the full richness of WSNs.

2.6.2.3 Energy Efficient Routing

Optimal routing in WSN is not practically feasible because it requires future knowledge [64] about the networks. However, the requirements for energy efficient routing can be simplified towards considering over all possible future communications of the network. Several routing protocols [21, 64] considered the effective consumption for communications in WSN.

2.6.2.4 Efficient Node Placement

Some sensor applications allow nodes to be deployed using predetermined method. There are several research works [47, 70] for efficient node placement to achieve the desired network lifetime. Efficient node placement is required to get the full advantage of using sensor nodes to collect data from environment and reduces power consumption of the nodes for the minimum number of communications with base station.

2.6.2.5 Node Admission Control

Malfunctioning or dying up of a node degrades the performance of WSN as resources such as time, energy and information are lost when they arrive into a non-functional node. Node admission control deals with admission and replacement of sensor nodes [5, 34, 52, and 53] with the inclusion of security mechanisms to cope with dynamic membership in terms of important security primitives (such as key management) and services (such as secure routing) without the assistance of any centralized trusted authority. Since WSNs are often composed of tiny or resource-limited devices, admission control policy must be efficient in terms of computation and communication.

2.7 Security of Wireless Sensor Networks

The limited data storage and computational power of WSN are the major obstacles to the implementation of traditional computer security techniques. The unreliable communication channel and unattended operation, i.e. providing activities towards the sleeping mode make the security defenses even harder. The industrial trend is to reduce the cost of wireless sensors while maintaining similar computing power. With that in mind, many researchers have begun to address the challenges of maximizing the processing capabilities and energy reservations of wireless sensor nodes while securing them from the common attackers [58]. The main aspects of WSN security are divided into three major categories: the obstacles to WSN security, attackers and defensive measures.

2.7.1 Obstacles to WSN Security

A sensor node has many constraints [7] which limits the security measures. A sensor node is a tiny device with only a small amount of memory and storage space for the code. In order to build an effective security mechanism, it is necessary to limit the code size of the security algorithm. Moreover, the extra power consumed by sensor nodes due to security is related to the processing required for security functions (e.g. encryption, decryption, signing data, verifying signatures), the energy required to transmit the security related data or overhead (e.g., initialization vectors needed for encryption/ decryption), and the energy required to store security parameters in a secure manner (e.g., cryptographic key storage).

Unreliable communication is another threat to sensor security. Normally the packet-based routing of the WSN is connectionless and thus inherently unreliable. Packets may get damaged due to communication channel errors or congestion. The result is lost or missing packets. Furthermore, the unreliable wireless communication channel also results in damage packets. Higher error rate in the communication channel also forces the software developer to devote resources to error handling. More importantly, it the protocol lacks the appropriate error handling it is also possible to lose critical security packets.

2.7.2 Attacks

WSNs are particularly vulnerable to several key types of attacks [83]. Attacks can be performed in a variety of ways, most notably as denial of service (DoS) attack, but also through traffic analysis, privacy violation, physical attacks, and so on. DoS attacks on WSNs can range from simply jamming the sensor's communication channel to more sophisticated attacks in different layers.

A standard attack on WSNs is simply to jam a node or set of nodes. Jamming, in this case, is simply the transmission of radio signal that interface with the radio frequencies being used by the WSN. The jamming of a network can come in two forms: constant jamming and intermittent jamming. Constant jamming involves the complete jamming of the entire network and no messages are able to be sent or received. If the jamming is only intermittent, then nodes are able to exchange message periodically, but not consistently. This too can have a detrimental impact on the WSN as the message being exchanged between nodes may be time sensitive.

Attacks can also be made on the link layer itself. One possibility is that can attacker may continually transmit messages in an attempt to generate collisions. Such collisions would require the retransmission of any packet affected by the collision. Using this technique it would be possible for an attacker to simply deplete a sensor node's power supply by forcing too many retransmissions.

At the networking layer, a node may take advantage of a multi-hop network by simply refusing to route message. This could be done intermittently or constantly with the net result being that any neighbor who routes through the malicious node will be unable to exchange messages with, at least, part of the network.

The transport layer is also vulnerable to attack, as in the case of flooding. Flooding can be as simple as sending many connection requests to a sensor node. In this case, resources must be allocated to handle the connection request. Eventually a node's resources will be exhausted, thus rendering the node futile.

2.7.3 Defensive Measures

Each layer is vulnerable to different DoS attacks and has different options available for its defense. Some attacks crosscut multiple layers or exploit interactions between them. Table 1 lists the attacks at different layers of WSN and defensive methods.

Layer	Attacks	Defenses
Physical	Jamming	Spread-spectrum, priority message, lowers duty cycle, region mapping and mode change.
	Tampering	Tamper proofing, hiding
Link	Collision	Error correcting code
	Jamming	Region Mapping
	Exhaustion	Rate limitation
	Unfairness	Small frames
Network and Routing	Neglect and greed	Redundancy
	Homing	Encryption
	Misdirection	Egress filtering, authorization and monitoring
	Black Holes	Authentication, redundancy and probing
Transport	Flooding	Client puzzles
	De-synchronization	Authentication

Table 1: Several attacks in different layers of WSN and defensive methods

2.8 Comparative Analysis

Other that WSN, different types of closely related wireless models [18] such as cellular network, wireless local area network (WLANs) and wireless ad-hoc data networks are also widely used now-a-days. WSNs vary from other types of related networks by a number of points. These are addressed below.

2.8.1 Difference with Cellular System Networks

Cellular network is a wireless network having a unique architecture which differentiates it from others. Cellular network is implemented mainly for voice communication and it is getting popularity for data transmission as well now-a-days. Some of the differences between cellular network and WSN are mentioned below:

- **Frequency Spectrum:** Cellular network uses a broad frequency spectrum; it needs to acquire a license from the government to use the spectrum. However, WSN uses the free RF spectrum and in most of the cases it uses a single frequency for data transmission and collection.
- **Layering:** Cellular network does not use the same layered model that is used in WSN.

2.8.2 Difference with Wireless Local Area Networks

A WLAN links two or more computers or devices using spread-spectrum, enabling communication between devices in a limited area. This gives user the mobility to move around within the coverage area and still be connected to the network. For the home user, WLAN has become popular due to ease of installation and location freedom with the gaining popularity of laptops. Some of the differences between WLAN and WSN are,

- **Architecture:** WLAN uses authorized devices such as a router which routes the packets to their respected destination. There are also other devices such as gateway and access points based on the functions they perform. There is no such high power device in WSN. However, these responsibilities are encapsulated in the nodes of this network. Such as for routing each node acts as a potential router.
- **Radio Spectrum:** Like WSN, WLAN uses the free spectrum but it uses multiple frequencies for data transmission and other job it performs. Intentional jamming in wireless network is not easy for an attacker because it is difficult to know at once on which frequency the network is transmitting. Since to acquire that know information the attacker needs some time to acquire that information, meanwhile the network may choose to use a different frequency for transmission.

2.8.3 Difference with Wireless Ad-hoc Networks

A category of wireless ad-hoc network is the WSN. WSNs have some characteristics common to ad-hoc wireless networks and have some own characteristics. Some of the distinct features of WSN are,

- **Communications:** Ad-hoc networks typically support routing between any pair of nodes whereas WSNs have a more specialize communication pattern. Most traffic in WSN can be classified into one of three categories:

  Many-to-One: Multiple sensor nodes send sensor reading to a base station or aggregation point in the network.

  One-to-Many: A single node (typically a base station) multicasts or floods a query or control information to several sensor nodes.

  Local Communication: Neighboring nodes send localized message to discover and coordinate with each other. A node may broadcast message intended to be received by all neighboring nodes or uncast messages intended for single neighbor.

- **Resources:** Nodes in ad-hoc networks have generally been considered to have limited resources, but sensor nodes are more energy constrained. After deployment, battery recharging may be infeasible or impossible for most of the applications of WSN.

- **Trust Level:** Nodes in WSNs often exhibit trust relationship. If each node sends a packet to the base station in response, precious energy and bandwidth are wasted. For pruning this redundant message to reduce traffic and save energy, WSNs require in-network processing, aggregation, and duplication elimination. This often necessitates trust relationships between nodes that are not typically assumed in other types of ad-hoc networks.

- **Type of Application:** In wireless ad-hoc network, nodes are mobile. Nodes enter and leave the network frequently. They often change their position in the network. However, in WSN, nodes are topologically static for most of the applications.

- **Components of the Network:** In wireless ad-hoc network the nodes are mainly general purpose device such as laptop, PDA where as WSN mainly consists of special autonomous independent nodes, known as motes.

2.9 Research Challenges

Considerable work and effort has focused on designing communication protocols for WSNs. However, no single protocol has emerged as a major contender and research on this issue is very much active and ongoing. WSNs are currently receiving significant attention due to their unlimited potential. However, it is still very immature and many research challenges exist.

2.9.1 Collaborative information Gathering

The main task of a WSN is to collect information from the environment and send to the base station. Cooperation among the nodes [2] reduces the collisions among the packets sent from various sources and reduces energy consumption. There are some processing tasks that can be done by the sensor node itself (such as data aggregation) which saves time. Since the WSN architecture is not robust, collaborative information gathering process must be energy efficient. Hence, it's a challenging are of research.

2.9.2 Development of a New Transport Protocol

New transport schemes need to be introducing which will focus on energy efficiency and take advantage of the collaborating nature of WSNs. For example, one strategy would be to design a transport protocol that can receive feedback about a variety of attributes (energy state of the nodes, quality of wireless link, etc) and use the information in other layer as necessary.

2.9.3 Development of a New Routing Protocol

Disseminating information in WSNs with tight energy constraints is still an open problem and there is a need for routing protocols specifically tailored for ultra low energy and asymmetric communication with realistic assumptions about the rate of topology changes and the number of nodes in the network. Quite a few routing protocols have been proposed [44, 51, 25, 87, 84, 79, 37, 39, 78, 31, 82, 86, 26] but most of them need to be improved because they assume mostly static topologies and small number of nodes.

2.9.4 Wireless Sensor Deployment

WSNs when deployed in a real-world setting, fail to meet application requirements even though they have been tested in the lab prior to deployment [8]. Many WSNs have been reported to deliver only between 40% and 80% of the sensor data they are expected to produce. The reasons for such failures can be for many reasons, including hardware problems (e.g., condensing humidity may cause short circuits in a sensor), software bugs (e.g., communicating nodes fails to wake up concurrently due to excessive clock drift caused by extreme temperature variations). Unfortunately, these problems are often not encountered during pre deployment tests, because the environmental conditions that trigger these problems are hard to simulate in the lab. Hence, it is required to reveal the reasons of the problems. On the other hand, sensor node deployment issues [47, 70] for the optimal coverage is required more analysis to achieve the desired network lifetime.

2.9.5 Security

Due to the feeble infrastructure of the WSN and the type of applications, security is a major concern in this area in terms of easy means of attacks. There are many works [83] dealing with principal threats and possible attacks for the correct functioning of WSNs but still require a detail analysis and energy efficient defensive mechanisms. Hence, there is still a wide open field for research.

2.9.6 Cross Layer Design

Cross layer design requires that the interdependencies between layers are characterized, exploited and jointly optimized. Cross layer design clearly requires information exchange between layers; adapt the information at each layer. The unique attributes of energy-constrained networks make them prime candidates for cross-layer design. If node batteries cannot be recharged, then each node can only transmit a finite number of bits before it dies, after which time it is no longer available to perform its intended function (e.g. sensing) or to participate in network activities such as routing. Thus, energy must be used judiciously across all layers of the protocol stack to prolong network lifetime and meet the requirements of the application. Energy-efficiency at all layers of the protocol stack typically imposes tradeoffs between energy consumption, delay and throughput. However, at any given layer, the optimal operating point on this tradeoff curve must be driven by considerations at higher layers. For example, if a node transmits slowly it preservers transmit energy, but this complicated access for the other nodes and increases end-to-end delay. A routing protocol may use a centrally located node for energy-efficient routing, but this will increase congestion and delay on that route, as well as burn up that node's battery power quickly, thereby removing it from the network. Ultimately the tradeoffs between energy, delay, throughput and node/network lifetime must be optimized relative to the application requirements. Hence, energy efficient cross layer design is an interesting area of WSN to explore for the researchers.

2.10 Motivation of Research on Tree Based Scheme

WSN is mainly deployed to collect data from the environments. The data gathering process can be interrupted by a number of ways due to the nature of applications. On the other hand, energy efficiency must be considered in WSN due to energy limitations of the sensor nodes. To achieve the energy efficiency in limited resources three different data gathering methodology; cluster-based, chain-based, tree-based are available in WSN. Though the cluster-based methodology gains its popularity by ensuring the energy efficiency, it introduces some new problems. In cluster-based schemes the distribution of the cluster head is not symmetric and some cluster-head die-out before the round-time of the network. Also the size of the cluster is difficult to control in cluster-based schemes increase the energy loss per round in sensor nodes. In chain-based schemes, die-out an intermediate node break-up the path of the chain and increase the data loss of the network. Also, longer chain increases the delay of the network which is not effective in delay bounded applications. On the other hand, tree based scheme achieves the goal by efficiently constructing a load balancing tree. Here, intermediate nodes collect the data from their descendent nodes and aggregate the collected data with its own data which also reduce the size of the data packet i.e.

traffic of the network. The leaf nodes in the tree conserve their energy by periodically going to the sleeping mode. Since, tree structure is simple and minimum graph structure, tree based schemes plays a significant role in WSN in terms of efficient data gathering for a desired time period by sensing data from the environments. Therefore, it requires attention to the energy efficient tree based schemes [46, 51, 25, 87, 84, 79] to explore the problems and need a solutions to construct energy efficient load balancing tree effectively.

2.11 Summary and Comments

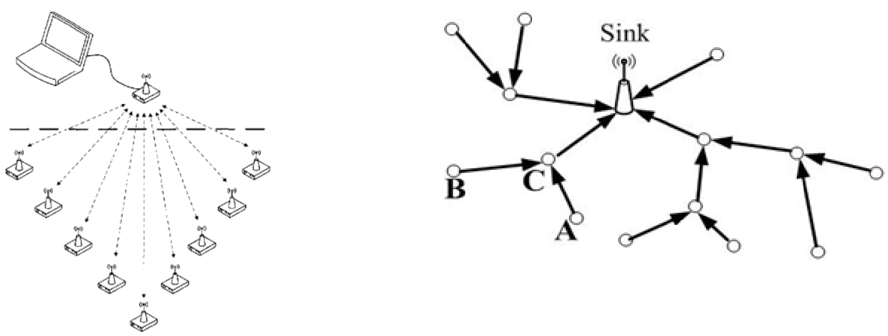
Wireless sensor networks (WSNs) are more than just a specific form of ad-hoc networks. The necessity for the economic use of energy and low computational power make the WSNs different from other networks. The technology of WSN is still immature for practical deployment in general. Energy efficient load balancing tree construction based on the residual energy of the nodes is the most challenging areas of research. This chapter discusses some basic concepts of the emerging network technology having significant importance in the next generation communications, monitoring and investing applications.

Data Gathering Tree

Energy is the main obstacle for deploying of Wireless Sensor Networks (WSNs) in practical application as tiny sensor nodes cannot accommodate sufficient energy required for achieving the desire level of usage. Efficient utilization of sensor node's energy is, therefore, the most demanding research challenge now-a-days. The tree based routing protocols attain the acme popularity among the researches due to making effective usage of residual energy preserved in sensor nodes through implementing the concept that the leaf nodes use less energy than the intermediate nodes in communication phase and preserve their energy by periodically going to the sleeping mode. Also, the intermediate nodes can reduce their child degree and the size of the data packet by aggregating collected data with its own data which reduce the communication cost and total traffic of the network. Researchers are, therefore, get motivations from this energy saving communication scheme and extend it in various dimensions. This chapter narrates the pros and cons of different energy saving tree based protocols to create motivations towards innovating new schemes for bringing the sensor network technology well ahead.

3.1 Introduction

For the immense importance of tree based protocols, many research works [44, 51, 25, 87, 84, 79, 37, 39, 78, 31, 82, 86, 26] have proposed different tree based protocols by optimally use of energy to collect data from environment for a desired time span. Figure 3.1(a) depicts an application where sensor nodes periodically transmit information to the base station without forming tree. Figure 3.1(b) shows the tree based scheme where the intermediate node C collects data from its child nodes A and B, aggregates the collected data with its own sensed data and finally sends it to the neighbor nodes. The process continues until the collected data reach to the base station. The main problem on non-tree based algorithms are: i) message flooding is obvious when density of nodes becomes higher; ii) the packets coming from different nodes may be dropped due to the collision at the base station; and ii) the nodes which are at distant place from the base station die quickly due to energy loss.



(a) (b)

Figure 3.1: Data gathering in WSN, a) without tree construction b) with tree construction

Tree structure improves the network lifetime by, i) reducing energy loss of the nodes by multi-hop communication, ii) intermediate nodes aggregate data coming from their child nodes which reduce the size of the data packets, iii) leaf nodes reduce their energy consumption by periodically going to the sleeping mode, and iv) reduce the collision among the data coming from sensor nodes to the base station.

Various tree based data gathering algorithms [44, 51, 25, 87, 84, 79, 37, 39, 78, 31, 82, 86, 26] have been proposed in the WSNs. During tree construction, the algorithms focus on the issues of residual energy, path-cost, load of the nodes, node degree and the delay. Since the construction of energy efficient data gathering tree is NP-hard problem, different schemes point the different issues during tree construction. Here, most of the schemes construct an energy efficient data gathering tree by considering the residual energy of the nodes along with the path-cost. However the load of the sensor nodes become imbalanced and some nodes become heavy loaded and die-out earlier in those algorithms. To overcome those problems, some other algorithms consider the load of nodes along with the residual energy and construct a load balancing tree. However, they are not applicable in real-time applications and the heights of the tree are also hard to control. Some of other load balancing tree based algorithms with limited height are also proposed but they do not give any guarantee to survive specific round-time. Considering the issues discussed above, all the algorithms organize into two distinct categories: *non path-cost and path-cost based data gathering tree*. The non path-cost based algorithms focus on the load of the sensor nodes along with residual energy during tree construction and the neighbor cost of all nodes are considered as equal. On the other hand, the path-cost based algorithms considered the cost of the path as an energy consuming factor along with the load of the nodes to construct a load balancing tree.

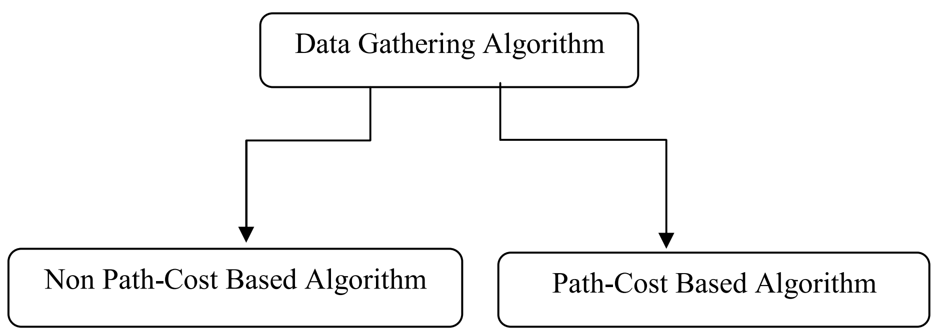


Figure 3.2: Types of data gathering algorithms

The remaining of this chapter is organized as follows. An overview of the different non path-cost based and path-cost based data gathering tree based algorithms are discussed in the Section 3.2 and Section 3.3 with their pros and cons. Finally some concluding remarks are given in Section 3.4 which motivate to construct an energy efficient load balancing tree, will be discussed in next chapter.

3.2 Non Path-Cost Based Data Gathering Tree

Delay Bounded Minimum Degree Spanning Tree (DB-MDST) [39], Centralize Lifetime Maximizing Tree (CLMT) [78], Energy Aware Tree Routing (ETR) [31], Delay Constrain and Maximum Lifetime Data Gathering (DCML)[44], Energy-Aware Maximal Leaf Nodes Data Gathering Tree (EMLN-DG) [51], are well known non path-cost based data gathering algorithms. The brief discussions about those algorithms are given below:

3.2.1 Delay Bounded Minimum Degree Spanning Tree (DB-MDST)

DB-MDST proposed by Kown S. el at [39], constructs an energy efficient data gathering tree with limited delay. As for energy efficiency, the algorithm focuses the issues of overall energy consumptions and per-node fairness of the network. This algorithm solves the problem to find the energy efficient spanning tree by constructing a minimum a spanning tree with limited height. For per-node fairness, the scheme focuses on the most overburdened nodes and minimizes their degree to remove them from overburdened.

The algorithm focuses on two properties of the tree structure, the height and the maximum degree. The efforts of the two properties on the energy and delay performance are as: i) energy consumption of the nodes is proportional to its degree, ii) node's waiting time for its sub-tree data is proportional to the number of its descendent. Therefore, delay is mainly propositional to the height of the tree.

The algorithm minimizes the maximum degree while maintaining the tree height below predefined limit. To construct a tree the algorithm uses the approximated MDST [87] algorithm which iteratively removes the edge from the maximum degree vertex and adding an appropriate edge to maintaining the connectivity. DB-MDST maintains the height of the tree below a predefined limit by using a height limiting factor during the edge adding part of the MDST iteration.

Figure 3.3 shows a comparative tree structure for the given topology shown in Figure 3.3(a). In Figure 3.3(b), the shortest path tree (SPT) scheme constructs a tree with maximum node degree 9 and height 6. In Figure 3.3(c), MDST constructs a tree with maximum node degree 3 and height 32. In Figure 3.3 (d), DB-MDST constructs a tree with the maximum degree 4 and limited the height 6.

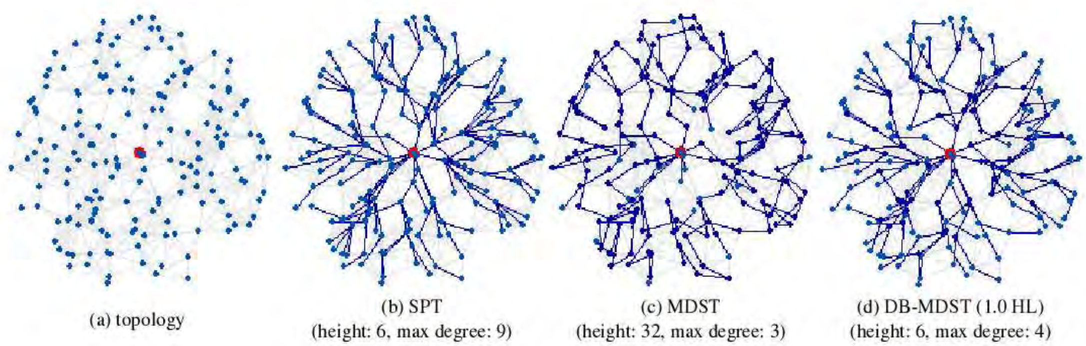


Figure3.3: DB-MST tree

Though the algorithm constructs a tree with limited height, the loads of the nodes are difficult to balance. Different nodes have different residual energy and their load balancing factors are also different, i.e. intermediate nodes support different node degree to balance their load. Also, the algorithm does not consider the path-costs during tree construction.

3.2.2 Centralize Lifetime Maximizing Tree (CLMT)

The CLMT proposed by Satbir J. et al [78] is a centralized algorithm creates a maximum lifetime data gathering tree, spanning the entire nodes as vertex and takes care that no loop is created. All the parent nodes in the tree are selected in such a way that they will have maximal-available energy resource to receive data from their children and the minimum energy node is only used to send data to their parent for data aggregation. Such aggregation will extend the life time of the tree and decrease the data loss due to a broken link before the tree reconstruction.

Primarily, CLMT stores all the nodes in a list with their residual energy. Then it selects a least-energy node, removes it from the list and tests that if the removal of all network links to this node except from its highest-energy neighbor disconnects the existing graph. If so, the removal node is declared as a bottleneck node and there is no better way than to collect data by this node. Therefore, the removed links are thus restored. If the above mention condition is not satisfied, the removed links do not contribute to the construction of the CLMT and permanently drop the removed link and move to the next node. The energy of the candidate parent neighbor has to be greater than that of the node under the test. When such neighbor does not exist, the node has to be a parent for at least one of its neighbors, and thus the links are preserved and the lifetime is maximized. In that case when more than one highest-energy node has equal energy, node ID is used to break the tie. The processes will continue until a single node exists in the list.

The algorithm is good enough to increase the lifetime of the network but does not give any direction to avoid or removal of bottleneck node in the network. Also the scheme considers the bottleneck nodes are those nodes that do not have any alternate path. However, in tree based scheme bottleneck nodes are those whose residual energy is not enough to survive a specific round-time for a particular load. Also, the algorithm does not consider the issue of path-cost during tree construction and the height of the tree is difficult to control.

3.2.3 Energy Aware Tree Routing (ETR)

The Energy-aware Tree Routing (ETR) proposed by Kim B. et al [31], is a distributed algorithm related to reliable and energy efficient data routing by selecting a data transmission path in corresponding of residual energy at each node to disperse energy consumption across the network and reliably transmit the data through a detour path when there is a link or node failure.

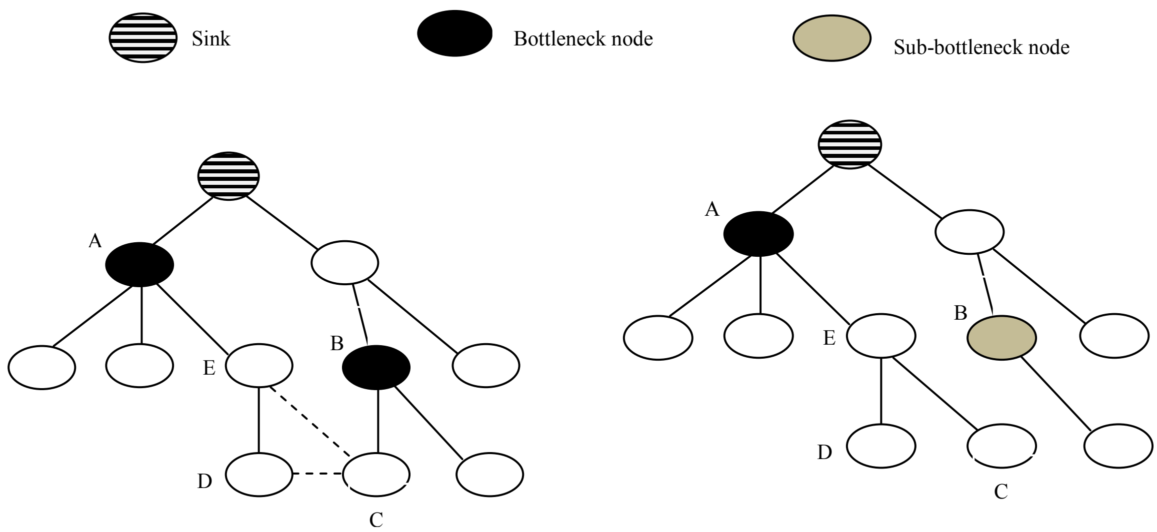
Initially, sink node with level zero initiate the route setup phase by broadcasting a setup message to its neighbors with node ID and become the parent of all its neighbors. When a node receives request from more than one node then it selects a node with highest residual energy as its parent and store all other node’s information in its candidate parent-node table. Each sensor node broadcasts a route setup message including its own address and the level which is set to {parent_level +1}. This operation repeat until all the nodes in the network respectively set their own level. When a node does not receive a route setup message or a new node joins the network, it waits for predefine listen interval to be assigned a level. If a nodes is assigns a level within this period, it broadcast a rout setup request message and then selects a parent and store candidate parent information with the replies of its neighbor. Once a node sets it level, the node is no longer receives any setup message from its neighbor.

Since the algorithm is distributed, the sensor nodes collect their neighbor information from the acknowledge message. The parent node sends an acknowledgement message with its own address and residual energy. If a node does not receive acknowledgement in a predefine time period it resends data. For a certain number of failures, the node then reselects its parent.

Though the algorithm constructs a minimum spanning tree and provides the facility to add new node in the network, the load of the nodes is imbalanced. Also it does not consider the path-cost during tree construction and extra setup packets increase the traffic of the network and decreases the network lifetime.

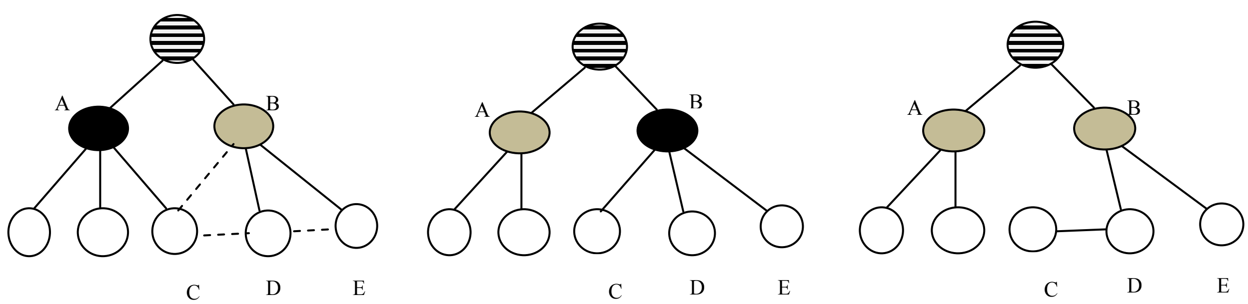
3.2.4 Delay Constrain and Maximum Lifetime Data Gathering (DCML)

The DCML proposed by Liang J. et al [44], is a delay sensitive and durative surveillance data gathering scheme with limited height. Primarily, DCML constructs a fewest hope tree (FHT), by using Dijkstra algorithm where the distance of the nodes from the root is minimum. The algorithm selects the highest residual energy node as the root with constant edge cost during tree construction. Then the algorithm divides the nodes into three disjoint set as: bottleneck nodes, sub-bottleneck nodes and rich nodes by comparing their round-time with the network life time. Here, bottleneck nodes do not survive same round-time as the network lifetime where the rich nodes have the enough energy to survive specific round-time. Rich nodes do not become bottleneck even when the node degree is increased by one. On the other hand, sub bottleneck-nodes become bottleneck if their child degree is increased by one. After categorization, the algorithm performs the optimization operation to remove the load of the bottleneck nodes. Two categories of optimization: direct and indirect optimization is used by this algorithm during optimization.



(a) Initial Tree (b) Optimize Tree

Figure 3.4: Direct optimization



(a) Initial Tree (b) Unsuccessful Optimization (c) Successful Optimization

Figure 3.5: Indirect optimization

In Figure 3.4(a), node B is the bottleneck node. When node C, child node of B removed and connected with node E, node B relives from bottleneck and becomes sub-bottleneck which is shown in Figure 3.4(b) and it is called direct optimization. In Figure 3.5(a), node A is the bottleneck and B is the sub-bottleneck node. If we remove node C from node A and add it to node B, then sub-bottleneck node B becomes bottleneck which is shown in Figure 3.5(b). However, if we connect the node C to D, both node A and B are relives from bottleneck and it is called indirect optimization.

To balance the load of the nodes after tree construction is a weak problem and height of the tree is also hard to control in this algorithm. Also, the transmission-cost and path cost is the vital issues related to the lifetime of the network.

3.2.5 Energy Aware Maximum Leaf Nodes Data Gathering Tree (EMLN-DG)

The EMLN-DG scheme proposed by Meghanathan N. et al [51], is the tree based scheme which uses to periodic data collection and transmission in wireless sensor networks. For each round of data gathering, an EMLN-DG tree spans the entire sensor network based on the residual energy available at the nodes and the number of uncovered neighbors of the node during the tree formation. Only the nodes having a relatively larger number of neighbors as well as higher energy level are include as an intermediate node. Nodes having the lower energy become leaf nodes and conserve their limited energy by periodically going to the sleeping mode. By maximizing the number of leaf nodes in a tree and considering the energy level available at the nodes, while forming the tree, this algorithm reduces energy consumption per round as well as balances the energy level across all the nodes in the network.

Initially, EMLN-DG calculates the weight of all the nodes in the network and selects the highest weighted node as the root. The weight of a node is the product of the number of uncovered neighbors (the neighbors that are not connected to the network) and the node's residual energy. For two or more nodes with same weight, it randomly chooses a node as the root and adds it into the tree. Root is the highest weighted node in the tree and all its uncovered neighbors are added to the tree by connecting with the root. The new connected nodes are added into a list called covered list.

EMLN-DG recalculates the weight of the nodes in the list by updating their neighbors list. It selects a highest weighted node from the list and the selected node becomes intermediate node by connecting all it uncovers neighbors. The newly covered nodes that are not in the covered-list are added into the covered list. Then the scheme updates the neighbor list of covered nodes and re-calculates their weight. Select the next highest weighted node and the processes continue until all the nodes in the network are covered.

This algorithm constructs a maximum leaf node data gathering tree by decreasing the average distance from the root. It also reduces the energy consumption of the nodes and combines more packets together and lightens the overall traffic of the network.

The EMLN-DG scheme, however, is restrictive from a number of limitations:

1. Though the algorithm constructs a minimum spanning tree, the loads of the intermediate node are not balanced and lower energy nodes have the probability to be intermediate node and become overburden (bottleneck), die-out earlier in the network.
2. The transmission cost of the node is also a vital issue related to the lifetime of the node. Longer distance causes larger transmission-cost and decreases the lifetime of the nodes.
3. Die-out of the bottleneck node causes packet loss and decreases the throughput of the network.
4. This scheme does not guarantee any alternate paths which are related to network reliability. Hence, when intermediate nodes die-out in the path, all the data along this path are also lost.

3.3 Path-Cost Based Data Gathering Tree

The Power Efficient Data Gathering and Aggregation (PEDAP) [25], Tree Structure Based Data Gathering for Maximum Lifetime (MLDGA) [88], Maximum-Lifetime Data Gathering Tree (ML-DG) [84], Data Gathering Routing Algorithm Based on Energy Level (DG-EL) [79] and Energy-Efficient Routing Algorithm to Prolong Lifetime (ERAPL) [89], are well known data gathering tree, consider path-cost along with residual energy of the nodes. Some of those algorithms also consider the load of the nodes during tree construction to construct a balanced tree. Brief discussions about those algorithms are given below:

3.3.1 Power Efficient Data Gathering and Aggregation (PEDAP)

The PEDAP proposed by Ho T. et.al [25] suggest two different schemes: PEDAP and Power Aware version of PEDAP (PEDAP-PA) to construct an energy efficient data gathering tree. Both the schemes are centralized and the base station is responsible for computing the routing information. It also assumes that the locations of all the nodes are known by base station and base station balances the communication load of the system by considering the resource of the nodes.

The algorithm computed the routing information using Prim's minimum spanning tree scheme. Initially, the base station is the root in the new constructing tree. After that, in each iteration, it selects a minimum weighted edge (minimum cost path) from a set of nodes which are not in the tree and add that edge to the tree i.e. the node that is included in the tree will send its data through that edge. The algorithm repeats this process until all nodes are added to the tree.

Computing a minimum spanning tree over the given network using the cost functions and by routing the data packets according to that spanning tree, this algorithm achieves a minimum energy consuming system. Besides knowing the locations of the nodes, the base station can also estimate the remaining energy levels of the nodes by using the cost model, since it knows how much energy a node spends in a round. After a certain number of rounds passed, the base station re-computes the routing information excluding the dead nodes. The based station then sends the required information (i.e. the node's parent in the tree in order to reach the base station; the time slot number when the node will send its data to its parent in a round; from how many different neighbors the node will receive packets in a round and when, etc.) for that node. Hence, the cost of setting-up the system with the newly routing information is equal to only the sum of costs of running the receiver circuitry of each node.

Here, the protocols are same for the both proposed algorithm but only thing that must be changed is the cost function. Therefore, switching between the proposed algorithms requires only a small change in cost function of the base station.

PEDAP and PEDAP-PA algorithm suffer from some major problems:

1. During tree construction the algorithm only considers the path-cost.
2. Due to the die-out of the nodes in the network, the dense network become sparse after each phase and every node requires more energy to send the data to their neighbor which decreases the lifetime of the nodes.
3. Though the algorithm does not consider the load of the nodes, it constructs an imbalanced tree in the network.
4. In each repeated setup phase, sink needs to broadcast the information to all the nodes in the network which causes extra setup cost.

3.3.2 Tree Structure Based Data Gathering for Maximum Lifetime (MLDGA)

The MLDGA proposed by Zhang Q. et al[88], constructs a routing tree for each round where the geographical position of the nodes are known to sink and every nodes have different residual energy. Initially, all the nodes in the network are the member of a list called candidate list. The highest energy node of the candidate list is connected to the tree T as a root and is also added into a new set called connected set. The scheme calculates the weight of the edges on between the connected and candidate set nodes. It selects the highest weighted edge and adds it into the tree which also adds candidate node of the edge's in the connected set. Here, weight of the edge is defined as the ratio of the edge lifetime (edge life time is the minimum lifetime between two nodes that are connected by the edge) and the path cost. A lifetime estimation technique is used to determine the lifetime of the nodes. From this edge calculation, two points that fit with the strategy about the maximum lifetime data gathering problem: i) if the transmission cost between the sending and receiving node is low, the edge weight becomes high and the sensor has the higher possibility to add to the network, which meets the strategy to minimize the total energy consumption in each round, ii) if the lifetime of the edge is high, the edge weight tends to be high which adopts a heuristic greedy strategy that the higher the weight of an edge, the higher the possibility of the edge to be added to the tree. The process continues until all the nodes of candidate set are the member of connected set i.e. connected to the tree.

The MLDGA scheme also experiences some limitations:

1. During the life calculation of the nodes this algorithm does not consider the actual load of the nodes which are depended on their path-cost and node-degree.
2. If two nodes in an edge have higher energy difference, the lower energy node limits the higher energy node's probability to be intermediate node.
3. The algorithm does not focus on the delay during tree construction.
4. The algorithm does not guarantee any alternate path which decreases the reliability of the network.

3.3.3 Maximum-Lifetime Data Gathering Tree (ML-DG)

The ML-DG proposed by Fahmy S. et al [84], starts from a arbitrary tree and then reduces the load of overburden node. Here, overburden nodes are those nodes that likely to soon deplete their energy due to either higher degree or low remaining energy. The algorithm terminates at polynomial time and gets the optimal data gathering tree.

It gives a mathematical model to calculate the lifetime of nodes and converts the optimum data gathering tree problem into minimum degree spanning tree (MDST) problem, i.e. finding a spanning tree where maximum degree is minimum among the spanning tree.

Initially, the algorithm constructs an arbitrary tree by considering the path-cost and divides the nodes into three distinct categories as: bottleneck node, sub-bottleneck node and safe node. The bottleneck nodes in the tree die-out earlier before the lifetime of the network and the safe nodes alive a specific round. Sub-bottleneck nodes become bottleneck if we increase their child degree one. The algorithm then reduces the degree of bottleneck node using a single iteration and then the iterative approximate technique.

Though the algorithm constructs a energy efficient load balancing data gathering tree, the height of the tree are hard to control in this algorithm which increase the delay of the network. Also load balancing operation causes extra cost of the network which also decreases the lifetime of nodes.

3.3.4 Data Gathering Routing Algorithm Based on Energy Level (DG-EL)

The DG-EL proposed by Wang Z. et al [79], is the cost effective multicast routing algorithm, builds the shortest path tree by enhancing the link sharing between source and a set of destination. It considers the path-cost of the nodes along with the residual energy to construct a tree to prolong the lifetime of the networks and enlarge number of data packets received by the processing terminal.

The algorithm uses, energy-level of the path and a comparative parameter (ELDP) of residual energy and path-cost during tree construction. The energy level of the path is the minimum energy load of the node along the path. A node calculated its load by dividing the residual energy with the neighbor cost. On the other hand, ELDP is the compromising statistics between the energy level of a node and the total-path cost from the sink. Higher energy-level nodes have higher ELDP value than its neighbor. On the other hand, all nearest nodes from the root get the higher priority than a node farthest from the root though they have the same residual energy.

Initially, DG-EL connects a highest energy node as a root for the tree. Then all the neighbors of the root node get a connection request and join the tree as a leaf node of root. The new added leaf nodes calculate their energy-level and ELDP value and send a connection request to their neighbors. If a node receives more than one request, it selects its parent by considering the ELDP value of the requested node. In the case of equality, energy-level and path cost are used to break the ties. This algorithm also uses an energy coefficient which allows the energy level to fluctuate in an appropriate boundary.

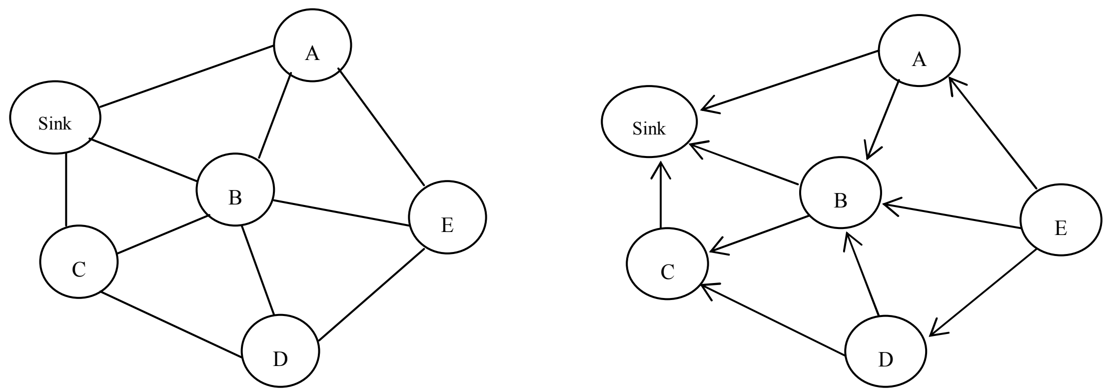
Considering the residual energy and energy level of the path this algorithm constructs a data gathering tree where higher residual energy with minimum distance node becomes intermediate and increases the life time of the path i.e. the life time of the network. It also considers total path-cost during tree construction which ensures a minimum energy lost per round during data collection.

The DG-EL scheme also suffers from some major problems:

1. In this scheme, the height of the tree is hard to control in this scheme which increases the overall delay of the network.
2. The algorithm only considers the residual energy along with the path-cost. However, child-degree of the nodes, i.e. the number of descendent node is also effect the load of the intermediate node.
3. The nearest lower energy node from the root has the higher possibility to become intermediate node. In such scenario, some nodes become bottleneck (loaded) and die-out before the network lifetime, increase the packet loss of the network.

3.3.5 Energy-Efficient Routing Algorithm to Prolong Lifetime (ERAPL)

The ERAPL proposed by Zhu Yi. et al [89], is a centralized algorithm which runs at the sink and constructs an energy-efficient data gather tree where the location of node is known to the sink. It also solves the mutual and loop transmission problem among the sensor nodes. The first problem, mutual transmission, is in the case where node A sends a packet to node B but node B sends the packet back to A directly; where the latter (loop transmission) is the case in where node A sends a packet to node B, but the packet is at least sent back to A after it is transmitted to multiple node. These two problems generate un-necessary packets in the network and increase the loss of node's energy. The ERAPL algorithm uses the following three steps to construct a energy-efficient data gathering tree: i) constructs a Data Gathering Sequence (DGS), ii) uses a mathematical model to calculate outgoing traffic proposition (OTP), whose objective function incorporate minimal remaining energy and total energy consumption of the nodes and iii) uses generic algorithms (GAs) [26] with compressed chromosome coding scheme to find the optimal OTP matrix.



(a) (b)

Figure 3.6: A network scenario, a) without DGS b) with DGS

A DGS, denoted as $Seq(i_1, i_2, i_2, \dots \dots, i_n)$, is the ordered listed nodes stating from node i_1 and ending with node i_n as $i_1 \rightarrow i_2 \rightarrow i_3 \rightarrow \cdots \rightarrow i_n$, where i_n is the sink of the network. In DGS, a node is allowed to transmit data to its neighbor nodes in the forward direction rather than backward direction.

Figure 3.6(b) shows a DGS, where every node is directed in forward direction, i.e. directed towards sink node. Here, a node can receive data from their predecessor nodes and send data to their successor nodes.

For a given graph G, the key of DGS-construction algorithm is to repeatedly perform the following two steps until all the nodes are added to the DGS: 1) appends all the leaf nodes (node with degree one) to the DGS and then removes these nodes from the graph, 2) appends the nodes, which has the maximum distance (Euclid distance) from the sink, and removes the node from the graph if there is no leaf node.

After constructing DGS, the algorithm uses a mathematical model to calculate the outgoing traffic proposition (OTP) of every node by considering their sending and receiving cost and construct an OTP matrix. Here OTP depend on sending and receiving cost of data and receiving cost depends on the number of child nodes. The OTP matrix is used to calculate minimal remaining energy of all the nodes and the total energy consumption of the network. Minimize the energy consumption of the network; this algorithm uses generic algorithms (GAs) which determine the optimal OTP matrix. Based on this optimal OTP matrix the algorithm construct an energy efficient data gathering tree with limited delay.

The algorithm also has some major limitation as:

1. It faces some overhead during exchanging message between the sink and the nodes to determine the optimal OTP.
2. This algorithm uses Euclid distance between the nodes and the root during DGS construction. Though the WSNs are the multi-hop communication based network, Euclid distance from the root does not always ensure minimum cost path in the network. Two nodes with minimum Euclid distance can have the larger path cost.

3.4 Summery and Comments

Different tree based algorithms make the wireless sensor network (WSN) technology robust and bring it to a standardized level for deploying it in practical applications for a specific network lifetime. Despite all of these, all the scheme suffer from a number of limitations as discussed in previous sections of this chapter that are yet to be solved. As the demand of WSN deployment in different spheres of our modern life in the era of information technology, the solution of all these problems become in extreme demand now-a-days. The application of WSNs will surely be overwhelming in near future if those problems can be solved to make the WSN technology rigorous.

Chapter 4

Simulation Setup

The acceptance of a proposed technique depends on the comparative performance analysis compared with the existing schemes. This chapter presents the detail of experimental setup of the proposed and existing schemes. Considering the time and cost factors, simulation is an attractive alternative to experiments of WSN applications. Several sensor network simulator are currently available [67, 15, 36, 40, 41, 54, 55, 59, 77] for simulation of WSN applications. The experiments are carried out using the popular process based discrete event simulation package SimJava [67] which uses Java platform and developed by University of Edinburgh. A SimJava simulation is a collection of entities which run in its own thread. These entities are connected together by the port and can communicate with each other by sending and receiving event objects. A central system class controls all the threads, advances the simulation time and delivers the events. The processes of the simulation are recorded through the trace messages by the entities and save it in the file. There are some references [80, 81] which used SimJava to implement the simulation environment in WSNs. For investigation the clear picture of performance improvements and each of the components anticipated in, the proposed scheme is studied separately with the existing ones, and then the overall performance of the proposed scheme is studied against the existing algorithm. The results of experiments prove clearly the superiority of each component of the proposed scheme and the overall proposed scheme itself.

4.1 Introduction

In our experiment, first we simulate the proposed non-path cost based scheme EBA-DG and then compare its simulation result with the simulation results of existing EML-DG. During the simulation of EBA-DG, the simulator uses a mathematical model for load calculation and then categorizes the nodes based on their residual energy and energy requirements before tree construction.

For the path-cost based scheme EBASP-DG, the simulation results are compared with the results of existing path-cost based scheme DG-EL based on the performance parameter. In the simulation of EBASP-DG, at the very beginning the simulator constructs a path-cost based data gathering sequence (DGS) by using proposed DGS construction technique. Then it calculates the load of the nodes and categorizes them based on their residual energy and energy requirements, and finally simulates the EBASP-DG.

This chapter proceeds with a introduction in Section 4.1. Section 4.2 discusses about the simulator (SimJava) and set the value of simulation parameters. Section 4.3 briefly discusses the performance parameters that are used to compare the simulation results. This chapter ends with a conclusion in Section 4.4.

4.2. Simulation Environment

This section gives a brief idea about the simulation platform and the architecture of SimJava. Some important parameters used in the experiment are also listed in this section.

4.2.1 SimJava

SimJava is a discrete event, process oriented simulation package. It is an API that arguments Java with building block for defining and running simulations. The original SimJava was based on

HASE++ and C++ simulation library. HASE++ was in turn based on Jade's SIM++. It comes with some animation facilities.

The approach to simulating systems adopted by SimJava is similar to other process based simulation package. Each system is considered to be a set of interconnecting processes or entities as they are referred to in SimJava. These entities are connected together by port and communicate with each other by passing event. A central system class controls all the threads, advances the simulation time and delivers the event. The process of the simulation is recorded through trace message produced by the entities and saved in the file.

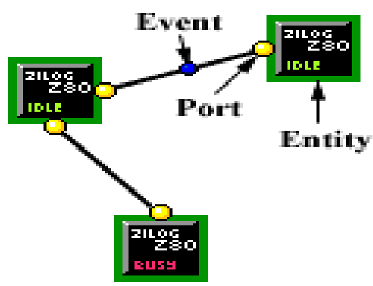


Figure 4.1: A simulation layout

SimJava has also been regimented with considerable statistical and reporting support. The modeler has the ability to add detailed statistical measurement to the simulation's entities and perform output analysis to test the quality of collected data. Furthermore, much offer has gone into the automation of all possible tasks allowing the modeler to focus on the pure modeling aspects of the simulation. Automated tasks range from seeding the random number generators used in the simulation, to producing detailed and interactive graphs.

4.2.2. Simulation Parameters

Several experiments were carried out to justify that the proposed schemes increase the round-time of overburden nodes which boost the overall lifetime of the network. The schemes also decrease the packet dropping rate by increasing the average lifetime of the intermediate nodes and limiting the tree height. Some of the noted simulation parameters are listed in Table 2 and the values of most of the parameters that are mentioned here used by [85].

Parameters	**Values**
Number of Nodes N	300 nodes ~ 500 nodes
Initial Energy of the Sensor Nodes $E_r(v_i)$	80J ~ 200J
Transmission Electronics E_e	50nJ
Transmission Amplifier ε_{mp}	100pJ
Data Packet size l	2000bits
Aggregation Cost $E_{ax}(v_i)$	5nJ
Node Deployment	Random
Simulation Area	500 × 500

Table 2: Simulation parameters

4.3 Performance Parameters

WSN uses some performance indicators to compare the performance of the existing and proposed tree based schemes. This section discusses some of the most common performance parameters; energy loss per round, network lifetime, average lifetime of the bottleneck nodes, throughput, tree height; used to compare the performance of proposed scheme with existing others. All the parameters are used to compare the results of both path-cost and non path-cost based schemes except the parameter path-cost which only used in path-cost based schemes. Detailed descriptions of those parameters are discussed below.

1. **Energy Loss per Round:** The sensor is battery operated non replaceable node responsible to sense, collect and then aggregate the data before sending to the neighbors' nodes. Rendering energy is very scarce resource that must be wisely managed in order to extend the lifetime of the network. The energy loss per round is the sum of total energy loss of all nodes in a round-time during data communication. We have to extend the lifetime of the node to minimize the energy loss.
2. **Network Lifetime:** The lifetime of the network is defined as the round-time: the first node or a percentage of nodes die-out in the network. In tree based scheme, die-out of leaf nodes do not break the path of the network and do not cause any data loss. Therefore, in tree based scheme, the network lifetime is defined as the amount of time till the first intermediate node die-out in the network.
3. **Throughput:** In tree based scheme sense data must need to deliver to a single base station i.e. the root. The root is the higher energy node gather data that are generated by the nodes in the network. Throughput is defined as the amount of data packet collected by the root in its round-time or the lifetime-time of the network. However, in tree based scheme the traffic load of critical (loaded) node is heavy and die-out before the round-time of the network causes packet loss and decreases the throughput of the network**.**
4. **Average Lifetime of the Bottleneck Nodes:** In tree based scheme some lower energy nodes become overburden and died earlier in the network causes packet and energy loss of the network. This thesis proposed two tree based schemes to balance the load of the overburden nodes by decreasing their child-degree and transmission-cost. The experimental results compare the proposed scheme with existing others based on the average round-time of the bottleneck nodes to prove the correctness of the proposed theatrical model.

5. **Height of the Tree:** In tree based algorithm, the height of the tree is the maximum hop count from leaf to root and entail the delay of the network. It is hard to control the height of the tree in tree based schemes. However, in delay guaranteed service the data need be delivered to the user within a certain delay. Hence, the proposed algorithm should be constructed in such a way that will limit the tree height.
6. **Path-cost:** Communication cost of the nodes depends on its transmission-cost and sum of the transmission-cost of the nodes from root to node is the path-cost of the nodes. Higher path-cost increases the overall communication cost and decreases the lifetime of the network. To construct a load balancing tree both transmission-cost and path-cost should be kept minimum. In this thesis, we considered both transmission-cost (local path-cost) and path-cost (global path-cost) to decrease the communication cost of the nodes.
7. **Average Lifetime of the Nodes:** Average lifetime of the nodes are the average round-time of nodes (rich, bottleneck and poor) in the network. In tree based scheme some lower energy nodes become overburden and die-out earlier in the network. To construct a load balancing tree, we have to balance the load of overburden nodes which minimize the lifetime variation of all types of nodes.

4.4 Summery and Comments

The simulation parameters mentioned previously have been used in the next two chapters to simulate the non path-cost and path-cost based schemes. The performance parameters are used to compare the simulation results of both proposed and existing path-cost, non path-cost schemes. The next chapter proposes a non path-cost based scheme EBA-DG and compares the simulation results with existing EML-DG scheme. Chapters 6 propose the path-cost based scheme EBASP-DG and then compare the simulation results with existing DG-EL scheme.

Chapter 5

Proposed Energy Aware Bottleneck Nodes Avoidance DG

Now-a-days a well designed tree based scheme is the crucial need in achieving the expected lifetime and to obtain the required facilities for Wireless Sensor Networks (WSNs). Although some tree based schemes gain their popularity to provide claimed energy efficient communication protocols for WSN up to certain extent, yet they suffer from a number of points including the intermediate node selection, load balancing of intermediate nodes and path-cost, which result the wasting of energy and increase data loss of the network. This chapter describes a new scheme named "Energy-aware bottleneck-node avoidance data gathering tree (EBA-DG)" to meet the expected lifetime and balance the energy distribution based on the residual energy of the nodes. The proposed scheme ensures the expected lifetime of the networks and reduces the wastage of energy and information by efficiently balancing the load of the overburden intermediate nodes to sustain a specific round. With the above mentioned modification, this scheme can easily be claimed to be the matured tree based technique for energy efficient communications in WSN.

5.1 Introduction

Gathering data from environment and sending it to the base station is the challenging research issue in WSN. Lots of researches are still going on in the area of energy efficient tree construction to extend the lifetime of the networks. So, it's required to participate in the establishment of a standard technique having good service quality and energy efficiency to meet the goal of a desired time span.

Chapter 3 discusses different non path-cost based schemes thoroughly with the identification of some interesting problems which are needed to be solved immediately. The DB-MDST [39] constructs a spanning tree with the limited height and then removes the degree of the overburden nodes to construct energy efficient load balancing tree. However, this scheme does not focus on the residual energy and path-cost of the nodes during tree construction. On the other hand, CLMT [78] considers the residual energy of the nodes and constructs a tree ensuring that the lower energy nodes will be leaf node expect those that split the network. In CLMT, the higher energy nodes become overburden and larger transmission-cost increases the path-cost of the network.

In ETR [31] the sensor nodes select their parent by considering the residual energy and hops count. The algorithm starts from the root and constructs a minimum spanning tree. It also gives the opportunity to add a new node in the tree. However, extra setup packets increase the overall traffic of the network, which decrease the lifetime of the nodes. Also the loads of the intermediate nodes are not balanced and it is difficult to control the height during tree construction.

DCML [44] constructs a fewest hope tree (FHT) with limited height and categorizes the nodes into three distinct categories as: bottleneck, sub-bottleneck and rich node. Bottleneck nodes are the loaded nodes, die-out earlier in the network where rich nodes have sufficient energy to survive specific round-time. The sub-bottleneck nodes become bottleneck if DCML increases their node degree one. The scheme decreases the load of bottleneck node by removing one or more of its child nodes using direct and indirect optimization technique. The optimization is done in such a way that does not introduce any new bottleneck node. However, in DCML, it is difficult to control the height of the tree and the intermediate nodes do not ensure to survive specific round-time.

Residual energy of the node and uncovered neighbors are the key elements to construct a data gathering tree in EMLN-DG [51]. Higher residual energy node with larger node-degree increases the node's probability to be added in the tree as an intermediate node. This algorithm increases the number of descendent of intermediate nodes; combined more packets together and decreases the energy consumption of the network. However, the nodes having lower residual energy nodes with higher node-degree also have the opportunity to become intermediate nodes; die-out within the round-time and cause data loss of the network. The network also has the possibility to become disconnected during the round-time of the network.

Those major problems mentioned above are considered for analysis and then provide a solution in this thesis. The contribution of this thesis are: i) categorizing the nodes using a mathematical model which considers both node-degree and path-cost along with residual energy of the nodes, ii) a tree construction by ensuring that intermediate nodes will be survive specific round-time by balancing their node degree, ii) constructing a path-cost based data gathering sequence (DGS) which enhances the probability to construct an energy-based shortest-path tree with limited height and increase the effectiveness of load calculation and, vi) constructing a path-cost based data gathering tree by balancing the load of the overburden nodes to survive specific round-time.

The rest of this chapter proceeds as follows. Section 5.2 presents the motivations of the proposed scheme. Section 5.3 discusses a mathematical model for load (round) calculation of the sensor nodes and Section 5.4 categorizes the nodes based on their load and residual energy. A new scheme named "Energy-aware bottleneck-node avoidance data gathering tree (EBA-DG)" is discusses in Section 5.5. Section 5.6 shows the simulation results and makes a comparative analysis with the existing EML-DG scheme. This chapter ends with by making a concluding discussion in Section 5.7.

5.2 Motivation of the Proposed Scheme

A load balancing data gathering tree for maximizing network lifetime is an immensely demanding part of research in energy constrains WSNs. The proposed bottleneck node avoidance data gathering algorithm aims to achieve a significant extension in the network lifetime and decreases the packet loss by extending the lifetime of the overburden nodes. The motivations of the proposed scheme are:

1. In order to lengthen the network lifetime, the energy level of the nodes and their load must be defined. In tree based schemes, both the higher residual energy loaded nodes and lower residual energy nodes die-out before the expected lifetime of the network.
2. Lifetime of the nodes is inversely propositional to its node degree and transmission cost. Nodes with higher degree are more loaded than the lower one and die-out before the round-time of the network. Hence, during tree construction the node degree need to be kept modest considering the residual energy of the nodes.

3. In a network with different residual energy nodes; both the energy and data may be loss when low energy nodes become intermediate node and die-out before round-time. Therefore, the network need to be constructs in such a way that higher energy nodes should hold more traffic than the lower one.
4. In tree based scheme, it is possible to lighten up the load of the lower energy nodes by decreasing their node degree.
5. For real-time application, the sensed data should be transmitted to the sink under a certain delay. Hence, in delay constrain application tree based scheme with minimum height will be the best one.

5.3 Load Calculation of Sensor Nodes

Assume that $V = \{v_1, v_2, v_3, \dots \dots, v_n\}$ is the set of sensor nodes uniformly deployed over a field to continuously monitor the environment and they form a connected graph $G(V, E, D)$, where E is a set of bidirectional wireless link and D is the path-distance in between the nodes. If two nodes v_i and v_j are within the communication range with each other, then the link represent as $e(v_i, v_j) \in E$ and d_{ij} is the distance between the node v_i and node v_j where $d_{ij} \in D$. The network has the following characteristics:

- The network is centralized and static, i.e. the sink has the information about all the nodes of the network and both the nodes and sink are all stationary after deployment.
- Nodes are heterogeneous i.e. they have the different initial energy and the sink has infinite power supply.
- Root is the highest energy nodes and is responsible to communicate with the sink.
- All nodes have the finite communication ability and aggregate the collected data with their own sensing data which produce a single data packet.

In order to evaluate the energy dissipation between nodes v_i and v_j, we use the first order radio model as used by Ramanan et al [23], with the identical parameter values. The energy consumption to transmit an *l*-bit data packet is:

$$E_T(i,j) = l * \left(E_e + \varepsilon_{mp} * d^{\gamma}\right) \tag{5.1}$$

Here, E_e is the energy dissipated to operate the transmitter circuitry and $\varepsilon_{mp} * d^{\gamma}$ is the energy consumption for transmission of a single bit over a distance d and $\gamma \in \{2,4\}$ is the path-loss exponent.

The energy consumption of a node i to receive *l*-bit data packet is: $E_R(i) = l * E_e$ (5.2)

The sensor nodes also consume some of its energy during aggregation phase and the amounts is $E_{ax}(i)$ for the node i.

Consider a node v_i in a tree T produce an *l*-bit data packet in each round-time. Here, round-time is defined as the process of gathering all the data from sensor nodes to the based station called sink, regardless of how much time it takes [78] and the lifetime is the amount of round-time till first intermediate node die-out in the network. If v_i is an intermediate node, it needs to receive all data packets transmitted from its child, aggregates the collected data with its own data packet which generate a single *l*-bit data packet and transmit it to the parents. On the other hand, if v_i is the leaf node, it just transmits its own data to the parent node.

So, when v_i becomes an intermediate node in the tree T with child degree $C(T, v_i)$, the total energy consumption per round (load) is:

$$E(T, v_i) = E_T(i,j) + C(T, v_i) * E_R(i) + E_{ax}(i) \tag{5.3}$$

In tree based topology, the node degree $D(T, v_i)$ of the nodes depends on their child-degree, i.e. for node v_i, $D(T, v_i) = C(T, v_i) + 1$.

Using equation (5.1) and (5.2) in equation (5.3), we get the load of the nodes as:

$$E(T, v_i) = l * \left\{E_e * D(T, v_i) + \varepsilon_{mp} * d^{\gamma}\right\} + E_{ax}(i) \tag{5.4}$$

In equation (5.4), the value of E_e and ε_{mp} are constant. For the application where the sensor nodes periodically generate the data packet with constant size, the value of l is also constant. Therefore, the energy consumption per round i.e. the load of the nodes depends on their node-degree and transmission cost. To maximize the lifetime of the nodes, we have to construct a tree where intermediate nodes have minimum node-degree with lower transmission cost. However, this process increases the overall path-cost of the network and it is also difficult to control the height of the tree. To overcome the problems mentioned above, we have to construct a load balancing tree where degree of the nodes and their path-cost should be kept balance with their residual energy, which is the challenging research issue in the area of WSNs.

The lifetime $L_{nod}(T, v_i)$ of a node v_i is defined by the number of round-time τ it can survive in the network, i.e.

$$L_{nod}(T, v_i) = \left\lfloor \frac{E_r(v_i)}{E(T, v_i)} \right\rfloor \tag{5.5}$$

Here, $E_r(v_i)$ is the residual energy of the node and $L_{node}(T, v_i)$ is the number of round-times a node can survive for the load $E(T, v_i)$. We have to maximize the round-time of minimum energy nodes to increase the lifetime of the network. Hence, the lifetime of the tree T is:

$$L(T) = \max \min_{i=0,1,\dots,N} L_{node}(T, v_i) \tag{5.6}$$

To achieve the objective mentioned above we have to construct a tree in such a way that nodes have the lower residual energy, will consume less energy per round than the nodes with higher residual energy which is called load balancing tree in this thesis.

5.4 Nodes Categorization Scheme

In order to construct a balanced tree T, the load of the nodes need to be balance in such a way that all the nodes have to survive almost same round-time as the lifetime of the network by acting as an intermediate or leaf node in the tree. Lifetime of the network (round-time) is calculated by using the equation (5.5) and it is defined by the amount of round-time τ till a percentage of nodes die-out in the network. To survive the estimated lifetime of the network, calculate the energy requirement of the nodes using equation (5.4) for node degree one and $D(T, v_i)$. For a node with zero node degree, the energy requirements to survive a predefined round-time is called leaf-node energy requirement and represents as $E_L(T, v_i)$. For a node with degree $D(T, v_i)$, the energy requirements to survive the predefine round-time is called intermediate-node energy requirement and represent as $E_I(T, v_i)$. The value of $E_I(T, v_i)$ is always larger than the value of $E_L(T, v_i)$. Compare the energy requirement of the nodes with their residual energy and grouping them into three distinct categories as: *rich nodes* (V_1), *bottleneck nodes* (V_2) *and poor nodes*(V_3), where

- $\boldsymbol{V_1 = \{v_i: E_r(v_i) \geq E_I(T, v_i)\}}$. Here, $V_1$is the set of nodes whose residual energy $E_r(v_i)$ is larger than the intermediate-node energy requirement $E_I(T, v_i)$. They are called "**rich-node**" and have enough energy to survive specific round-time even if all their neighbors become their child. They are the intermediate node candidate for the new constructed tree.
- $\boldsymbol{V_2 = \{v_i: E_I(T, v_i) > E_r(v_i)} > E_L(\boldsymbol{T, v_i})\}$. Here, V_2 is the set of nodes whose residual energy $E_r(v_i)$ is bounded by their intermediate-node and leaf-node energy requirements. Those types of nodes are called "**bottleneck-node**". We have to balance the load of those nodes so that they can survive maximum round-time, which also increases the lifetime of the network. Bottleneck node are both intermediate and leaf node candidate for the new constructed tree.
- $\boldsymbol{V_3 = \{v_i: E_L(T, v_i) \geq E_r(v_i)\}}$. V_3 is the set of nodes whose residual energy $E_r(v_i)$ is less than their leaf node energy requirement $E_L(T, v_i)$ and defines them as "**poor-node**" in this thesis. Those types of nodes are not able to survive in the expected round-time even for the zero child degree. They are the leaf node candidates and conserve their energy by periodically going to the sleeping mode.

Here, the rich nodes have the higher status than bottleneck nodes and the poor nodes have the lowest status. We have to construct a tree in such a way that, the higher energy nodes handle more traffic load than the lower one during data transmission. Therefore, the poor-nodes need to be the leaf nodes to conserve their energy where rich-nodes route most of the traffic by being intermediate node. The load of bottleneck-nodes should manage in such a way that they can survive in predefine round-time either acting as an intermediate or leaf node in the new constructed tree. A new tree formed by balancing the load of the overburden nodes is called energy-aware load balancing tree in this research.

5.5 Energy-Aware Bottleneck-Nodes Avoidance DG (EBA-DG)

The goal of EBA-DG is to construct a load balancing data gathering tree which balance the load of the nodes based on their residual energy. In load balancing tree, the lower energy nodes become leaf nodes and conserve their energy by periodically going to the sleeping mode and higher energy nodes become intermediate nodes; route most of the traffic of the network. The number of child of an intermediate node depends on its residual energy and load of the nodes. The algorithm is executed for each round of data aggregation and generates a single l bit data packet. Figure 5.1 shows a network scenario consisting of the nodes listed by the ID from 1 to 15 and the value following each node v_i is the amounts of its residual energy, denoted by $E_r(v_i)$. The line edges indicates the possible links between pairs of nodes and the communication cost of those link are considered as constant in this scenario.

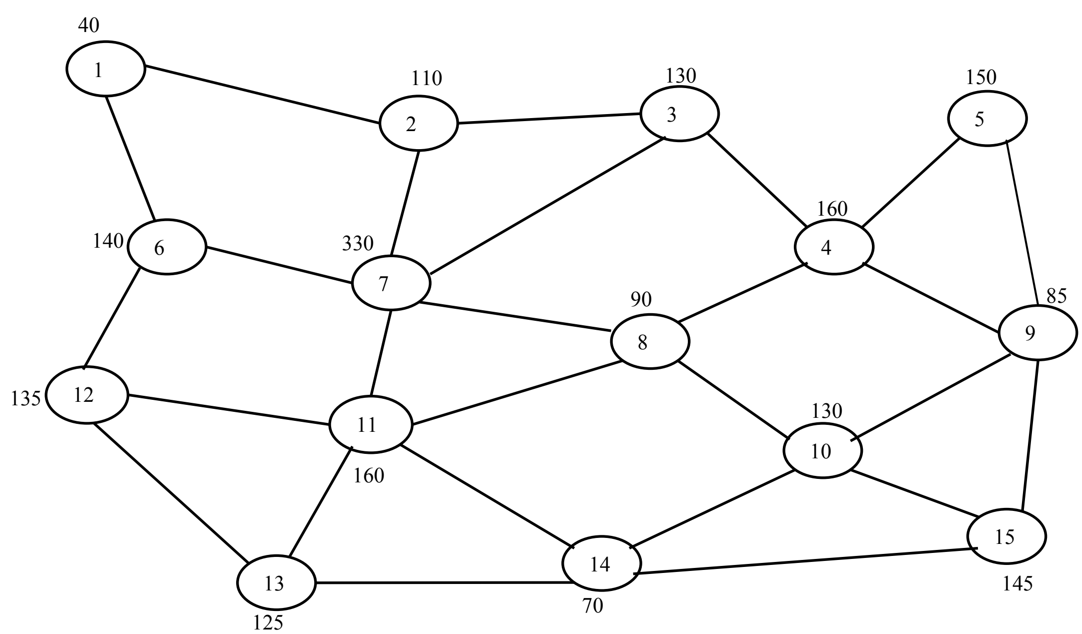


Figure 5.1: Network scenario with constant transmission-cost

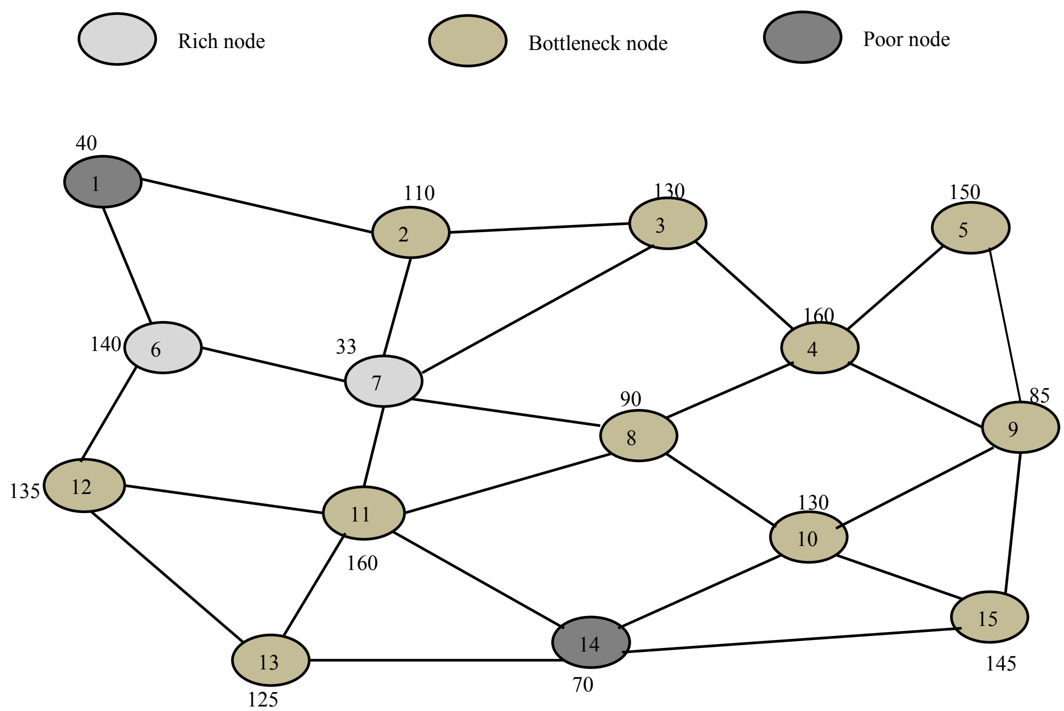


Figure 5.2: Node Categorization

EBA-DG is the centralized algorithm where sink has the information of all nodes in the network and is responsible to construct the tree. Initially, sink calculates the value of $E_L(T, v_i)$ and $E_I(T, v_i)$ of all nodes for a fixed round-time τ by considering their node degree as zero and $D(T, v_i)$. The comparisons between the residual energy $E_r(v_i)$ and the energy requirements divides the nodes into three distinct categories as: *rich nodes* (V_1), *bottleneck nodes* (V_3) and *poor nodes* (V_3). Here round-time τ is set by considering the lifetime of the nodes so that a specific percentage of rich nodes are available in the network. In Figure 5.2, all nodes are categorized as rich, intermediate and bottleneck node by comparing their residual energy with the energy requirement (leaf-node and intermediate nodes energy requirements) for the round-time $\tau = 400000$ unit. Here, rich node ID is 6, 7; poor node ID is 1, 14 and all the other nodes become bottleneck node. During tree construction higher energy rich nodes have the highest priority and the poor nodes have the lowest priority.

The proposed scheme selects the highest energy rich node as the root for the new constructed tree and the root is responsible to collects all the data from the network and route it to the base station. The root is the first covered node in the network connected with the tree T and stored in a list called covered list. The root sends the connection request to all its un-covered neighbors. Here, uncovered neighbors are those neighbors that are not covered, i.e., not connected to the new constructed tree T. When an uncovered neighbors is response to the connection request; it connects with the tree and also adds into the covered list as a covered node. The requested node becomes intermediate node and removes from the covered list.

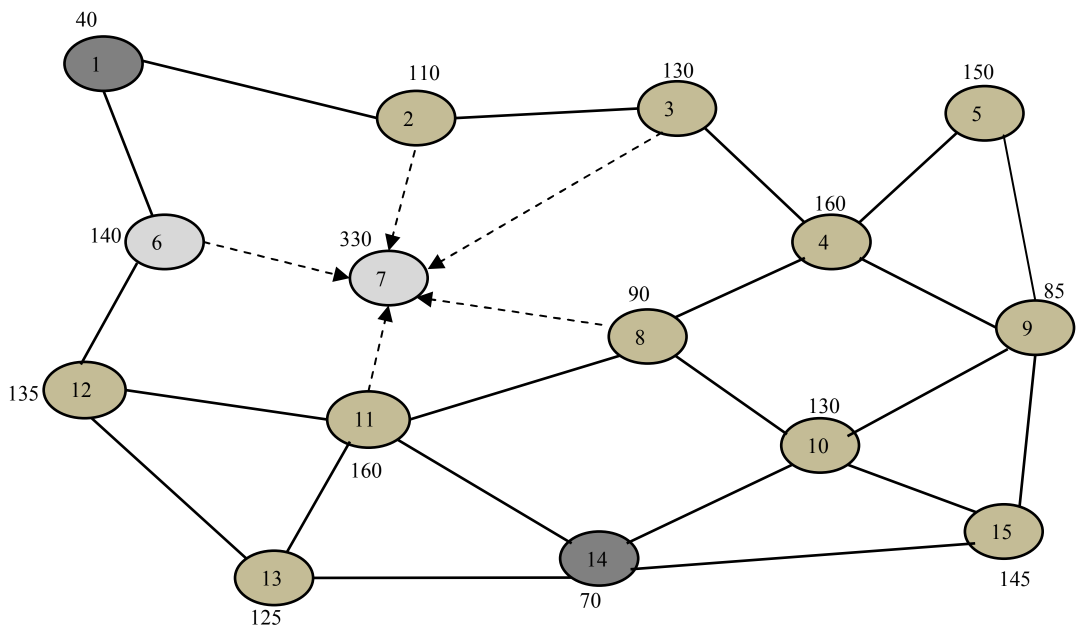


Figure 5.3: Tree construction (Iteration 1)

The scheme then recalculates the energy requirements of all nodes in the covered list by dropping the neighbors that are connected to the tree and change their status if possible. A node with highest residual energy and status is selects from the covered list as the next candidate node. The selected node sends the connection request to all its uncovered neighbors and become intermediate node by connecting the uncovered-neighbors. All the newly connected nodes become covered and add into the covered list and the requested node becomes intermediate node. Remove the intermediate node from covered list and recalculate the energy requirements of all the nodes by dropping the covered neighbors and change their status if possible. Select the next highest residual energy covered nodes as a candidate from the covered list and the processes continue until all uncovered nodes in the network become covered i.e., connected to the tree.

In Figure 5.3, node 7 is the highest energy node and selected as a root for the new constructed tree. Node 7 sends a connection request to all its uncovered neighbors, i.e. the neighbor nodes that are not connected to the tree. Nodes 2, 3, 6, 8 and 11 are response for this request and connected with node 7 represent by dotted edge and node 7 becomes intermediate node shows in Figure 5.3. Also the nodes 2, 3, 6, 8 and 11 become covered nodes and are added into the covered list.

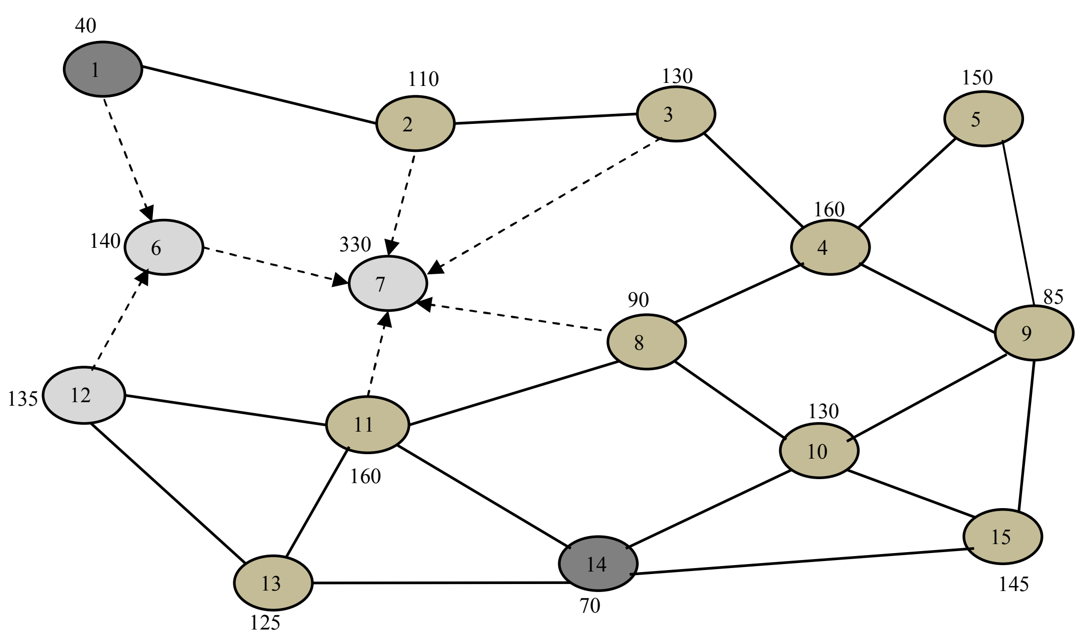


Figure 5.4: Tree construction (Iteration 2)

All nodes dropped the covered nodes 2, 3, 6,8,11 from the neighbor list, recalculate the energy requirements and change the status by comparing the residual energy with energy requirements shows in Figure 5.4 (a new node 12 becomes the rich node). Select the next highest energy rich node 6 from the covered list as the candidate which connects its uncovered neighbor nodes with ID 1 and 12, shows in Figure 5.4. Node 6 becomes the intermediate node and removes it from the covered list. The newly connected nodes are added into the covered list. All the nodes are then recalculate their energy requirements by dropping the covered nodes and the process continue until all the nodes become covered.

Algorithm EBA-DG (G, E_r)

Apply EBA-DG algorithm to the graph $G = (V, E)$, where V is the sensor nodes with edges E have the constant transmission-cost and $E_r(v_i)$ represents the residual energy of the nodes. The algorithm executes as:

Step1: Calculates the leaf-node energy requirements $E_L(T, v_i)$ and intermediate-node energy requirement $E_I(T, v_i)$ of all nodes by considering the node degree zero and $D(T, v_i)$. Categorize the nodes into three distinct set as, intermediate nodes (V_1), bottleneck nodes (V_2) and poor nodes (V_3) by comparing the energy requirements with their residual energy. Rich nodes get the highest priority than bottleneck nodes and poor nodes have the lowest priority.

Step2: Selects the highest energy rich node as the root for the tree. The root becomes intermediate node by connecting all its uncovered neighbors (nodes that are not connected to the tree). The newly covered nodes are added into a list called covered list.

Step3: Recalculate the energy requirement of all nodes in the network by dropping the neighbors that are connected to the tree and change status if require.

Step4: Selects the next node with highest residual energy and status from the covered list. The selected node becomes the intermediate node by connecting all its uncovered neighbors. All the newly connected nodes are added into the covered-list and continue the Step3 and Step4 until all the uncovered nodes in the network become covered.

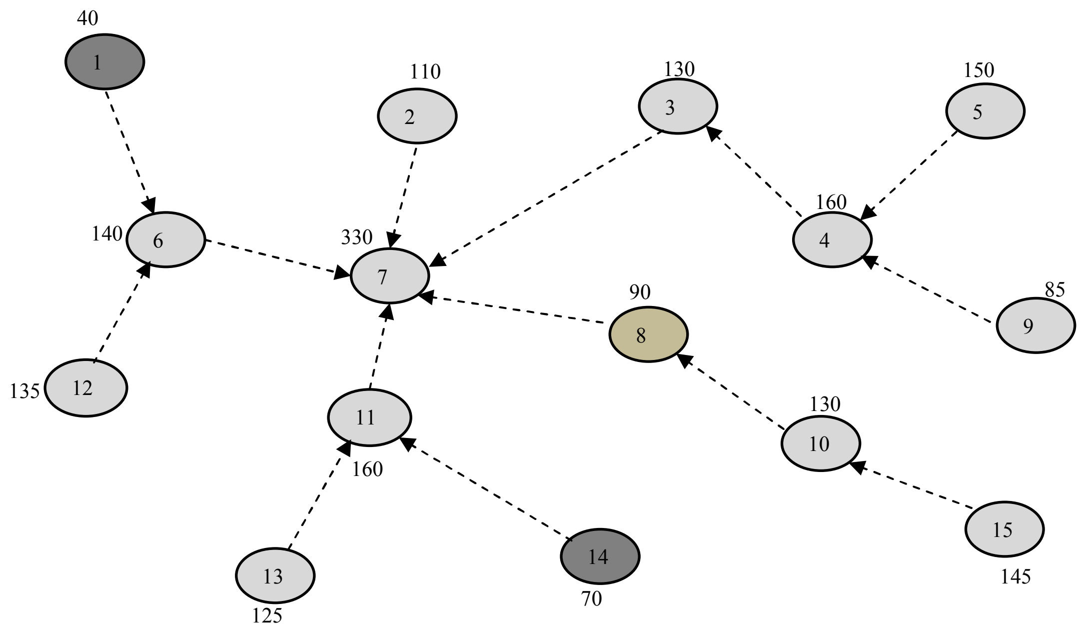


Figure 5.5: Final data gathering tree

Figure 5.5 shows a final data gathering tree and it ensures that the node with lower residual energy becomes the leaf node and the highest energy nodes become intermediate node in the new constructed tree. It connects the overburden (bottleneck) nodes with the tree as an intermediate node by decreasing their node degree which increases the lifetime of the network. The bottleneck nodes also have the opportunity to become leaf nodes to conserve their energy. The selected intermediate nodes ensure to survive a specific round-time which guarantees the network surveillance as predefined round.

5.6 Simulation Results and Comparative Analysis

This section discusses about the simulation results of the proposed scheme EBA-DG and then compares the results with the results of existing scheme EML-DG, based on the performance parameters. Different experimental results shows here by varying simulation parameters (number of nodes) of the network. As we have randomly deployed the sensor nodes in the network; for having the best result and gaining the accuracy, we have taken the same snapshot of experimental results for five times and finally take the average. We discuss and compare the experimental results and explained the causes of batter performance of the proposed scheme in the same simulation environment. The simulation results and comparative analysis of EBA-DG and EML-DG are discussed below.

5.6.1 Energy Loss per Round

In tree based scheme, the energy loss of the network is mostly depends on the lifetime of the intermediate nodes. Here, some intermediate nodes become overburden and die-out before the round-time of the network causes data loss of the network. The energy uses to generating and transmitting those data packets are also lost in this scheme.

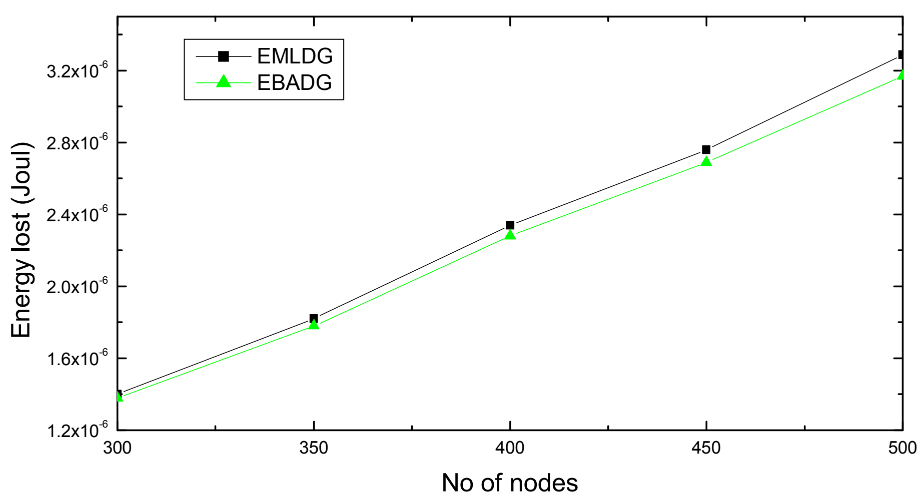


Figure 5.6: Energy loss per round in EML-DG and EBA-DG

For the number of nodes 300, Figure 5.6 shows that, the energy lost per round of EBA-DG is lower than the existing scheme EML-DG. This is because EBA-DG decreases the load of bottleneck nodes by decreasing their node degree, which increases the lifetime of those overburden intermediate nodes. As the number of node increases, the energy loss per round is also increases in both EBA-DG and in EML-DG but the increasing rate of EBA-DG is slower than the existing scheme. This is because, for dense network intermediate nodes become more overburden and die-out earlier in the network. For the nodes 500, the energy loss in EBA-DG is 3.65% lower than the EML-DG which was almost 1% lower for the nodes 300.

5.6.2 Average Lifetime of the Bottleneck Nodes

The bottleneck nodes are the overburden nodes control the lifetime of the network in tree based scheme. Die-out of the overburden (bottleneck) intermediate nodes break-up the link of the network and increases the packet and energy loss. The propose algorithm increase the average lifetime of the bottleneck nodes by decreasing their child-degree and some of the bottleneck nodes become leaf node; conserve their energy by periodically going to the sleeping mode.

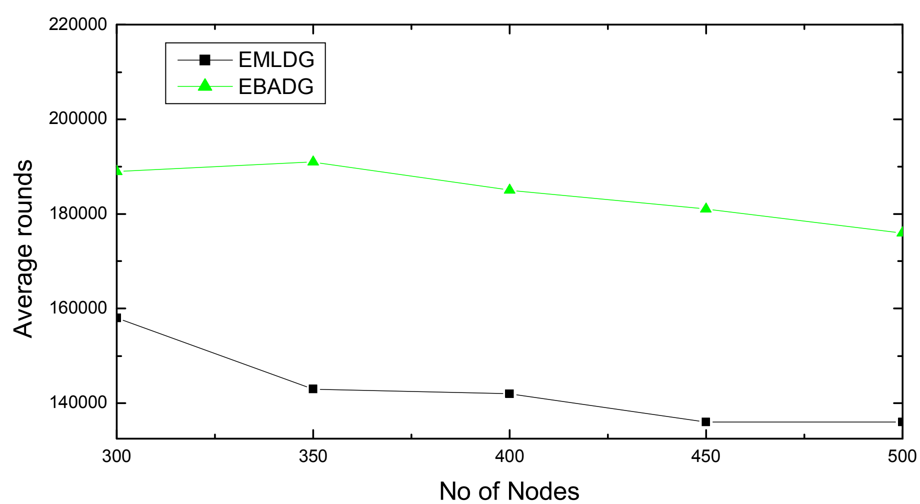


Figure 5.7: Average lifetime (round-time) of the bottleneck nodes in EML-DG and EBA-DG

Figure 5.7 shows that for the nodes 300, the average lifetime of the bottleneck nodes in EBA-DG is 16.39% higher than the existing EML-DG. As the number of nodes increases the bottleneck nodes become more loaded and their average lifetime is also decreases in both existing and proposed algorithm. However, the lifetime decreasing rate in EBA-DG is a little bit slower than in EML-DG. This is because; the EML-DG balances the load of the overburden nodes. For the nodes 500, the average lifetime of bottleneck nodes in proposed scheme become 22.79% higher than the existing scheme shows in previous mentioned figure. This simulation result gives the evidence of correctness and accuracy of the theoretical model of proposed scheme.

5.6.3 Lifetime of the Network

In tree based algorithm, die-out of the leaf nodes does not cause any data and energy loss in the network. Therefore, we have considered the network lifetime as the amount of round time till first intermediate node dies-out in the network. The proposed algorithm selects the rich nodes as the intermediate nodes and decreases the load of the overburden intermediate nodes, which dramatically increases the life time of network shows in the Figure 5.8.

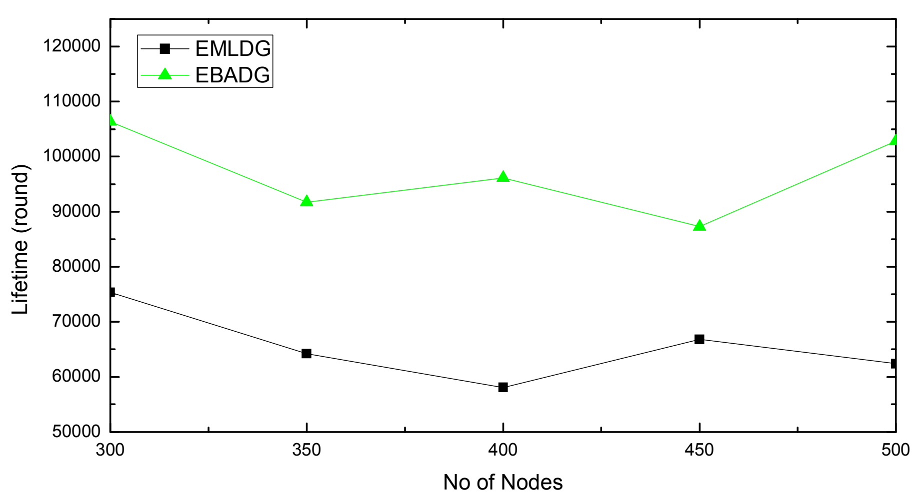


Figure 5.8: Lifetime of the network in EML-DG and EBA-DG

Initially for the nodes 300, the lifetime of the network in EBA-DG is 29.15% higher than the existing EML-DG scheme. This is because; the proposed algorithm increases the round-time of the overburden intermediate nodes. However, in existing algorithm some poor nodes become intermediate nodes and die-out within a few round-times. For dense network, the load of the intermediate nodes becomes higher which also decreases the lifetime of the network. For load balancing technique, the round-time decreasing rate in EBA-DG is a little bit slower than the existing EML-DG. For a network with 500 nodes, the lifetime of the network becomes 39.31% higher in EBA-DG than EML-DG scheme.

5.6.4 Height of the Tree

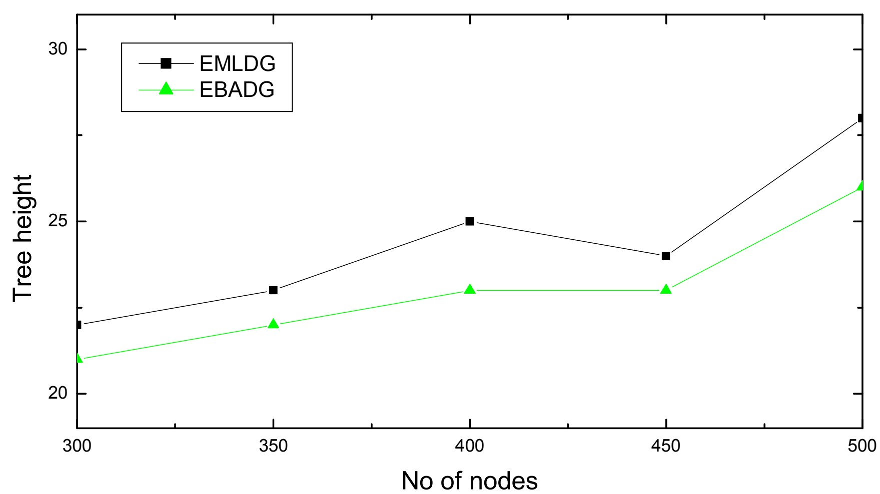


Figure 5.9: Height of the tree in EML-DG and EBA-DG

Figure 5.9 shows the comparison result of EBA-DG with EML-DG based on the tree height. The x-axis of the graph represents the number of nodes in the networks from 300 to 500 and y-axis represents the height of the tree. Here by definition, the height is defined as the maximum numbers of hop count from root to the leaf of the tree. For the nodes 300, the height of tree is 22 and 21 in both EMLDG and EBA-DG. As the number of nodes increases the height of the tree is also becomes higher in both schemes but their height difference is almost the same, i.e. the proposed algorithm constructs a tree with limited height same as the existing scheme.

5.6.5 Throughput of the Network

Throughput is defined as the amounts of data packets received by root in its round-time or the lifetime of the network. Since the root is hot-spot (node that route most of the traffic of the network) in tree based scheme, we have considered the highest energy rich nodes as the root node in this proposed scheme. Also the higher energy nodes increase the throughput of the network by becoming intermediate node in the network. The larger throughputs designate the lower packet and energy loss of the network.

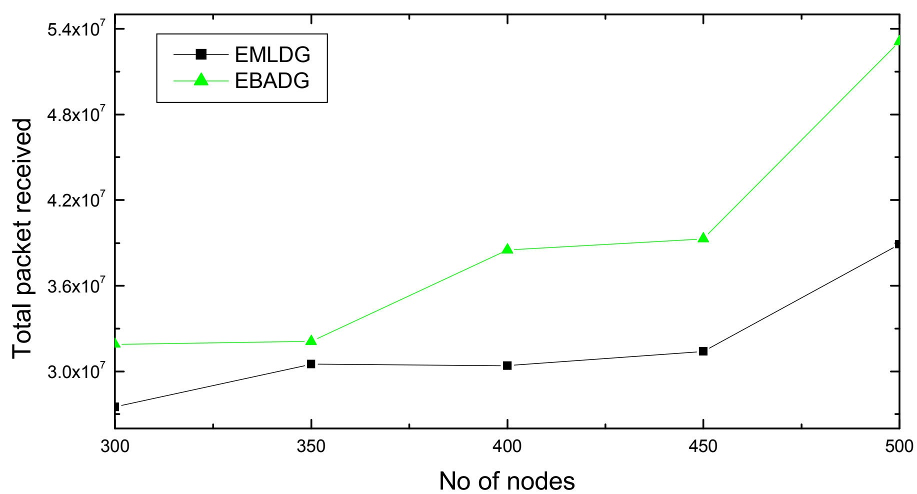


Figure 5.10: Throughput of EML-DG and EBA-DG

In Figure 5.10, x-axis represents the network's size (number of nodes from 300 to 500) and y-axis represents the total data packets received by the root in the round-time of the network. Initially for the nodes 300, EBA-DG receives 13.67% more data packets than the existing EML-

DG scheme in the round time of the network. This is because; EBA-DG selects higher energy nodes as the intermediate nodes and also balances the load of overburden intermediate nodes. As the number of nodes increases, throughput of the network becomes higher in both proposed and existing schemes. However, the increasing rate of EBA-DG is higher than EML-DG. For the node 500, the throughput difference becomes 26.82% in between proposed and existing schemes.

5.7 Summary and Comments

The proposed energy-aware bottleneck-nodes avoidance data gathering tree (EBA-DG) is designed and developed by addressing all the major drawback of the existing popular non path-cost based algorithms and hence provides strong methodology for the development of the emerging energy aware technology in WSNs. The scheme uses a mathematical model for load calculation of the nodes and then categorizes the nodes based on the load and residual energy. The experimental results show that the performance of the proposed algorithm is better than that of the existing others. Hence, the above mentioned technique will make the tree based algorithm robust and reliable in different WSNs applications.

Chapter 6

Proposed Energy Aware Bottleneck Nodes Avoidance Shortest-Path DG

The proposed non path-cost based scheme EBA-DG discussed in previous chapter enhance the round-time of the bottleneck nodes and construct a load balancing tree where lower energy nodes become leaf node and conserve their energy by periodically going to the sleeping mode. However, the communication cost (both transmission cost and path-cost) are also the crucial factor related with load of the nodes and the lifetime of the network. The load of the overburden nodes can trim down by reducing their transmission cost and nodes degree. Also, the lower cost path increase the overall lifetime of the network. To balance the load of the nodes and construct an energy-based shortest-path tree, this thesis considers the communication cost along with node degree and proposed a new tree based scheme named "Energy aware bottleneck nodes avoidance shortest-path data gathering tree (EBASP-DG)". EBASP-DG uses a new data structure named data gathering sequence (DGS) to increase the effectiveness of load calculation. To construct a load balancing energy based shortage-path tree, DGS explore the shortest-path of all nodes. The proposed algorithm ensures the expected round-time of the network and reduces the wastage of energy by efficiently selecting the intermediate nodes and balances the load of the overburden intermediate nodes. With the above mentioned modification, this scheme can easily be claimed to be the comprehensive tree base technique in WSNs.

6.1 Introduction

Construct an energy-aware path-cost based data gathering tree is an immense challenging issue in WSNs. A lot of researches [25, 88, 84, 79, 89] are going on to construct a load balancing energy based shortest -path data gathering tree. Chapter 3 discusses some the most popular path-cost based data gathering schemes and addresses some of their problems.

The centralize algorithm PEDAP [25] constructs a data-gathering tree by considering the path-cost during tree construction. After a certain number of rounds, the algorithm reconstructs the tree by collecting all the information of the network excluding the dead nodes. In this algorithm, lower energy nearest nodes from the base station become hot spot (nodes that need to route most of the traffic) and die-out earlier in the network. During setup phase, the sink needs to broadcast the information to all the nodes in the network which also causes extra communication cost of the network.

MLDGA [88] considers both the residual energy of the nodes and edge-cost to calculate the weight of the edge and the highest weighted edge get the most priority during tree construction. However, still it is not completely devoid the problems. During load calculation, the algorithm only considers the edge-cost and residual energy of the nodes, but node degree and path-cost are also the considerable factors affect the load of the nodes. Also, the lower edge cost decreases higher energy node probability to be intermediate node in the scheme.

ML-DG [84] started from the arbitrary tree and then balance the load of the bottleneck nodes that are shortly drain their residual energy due to the higher node degree or lower remaining energy. The scheme considers the edge-cost and residual energy of the nodes during tree construction. However, lower residual energy nodes decrease the edge probability to be connected to the tree. The height of the tree is also hard to control during load balancing of the overburden nodes.

DG-EL [79] constructs a cost effective shortest path data gathering tree by enhancing the link sharing between the source and set of destination. It considers the residual energy of the nodes along with path-cost during tree construction. In this scheme, some intermediate nodes become overburden and decreases the lifetime of the network. Also, the scheme does not provide any alternative path for link failure.

ERAPL [89] solves the mutual and loop transmission problems by constructing a data gathering sequence (DGS). Using DGS, the scheme calculates an outgoing traffic proposition (OTP) matrix by considering the communication cost. A generic algorithm (GAs) is then used to find the optimal OTP matrix, i.e. the optimal tree. The drawback of this algorithm is that the network becomes overhead for exchange the extra setup message between the sink and the nodes of the networks. Also the scheme does not give any guarantee to servility of the network in a predefine round-time. Practically, this type of algorithm is only suitable for those applications which generated constant rate data packets.

None of the above algorithms ensure the minimum path-cost load balancing tree survive a predefine round-times. Focusing the problems mention above this thesis proposes a scheme named "Energy aware bottleneck nodes avoidance shortage-path data gathering tree (EBASP-DG)." In this chapter, we discuss some contribution of this thesis that are mention previously as: i) a new path-cost based data structure named data gathering sequence (DGS), to effectively calculate the load of the sensor nodes, ii) construct an energy efficient load balancing energy-based shortest path data gathering tree.

The rest of this chapter proceeds as follows. Section 6.2 presents the motivations of the proposed path-cost based scheme. Section 6.3 proposed a path-cost based data gathering sequence (DGS). A proposed tree based scheme EBASP-DG discusses in Section 6.4. Sections 6.5 discuss about the simulation results of EBASP-DG and make a comparative analysis with the existing scheme. Finally, this chapter ends with by making a concluding discussion in Section 6.6.

6.2 Motivation of the Proposed Scheme

Now-a-days, to construct a load balancing data gathering tree with minimum path-cost is a research demand in WSN. The proposed scheme aims to achieve a significant extension in the network lifetime and decrease the path-cost of the network. The motivations of the proposed scheme are:

1. In a network with different residual energy nodes; both the energy and data will be loss when lower energy nodes become intermediate node and die-out before the specific round-time. Therefore, the network should be constructed in such a way that higher energy nodes hold more traffic than the lower one.

2. To construct a load balancing tree the load of the overburden intermediate nodes need to be balance by decreasing their child-degree and transmission cost.
3. The data gathering sequence (DGS) explore the shortest-path of all nodes through the root. So during tree construction every node knows their shortest-path and nearest neighbors predecessor nodes, which is effective to construct an energy-based minimum path-cost tree.
4. DGS use selective flooding which reduces the data flooding and decrease the overall load of the network.
5. To construct an energy-based shortest-path load balancing tree sensor nodes have to compromise in between round-time and path-cost of candidate parent nodes during parent selection.
6. For real-time application, the sensed data should be transmitted to the sink under a certain delay. Hence, the algorithm should be delay constrain with limited height.

6.3 Data Gathering Sequence (DGS)

A data gathering sequence $Seq(i_1, i_2, i_2, \dots \dots, i_n)$ is an order sequence of sensor nodes starting from a node i_1 and ending at node i_n as $i_1 \rightarrow i_2 \rightarrow \cdots \rightarrow i_n$, where i_n is the root of the network. In DGS, sensor nodes are only allowed to transmit the data to the neighboring node in forward direction specified in the DGS rather than in backsword direction. For instance, in the DGS $Seq(i_1, i_2, i_2, \dots \dots, i_n)$, node i_q can transmit to node i_{q+1} if there is a link between nodes i_{q+1} and i_q, but node i_q is not allowed to transmit to nodes i_{q-j} even if there exist a link between nodes i_{q-j} and i_q, $j = 1,2, \dots . n$. DGS is effective to calculate the load of the nodes and also used to solve the mutual and loop transmission problems of WSNs.

The DGS discuss in [89], considers the Euclid distances from nodes to sink during sequence construction. Since WSNs communication is the multi-hop communication, Euclid distance from sink to nodes is not effective to measure the path-cost of the nodes. In WSNs load of the nodes depends on its transmission-cost and child-degree where the transmission-cost depends on the Euclid distance of the parent nodes and the path-cost is the sum of the transmission cost of all nodes along the path from the node to sink. This thesis considers the Euclid distance of the neighbor's during DGS construction which ensure the shortest-path of all nodes and decrease the transmission cost. DGS also increase the effectiveness of load calculation and explore the shortest-path and nearest neighbor candidate parents of all nodes in the network.

To construct a path-cost based DGS, consider a weighted graph $G(V, E, D)$, where V is the set of nodes as $V = \{v_1, v_2, v_3, \dots \dots, v_n\}$, E is the set of bidirectional wireless link and D is the path-distance in between the nodes of the network. If two nodes v_i and v_j are within the communication range with each other, then the link represent as $e(v_i, v_j) \in E$, and d_{ij} is the distance between the nodes v_i and v_j where $d_{ij} \in D$.

Algorithm $\boldsymbol{DGS}(\boldsymbol{G}, \boldsymbol{D}, \boldsymbol{E_r})$

Apply DGS algorithm to a graph $G = (V, E)$, where V is set of the sensor nodes with edge E, have the distance value d where $d \in D$ and $E_r(v_i)$ represents the residual energy of the nodes. The algorithm execute as:

Step1: Select the highest energy node v_i as root for the sequence and calculate the shortest path (using Euclid distance) of all nodes from the root using Dijkstra algorithm.

Step 2: Store all the nodes in a list in ascending order sequence based on their path-cost. If two or more nodes have the same path-cost then the tie is break by considering the residual energy $E_r(v_i)$ of the nodes. The new constructed sequence is called DGS for the network.

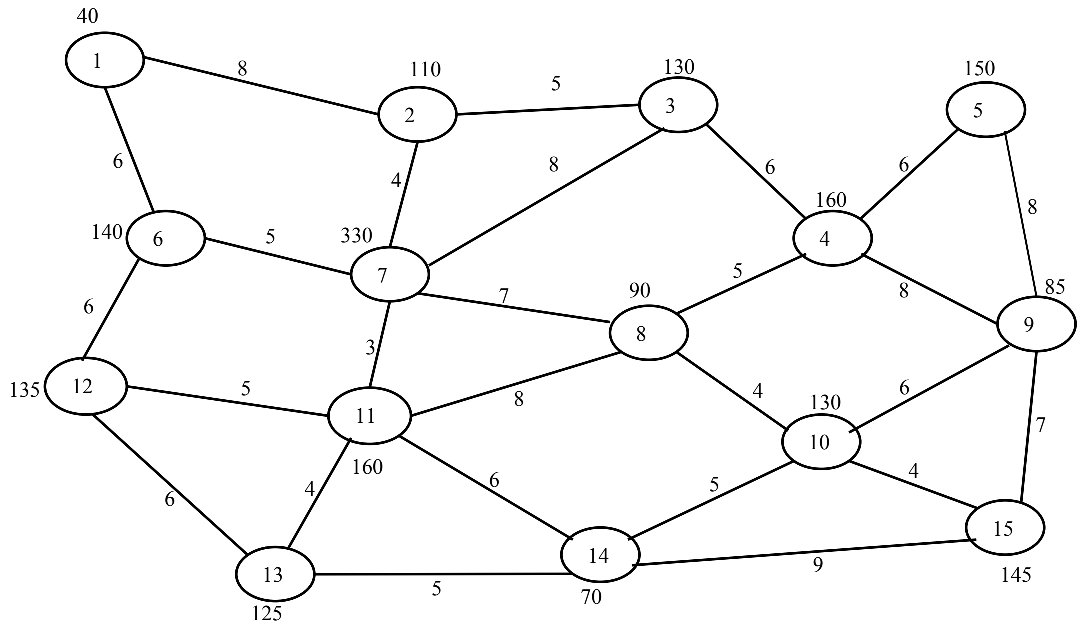


Figure 6.1: Network scenario with different path-cost

To construct a DGS, select the highest energy node as the root for the sequence and the root is responsible to collect all the data and send it to the sink. DGS scheme calculates the shortest path of all nodes from the root using Dijkstra algorithm and sorted them based on their path-cost. If two nodes have the same path-cost, the tie is break by considering the residual energy of the nodes. The new constructed sequence of sensor nodes is called the DGS.

In DGS the predecessor nodes become the candidate parent for their successor neighbors and the successor nodes become candidate child for their predecessor neighbors. A node is only allowed to send data to its candidate parent nodes and collect data from its candidate child nodes. All the nodes in the network act as a diffusion node except those that do not have any candidate child in the DGS.

Figure 6.1 shows a network scenario where sensor nodes are label by their node ID and assign them a numerical value call residual energy. The edges are labelled by the distance of the neighbors. In Figure 6.1, node 7 is the highest energy node and act as root for the DGS. The scheme calculates the shortest path of all nodes from the root node and stores them in a list in ascending order of path-cost shows in Figure 6.2. In Figure 6.3, all nodes connect to their predecessor neighbors as a candidate child node and connected with their descendent neighbors as candidate parent node. A node only receives data from it candidate-child nodes and route data to the candidate-parent nodes. The nodes that do not have any candidate-child nodes (node ID 1, 5 in Figure 6.3) are the farthest nodes from the root in the DGS and will be the leaf nodes in new constructed tree. In DGS, the node degree of a node is the number of descendent neighbors that have the link with this node plus one.

Node Id	7	11	2	6	8	13	3	12	14	1	10	4	15	9	5
Shortest- path	0	3	4	5	7	7	8	8	9	11	11	12	15	17	18

Figure 6.2: A DGS list

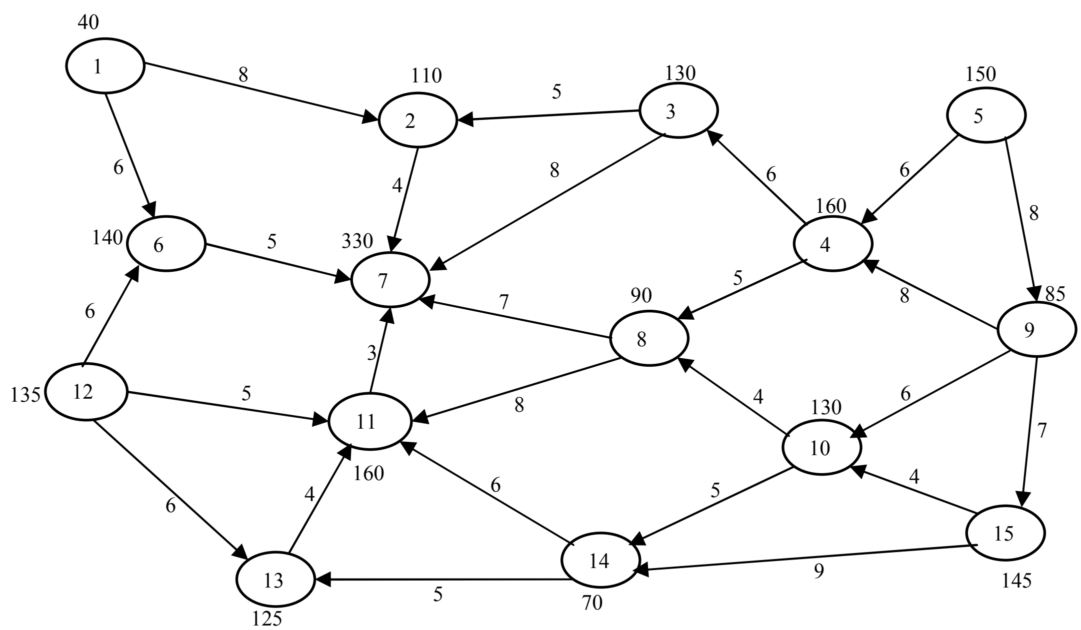


Figure 6.3: Network scenario with DGS

6.4 Energy-Aware Bottleneck-Nodes Avoidance Shortest-Path DG (EBASP-DG)

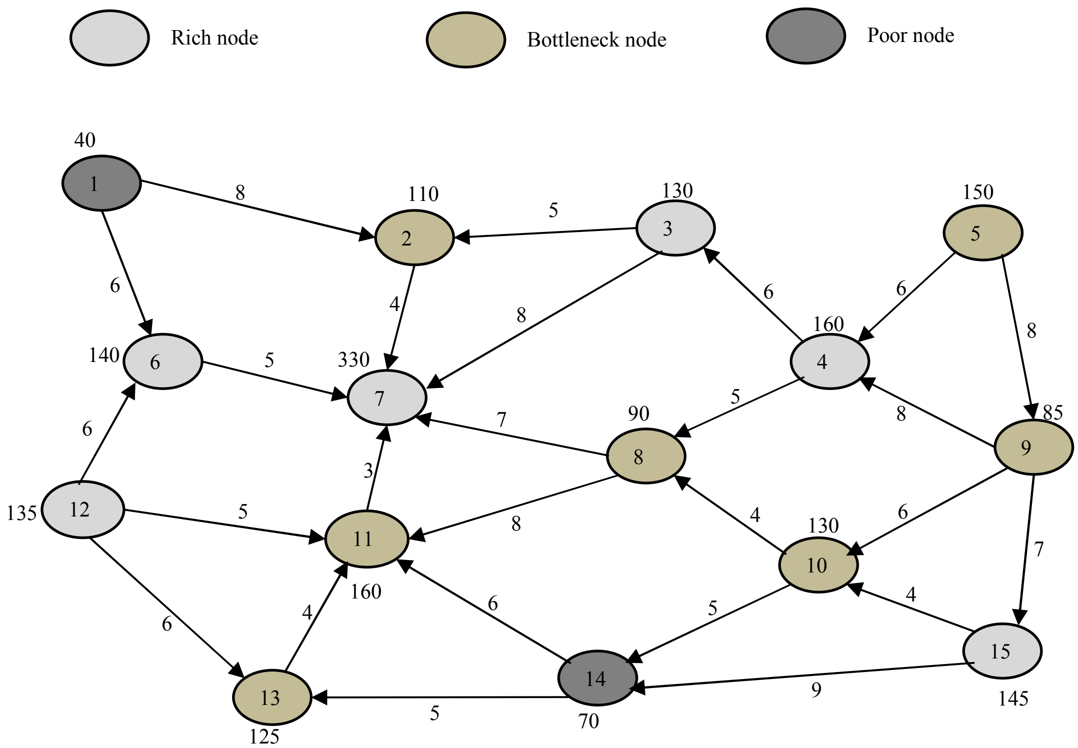


Figure 6.4: Node categorization without leaf node

To construct an energy-based shortest-path data gathering tree, the propose scheme uses the proposed mathematical model for round-time calculation and node categorization technique which was discussed in previous chapter. Initially, EBASP-DG constructs a DGS shows in Figure 6.3 for the network scenario of Figure 6.1, which increases the effectiveness of the load calculation and reliability of the network. Using the equation (5.4) with identical parameters values, the scheme calculates their energy requirement of all nodes in DGS by considering the transmission-cost (the average transmission-cost to send the data to its predecessor nodes) with node degree. For intermediate node energy requirements $E_I(T, v_i)$, the nodes degree $D(T, v_i)$ is considered as the number of descendent nodes plus one and for leaf node energy requirements $E_L(T, v_i)$, the node degree is considered as one. Comparing the residual energy $E_r(v_i)$ with energy requirements (intermediate node energy requirements $E_I(T, v_i)$ and leaf node energy requirements $E_L(T, v_i)$), divides the nodes into three distinct categories as: *rich node* (V_1), *bottleneck node* (V_2) *and poor node* (V_3) for the round-time $\tau = 400000$ shows in Figure 6.4. Here, round-time is set in such a way that a percentage of nodes in the network become rich node. Drop the poor nodes from the predecessor list of all nodes except those that do not have any alternate path rather than poor nodes. This is because; poor nodes do not have enough energy to survive the pre defined round-time as intermediate node and they will be the leaf node in the new constructed tree. The nodes having zero child-degree are also become leaf node in the new constructed tree shows in Figure 6.5. Calculate the round-time of all nodes by using equation (5.5) and the nodes store the ID of their shortest path and nearest neighbor descendent nodes (candidate parent nodes) to construct an energy-based shortest-path tree.

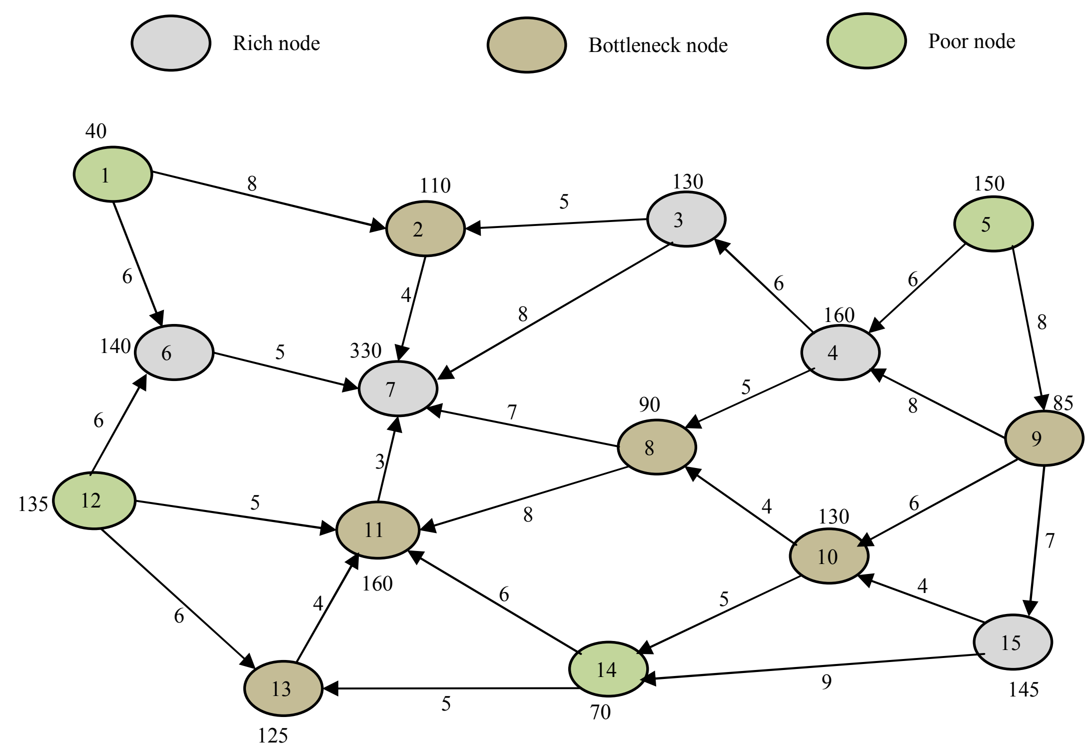


Figure 6.5: Node categorization with leaf node

At each iteration of EBASP-DG, select a lower path-cost node from the DGS following the sequence, except those nodes that do not have any descendent node and they will be participate in the network as a leaf node. Initially, for zero path-cost, the root node is selected as a candidate node and sends the connection request to all its descendent nodes. When a descendent node gets the first connection request from any of the predecessor node, descendent nodes directly connected with the requested node. When descendent nodes get the connection request from more than one predecessor node, the decision is taken by considering the round-time, status and path cost of the predecessor nodes. Higher energy rich nodes get the most priority than the bottleneck and poor nodes. The poor nodes have the lowest priority. When a node gets the connection request from two or more candidate parent nodes with same priority, the decision is taken by considering the path-cost and round-time of the requested nodes.

Algorithm $\boldsymbol{EBASPDG(G, D, E_r)}$

Apply EBASP-DG algorithm to a graph $G = (V, E)$, where V is set of the sensor nodes with edges E have the distance value d, where $d \in D$ and $E_r(v_i)$ represents the residual energy of the nodes. The algorithm execute as:

Step1: Select a highest residual energy node as the root and construct a DGS considering the shortest path of all nodes from the root.

Step2: Calculating the intermediate node energy requirement $\mathrm{E_I(T, v_i)}$ and leaf node energy $\mathrm{E_L(T, v_i)}$ requirements of all nodes where $\mathrm{E_I(T, v_i) > E_L(T, v_i)}$. Categorize the nodes by comparing the energy requirement with their residual energy $\mathrm{E_r(v_i)}$.

If $\mathrm{E_r(v_i) > E_I(T, v_i)}$ then nodes status is rich
else if $\mathrm{E_L(T, v_i) > E_r(v_i)}$ then nodes status is poor
else nodes status is bottleneck

Step3: Drop the poor nodes from the candidate-parent list of all nodes except those that do not have any alternate path rather than poor node. The nodes that do not have any descendent node become leaf node. Store the ID of shortest-path and nearest neighbor candidate parent of all nodes.

Step4: At each iteration, selects a node u (except the leaf node) from the DGS as a candidate by following the sequence of DGS. The candidate node sends the connection request to all its descendent nodes v_i. A node v ∈ V is connected with u if it satisfy the condition:

```
for all v ∈ descendent-neighbor [u]
   if (parent [v] = = null || (status [u]= = rich && status[parent[v]] ! = rich)) then
            the node v connect with node u
   else if (( u = = shortest-path parent[v]) &&
        (round-time[u] > ( round-time ( parent[v] ) * (1-α))) then
            the node v connect with node u
   else if ((u = = nearest-path parent[v] && (round-time [u] > round-time (parent[v])
        && ((shortest-path[u]+ Cost(u, v)) <= (shortest-path[v] * ( 1+ α)))) then
            the node v connect with node u
     else
        if ((round-time[u] > (round-time(parent[v]) * (1+α)) &&
           (shortest-path[u]+Cost(u, v) <= (shortest-path[v]*(1+ α )))) then
            the node v connect with node u
        end if
     end if
  end for
```

Step5: Continue the Step 4 until all the nodes in DGS send the connection request to their descendent neighbors.

If the requested node is shortest-path candidate parent, the decision is taken by comparing the round-time of the requested node with the round-time of the existing parent of the descendent node. For nearest-neighbors candidate parent, both its round-time and path-cost are compared with the round time and path-cost of the existing parent. For all the others request, path-cost and round-time of the requested node need to be good enough (with a factor α) than the path cost and round time of the descendent node's parent.

In this algorithm, Step1 construct a DGS by considering the path-cost of all nodes from the root. Categorize the nodes by comparing the residual energy with their energy requirements in Step 2. In Step 3, drop the poor nodes from the candidate parent list of all nodes except from those that do not have any alternate path rather than poor nodes. The nodes having zero descendent node become leaf node. All nodes store the ID of the shortest-path and nearest neighbor's predecessor for farther uses. At each iteration, a node (except leaf nodes) is selected as a candidate following the sequence of DGS which sends the connection request to all the descendent neighbors in Step 4. If the parent list of descendent node is null or the status of the requested node is rich, descendent node directly connected with the requested node. For the request from more than one node with same status the tie is break by considering the path-cost and residual energy. For the request of shortest-path candidate-parent, descendent nodes make the decision by comparing round-time with the round-time of its own parent and for nearest-neighbors candidate-parent; candidate nodes consider path-cost along with round-time during tree construction. The round-time and path-cost of the requested node need to be better than existing parent of descendent nodes for all the other cases. The process continues until all the nodes in DGS send their connection request. The value of α is set by analysis the simulation results to get the optimal results from EBASP-DG.

In Figure 6.6, node 7 is the lowest path-cost nodes and connects with the tree T as a root. In first iteration, node 7 sends the connection request to all its descendent neighbor nodes. For the first connection request, nodes 2, 3, 6, 8 and 11 directly connected with the node 7 and represent by dotted edge. In second iteration, node 11 sends a connection request to all its descendent neighbors 8, 12, 13, 14; and nodes 12, 13 and 14 satisfy the 1st condition and connected with the node 11. However, node 8 does not satisfy any of the condition and still connected with the node 7 shows in Figure 6.7.

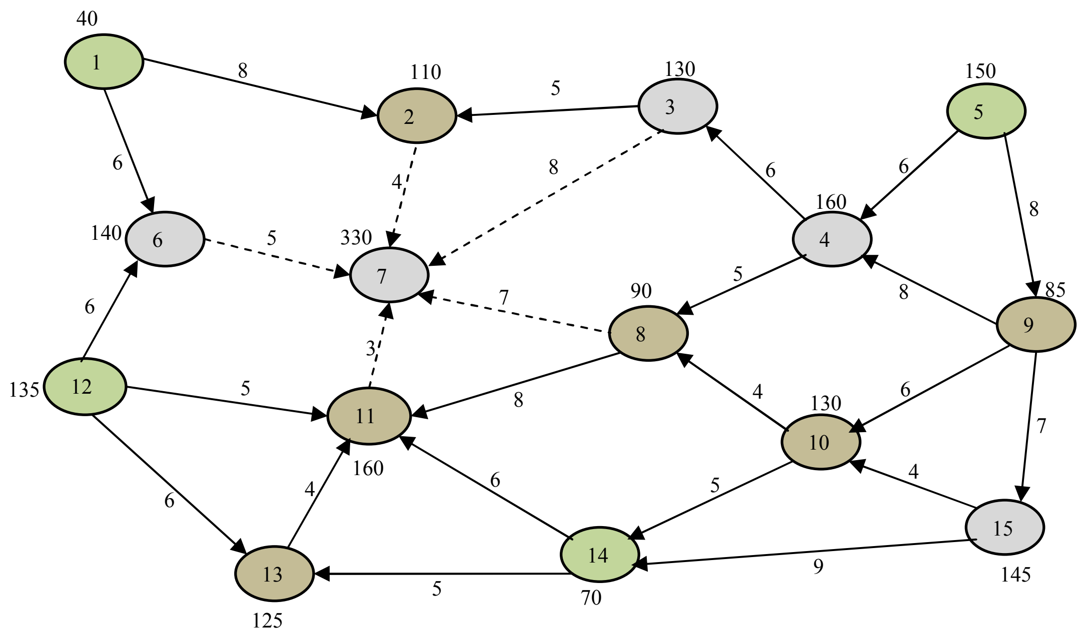


Figure 6.6: Tree construction (Iteration 1)

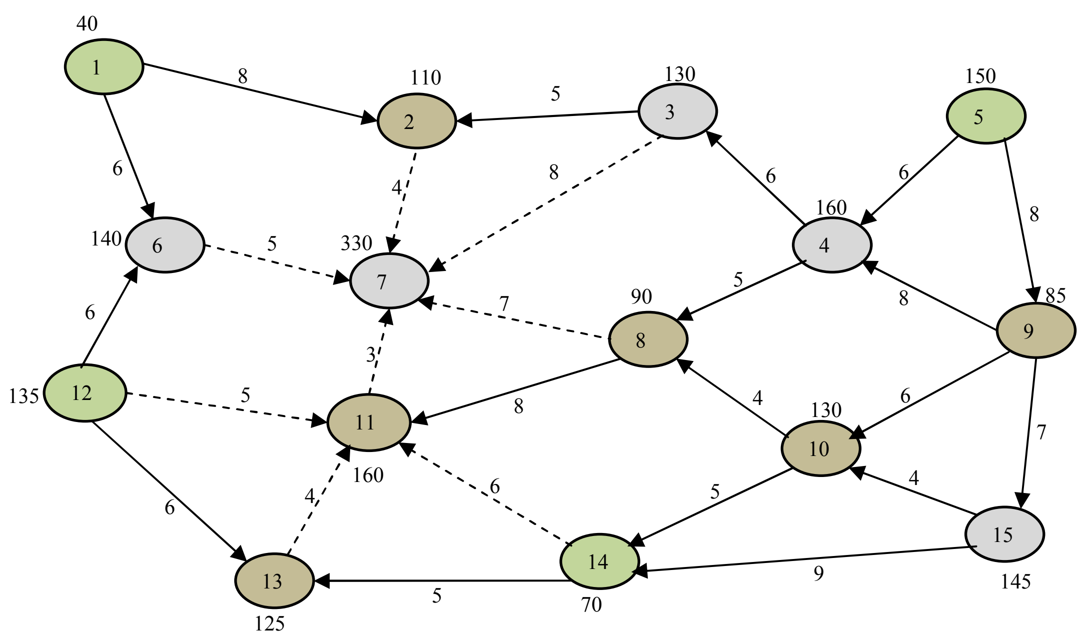


Figure 6.7: Tree construction (Iteration 2)

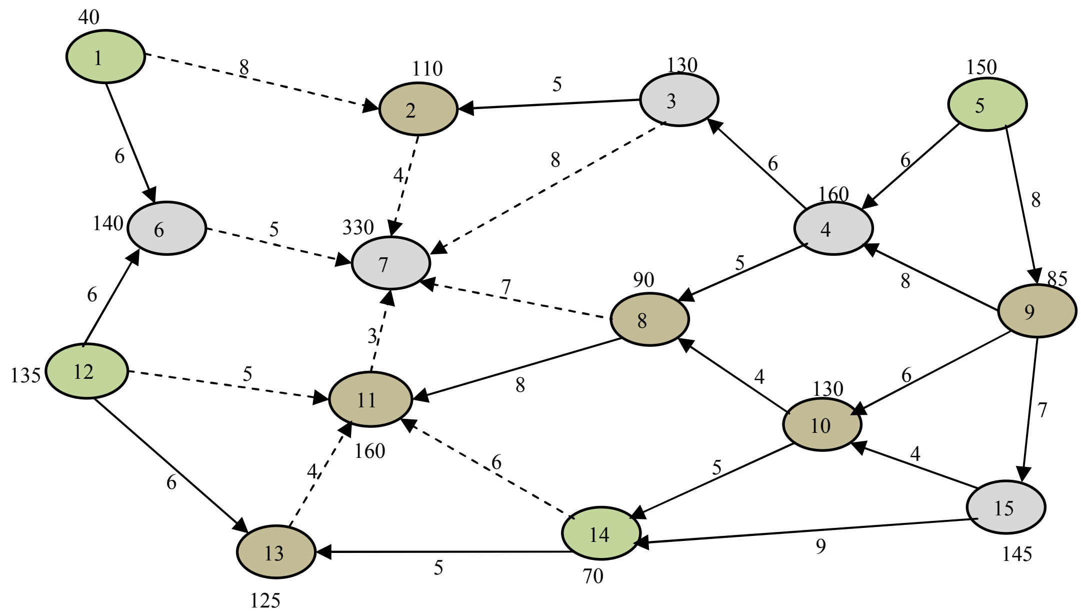


Figure 6.8: Tree construction (Iteration 3)

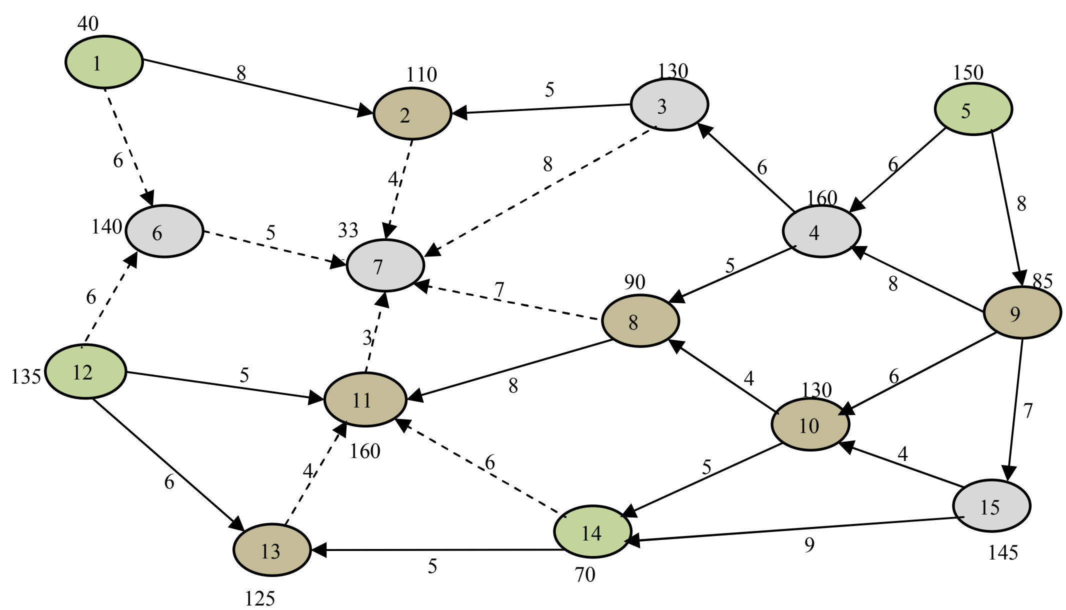


Figure 6.9: Tree construction (Iteration 4)

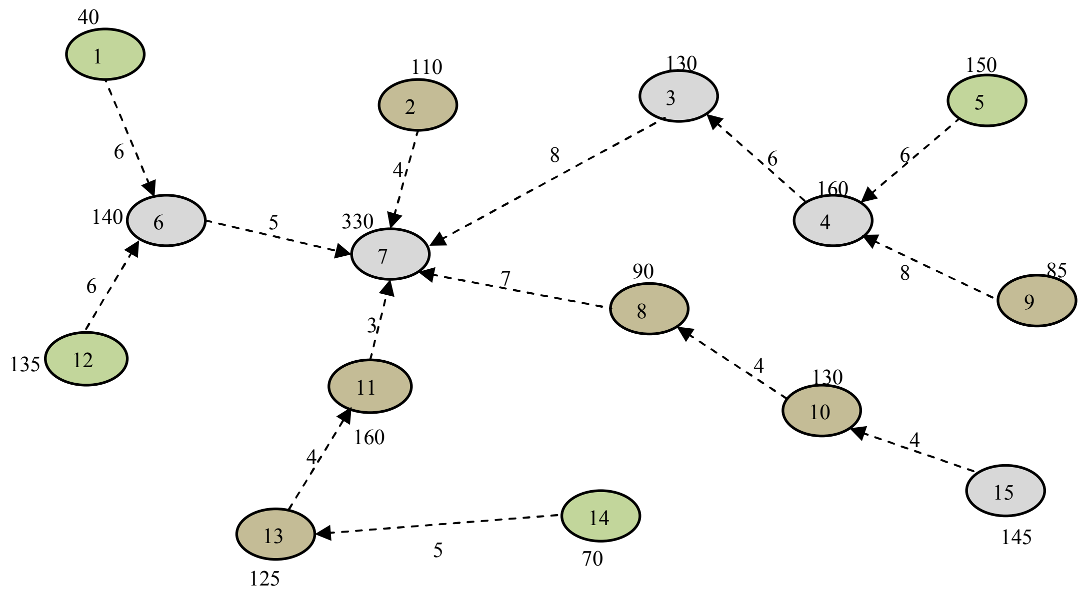


Figure 6.10: Final data gathering tree

In third iteration, node 1 connected with the node 2 but node 3 does not response the request of node 2 shows in Figure 6.8. In fourth iteration, node 6 sends the connection request to its descendent node 1 and 12 which already connected with node 2 and 11. Since, node 6 is the shortest-path candidate parent of node 1, node 1 take the decision by comparing the round-time of nodes 6 with its parent round-time and connected with node 6. For the node 12, the round-time of node 6 is higher than its parent and the path-cost along the node 6 is lower than the existing path-cost with a constant factor. Therefore, node 12 changes its parent and connected with node 6 shows in Figure 6.9. The processes continue until all the nodes in DGS (except the leaf nodes)sent the connection request to their descendent neighbors. Figure 6.10 shows the final tree for the scheme EBASP-DG.

6.5 Simulation Results and Comparative Analysis

This section shows the simulation results of path-cost and non path-cost based schemes (both proposed and existing) and compares them based on the performance parameters discussed in

Chapter 4. Different experimental results show here by varying parameters values. The next section of this chapter discusses the experimental results used to measure the most favorable value of alpha to get the optimal performance from EBASP-DG and the next few sections show the comparative analysis of simulation results and explain the causes of better performance of proposed algorithm.

6.5.1 The Optimum value of α (alpha) in EBASP-DG

The performance of path-cost based scheme is varying on different values of alpha. Hence, we have to pick the most favorable value of alpha to get the optimal result from EBASP-DG. Figure 6.11 shows that, the energy loss in EBASP-DG is gradually increases with increasing value of alpha. This is because the higher values of α selects the lower energy shortest-path candidate nodes as the intermediate nodes; die-out of those nodes before the round-time increase overall packet loss. Therefore, to decrease the energy loss, we have to select the minimum value of alpha. Here the simulator uses 300 numbers of nodes to simulate the experiments to get the most favorable value of alpha.

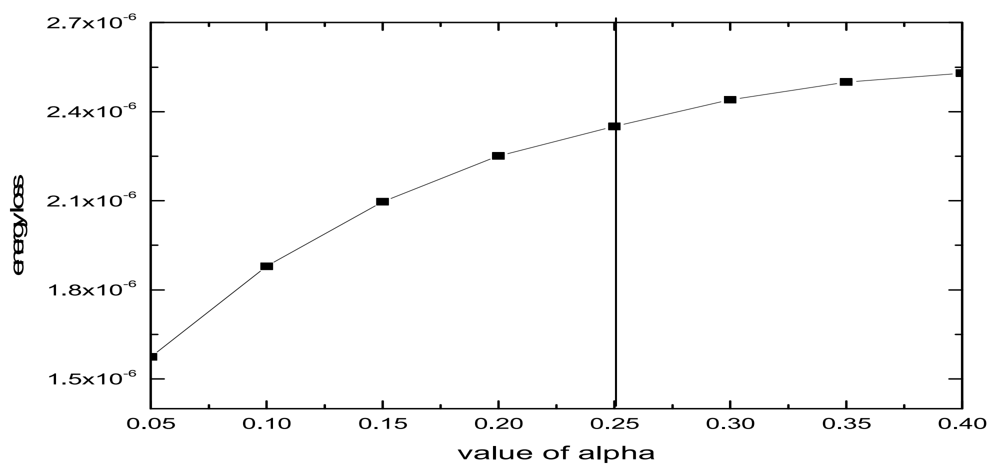


Figure 6.11: Energy loss per round for different value of (α) alpha

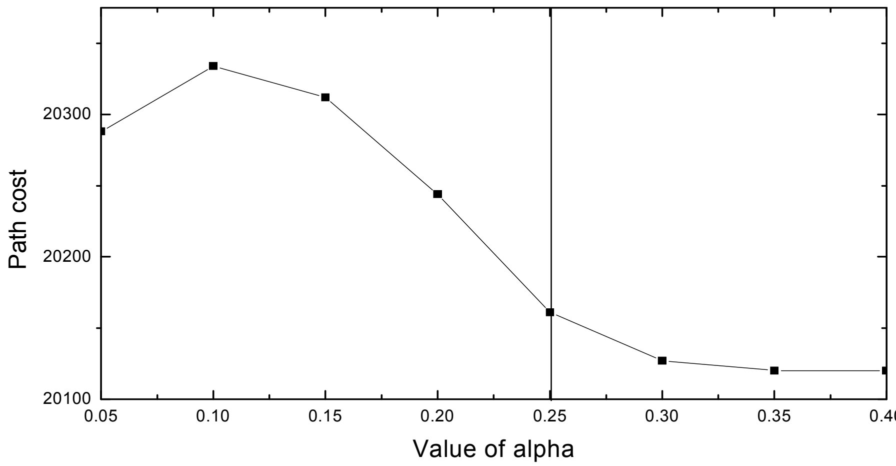


Figure 6.12: Path-cost per round for different value of (α) alpha

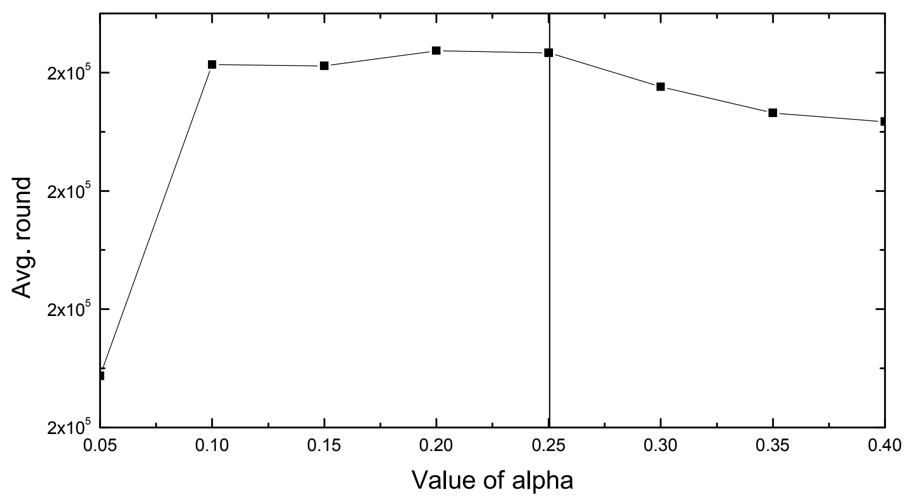


Figure 6.13: Average round of bottleneck nodes for different value of (α) alpha

Figure 6.12 shows that, path-cost of the tree in EBASP-DG is increases with the increasing values of alpha from 0.05 to 0.10. After then, for higher value of alpha, the shortest-path and nearest-neighbor candidate nodes becomes intermediate nodes and the path-cost are gradually decreases. However, the decreasing rate is almost constant at $\alpha = 0.25$ to 0.40. Hence, to get lower path-cost tree, we have to select the value of alpha in between 0.25 to 0.40.

Figure 6.13 shows that, in EBASP-DG the average round-time (round) of the bottleneck nodes are sharply increase with the increasing values of alpha. The round-time increasing rate become constant in between α = 0.10 to 0.25. Then the average round is gradually decreased with increase the value of α. Hence, to get the maximum round-time from bottleneck nodes, we have to selects the value of alpha in between 0.10 to 0.25.

From the simulation results in Figure 6.11, 6.12 and 6.13, it is clear that to get the better results in path-cost and in average round-time of bottleneck nodes; we have choice the larger value of alpha. However, to minimize the energy loss, we have to keep the value of alpha lower. From the discussion above, we pick the value of α = 0.25 to get the optimal results from the path-cost based scheme EBASP-DG.

6.5.2 Energy Loss per Round

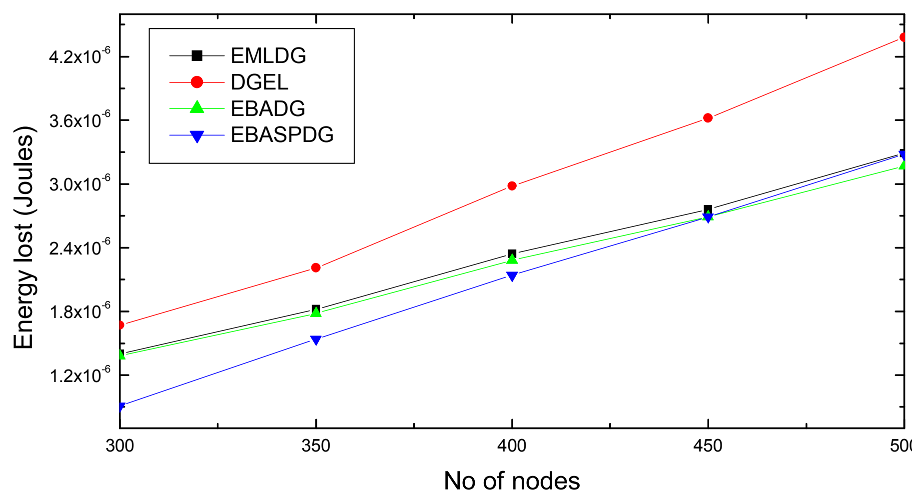


Figure 6.14: Energy loss per round in different schemes

In WSNs the loss of energy per round is depends on lifetime of the intermediate nodes. Overburden intermediate nodes have the lower lifetime in the network; causes energy and data loss of the descendent nodes. To decrease the energy loss per round, we have to construct a load balancing energy-based shortest-path tree.

In Figure 6.14, we take a common platform to compare EML-DG, DG-EL, EBA-DG and EBASP-DG schemes in energy loss point of view for the number of nodes from 300 to 500. In path-cost based scheme, the energy loss per round in EBASP-DG is 25.13% lower than the DG-EL for the nodes 300. This is because; the proposed scheme constructs an energy-based shortest-path tree with maximum round-time of intermediate nodes. As the number of nodes increases, the nodes in the network become more loaded and die-out earlier than life-time of the network which increase the energy loss rate in both schemes but the increasing rate in EBASP-DG is lower than the existing DG-EL.

The simulation result also shows that, the energy loss in EBASP-DG is lowest than the other three algorithms. This is because EBASP-DG decreases both path-cost and load of overburden intermediate nodes. As the number of nodes increase the energy loss are staidly increase in all the four schemes but the increasing rate in non path-cost based schemes is slower than the path-cost based schemes. In path-cost based schemes, dense network increases the load of the nodes and the proposed scheme select higher-cost path to balance the load of the nodes, which increases the overall energy loss rate. For the nodes 450 to 500, the energy loss of EML-DG, EBA-DG and EBASP-DG is almost same but the DG-EL performs worse than other three schemes.

6.5.3 Average Lifetime of the Network

In tree based schemes, die-out of the intermediate nodes break the communication path of the descendent nodes. Therefore, here the lifetime of the network is defined as the amount of time till the first intermediate node die-out in the network. Since the proposed schemes select the higher residual energy nodes as intermediate nodes and balance the load of overburden intermediate nodes, EBASP-DG dramatically increase the life time of the network shows in the Figure 6.15. Since the non path-cost based schemes only consider a constant transmission cost, the life time of the network in EBA-DG is higher than the path-cost based scheme EBASP-DG.

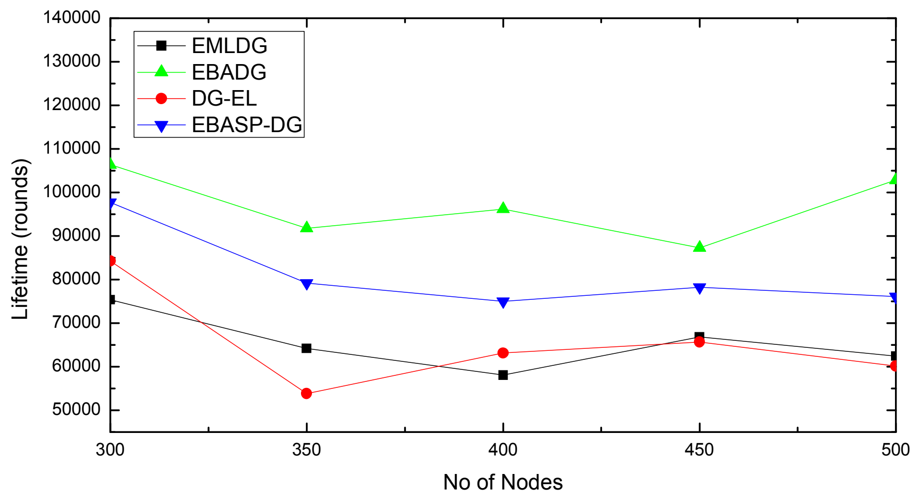


Figure 6.15: Average lifetime of the network in different schemes

As the number of nodes increases, intermediate nodes become more loaded which also decrease the round-time of the intermediate nodes of both categories of schemes. However, the lifetime of the network in proposed schemes is always better than the existing others for both path-cost and non path-cost based scenario.

6.5.4 Average Lifetime of the Bottleneck Nodes

The bottleneck nodes are the overburden nodes channel the lifetime of the network in tree based schemes. Science the proposed path-cost based scheme decrease the load of the bottleneck nodes by decreasing their node degree and transmission cost, the average lifetime of bottleneck nodes is 10.25% higher than the existing path-cost based scheme shows in Figure 6.16. As the network becomes dense, the overburden nodes also become more loaded which decrease the round-time of the bottleneck nodes in both path-cost and non path-cost based algorithms show in previously mentioned figure. However, the lifetime of the proposed schemes is always better than the existing others. This is because; both EBA-DG and EBASP-DG schemes balance the load of the overburden nodes by decreasing their node degree. Some of the bottlenecks nodes are also become leaf and conserve their energy by periodically going to sleeping modes in proposed schemes. Initially, the average round-time of bottleneck nodes in EBA-DG is a little bit higher than in EBASP-DG. As the network becomes denser; the path-cost based algorithm performs batter for their lower transmission cost.

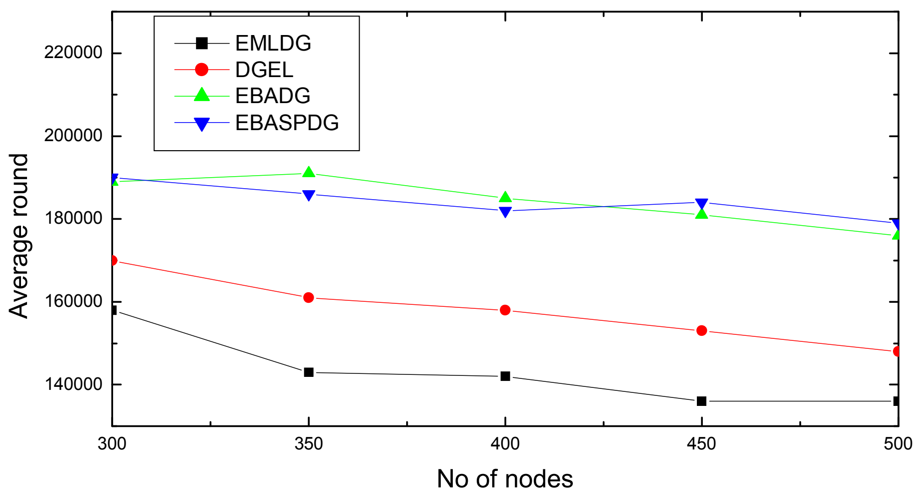


Figure 6.16: Average lifetime of the bottleneck nodes in different schemes

6.5.5 Average Lifetime of the Nodes

The average lifetime of the nodes is the average round-time of all nodes (rich, poor and bottleneck nodes) in the network. In path-cost based schemes, the average lifetime of the nodes in EBASP-DG is 1.36% higher than the existing DG-EL for the nodes 300. As the network becomes dense, nodes become more loaded which decreases the lifetime of the nodes in both path-cost and non path-cost based schemes. However, for load balancing of overburden intermediate nodes, the decreasing rate of propose scheme is lower than the existing schemes. For the nodes 500, the average lifetime of the nodes in DG-EL become 10.56% lower than in EBASP-DG shows in

Figure 6.17. For path-cost consideration, the average lifetimes of the nodes in path-cost based schemes are lower than the non-path cost based schemes. As the number of nodes increases the experimental results shows the same simulation output.

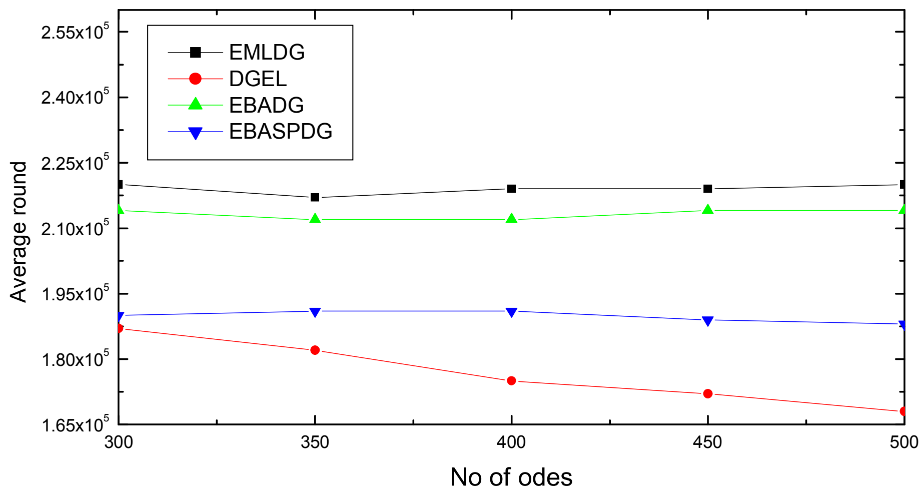


Figure 6.17: Average lifetime of the sensor nodes in different schemes

6. 5.6 Height of the Tree

Figure 6.18 shows that the height of the tree in proposed schemes is smaller than that produce by the existing others for the both path-cost and non path-cost based schemes. At the nodes 300, the height of the tree in DG-EL and in EBASP-DG are 15, 14 and it is decreases with the increasing number of nodes till 400. This is because, for increasing a lower percentage sensor nodes the propose scheme only increases the load of its intermediate nodes. After that limit, the height of the tree is increase as the number of nodes increase but the height of the tree of the proposed schemes is always lower than the existing others.

Figure 6.18 also shows that the height of proposed path-cost based schemes is always lower than the non path-cost based schemes; as the path-cost based scheme effectively balance the load of the nodes using DGS. For the nodes 300, the height difference in path-cost based and non-path cost based algorithms is 6 and for the nodes 400, it becomes maximum. After then, the height of the tree is staidly increases with the increasing numbers of nodes in both categories of schemes but the height of the proposed schemes are always lower than the existing others.

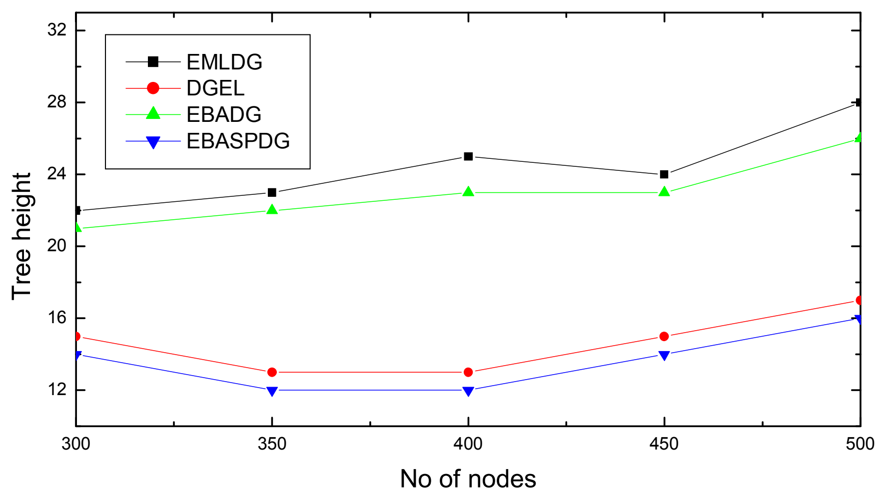


Figure 6.18: Height of the tree in different schemes

6.5.7 Throughput of the Network

In proposed path-cost based schemes, the root node receives more data packets than the existing others in the lifetime of the network shows in Figure 6.19. This is because EBASP-DG selects the highest energy node as the root node and also increases the round-time of the overburden intermediate nodes by decreasing their load. As the number of node increase, the network generates more data packets which also increase the throughput of the network. For nodes 300, the throughput of proposed algorithm is almost 13.67% to 21.76% higher than the existing others in both path-cost and non path-cost based schemes. For dense network, the throughput increasing rate in proposed schemes is higher than the existing others for their load balancing technique. For path-cost consideration, the intermediate nodes of EBASP-DG are more loaded than EBA-DG and the throughput of the proposed path-cost based scheme is a little bit lower than proposed non path-cost based scheme EBA-DG shows in the previous mentioned figure.

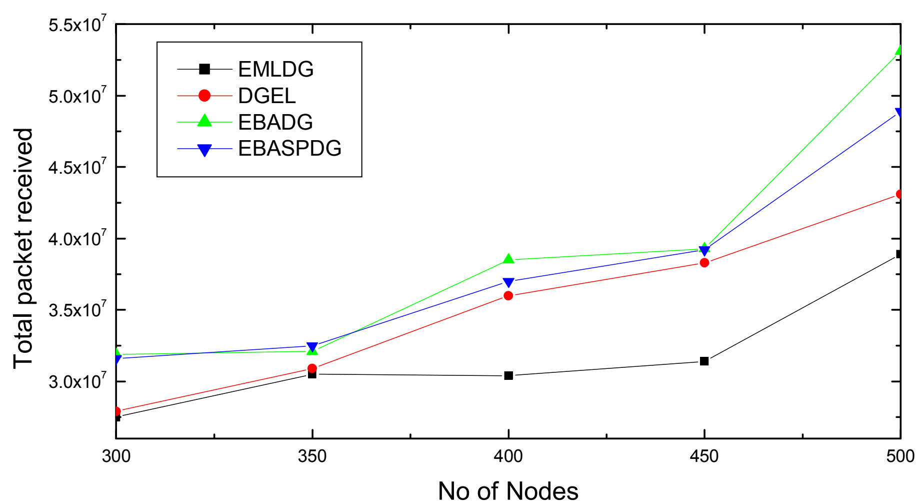


Figure 6.19: Throughput in different schemes

6.5.8 Total Path-cost of the Network

Path-cost is defined as the amount of cost form a node to root and the total path-cost is the sum of the path-cost of all nodes in the networks. Science the algorithm EML-DG and EBA-DG does not consider the pat-cost issue, we only compare the path-cost based schemes EBASP-DG with DG-EL in path-cost point of view. Initially, for the nodes 300, the total path-cost of EBASP-DG is lower than DG-EL shows in Figure 6.20. This is because EBASP-DG constructs a tree by connecting the shortest-path and nearest neighbor's predecessor nodes. As the number of nodes increase, the load of the intermediate nodes is also increases. To construct a load balancing tree in dense network, EBASP-DG relaxes the path-cost which increase the overall path-cost of the network in proposed schemes.

As the number of nodes increases the path-cost of the network is also staidly increases but the increasing rate of EBASP-DG is little bit higher than the DG-EL. For the nodes 380, the path-cost of the tree in EBASP-DG and DG-EL are the same. After than path-cost of EBASP-DG is a little bit higher than the existing scheme DG-EL, but their difference are in a limit.

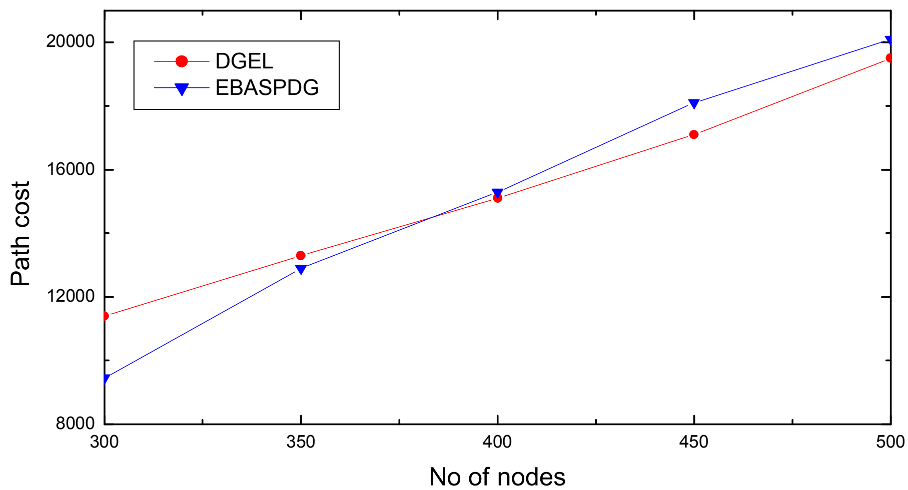


Figure 6.20: Total path-cost in different schemes

6.6 Summary and Comments

The experimental results shows that both the proposed EBA-DG and EBASP-DG schemes produce better results than the existing others in the suggested simulation environment. The proposed schemes decrease the energy loss of the network by increasing the lifetime of the intermediate nodes. Though EBA-DG degraded the average lifetime of the nodes, both the proposed algorithms increase the lifetime of the overburden intermediate nodes and ensure the expected round-time of the network. The proposed algorithms also limit the height of the tree and collect more data packets from environment than existing others in their lifetime. The experimental results also guide us to select and verify the optimum value of α (alpha) to get better performance from EBASP-DG. Those improvements mentioned above make the proposed schemes as the best preference to gather data in WSNs.

Chapter 7

Conclusions and Future Works

The application and adaptation of Wireless Sensor Networks (WSNs) in social and environmental purposes is increasing day by day from the past few years. WSN is a favorite choice to protect and monitor military, environmental, domestic infrastructure etc. for a specific period of time because of the low cost and easy implementation benefits. Energy efficiency operation and load balancing tree construction to gather data are the most challenging performance factors of WSNs. Efficient tree construction such as DB-MDST, CLMT, ETR, DCML, and PEDAP pay an important role to preserve energy by constructing a tree with lower path-cost. On the other hand, MLDGA, ML-DG and ERAPL perform the load balancing operation so that overburden nodes can shrink their energy loss which increases the lifetime of the network. However, the height of the tree is hard to control on those algorithms.

The contribution of this thesis starts with developing a mathematical model to calculate the load of the nodes which confirm that the load of the nodes basically depends on their neighbor-cost and node-degree. This model also shows that lifetime of the nodes depends on their residual energy and loads, and recommend to construct a balance tree by decreasing the load of the overburden intermediate nodes which increase the overall lifetime of the network.

This thesis also proposed a nodes categorization technique by comparing the node's residual energy with their energy requirement and divides the nodes into three distinct categories as: rich nodes, bottleneck nodes and poor node. The rich nodes have enough residual energy to survive specific round-time, even for all the neighbor becomes the child node and they become the intermediate nodes for the new constructed tree. On the other hand poor nodes cannot survive specific round-time even for zero child-degree and become the leaf node for the new constructed tree. The bottleneck nodes are the overburden nodes balance their load by decreasing their node degree and become intermediate or leaf nodes in the new constructed tree. An energy-aware bottleneck node avoidance data gathering tree (EBA-DG) is then proposed which constructs a load balancing data gathering tree by avoiding the overburden nodes as intermediate nodes during tree construction.

To construct the energy-aware bottleneck-node avoidance shortest-path data gathering tree (EBASP-DG) this thesis proposed a new data structure named data gathering sequence (DGS) which guide the direction of data flow of the nods through the root. DGS is effective to calculate the load of the nodes and the nodes get an idea about their transmission-cost and path-cost before the tree construction. Based on this DGS, EBASP-DG calculates the load of the nodes by using the mathematical model that mentions previously and then categorizes the nodes as: rich nodes, bottleneck nodes and poor nodes. Finally all the nodes in the network select their parent by considering their parent status, round-time, transmission-cost, path-cost and construct an energy-efficient load balancing data gathering tree.

The experiment results shows that the performance of the proposed algorithms is better than the existing algorithm in a network scenario of 300 ~500 nodes. Both the schemes lose less energy per round than the existing others. The algorithms also decrease the round-time variation of all nodes and increase the average lifetime of bottleneck nodes. A new data gathering tree constructed by the proposed algorithms has the limited height and lower path-cost. As the algorithms increase the lifetime of overburden intermediate nodes by decreasing their load and select the highest energy nodes as the intermediate node, the root collects more data packet during the lifetime of the network which increases the overall throughput of the network.

Though the proposed scheme significantly improves the low energy communication policy of WSN, yet some areas are still exist that require future study and experiments. Some of them are listed below:

i. **Testing the Scheme in a Test-bed Infrastructure:** In this thesis, all the proposed methods have been tested by simulating a real-world like sensor environment as the investigator did not have access to work with the real-world scenarios. Testing them in a test-bed real world infrastructure may further improve the schemes.

ii. **Prioritized Resource Allocation:** Resource reservation in sensor networks based on the nature of application can be considered as an unexplored area. Depending upon the applications, sensor networks may need to provide different priorities of services to several users in the upcoming future.

iii. **Security:** Due to the simple architecture of WSN and energy constraints, an intruder can easily launch attack to dormant the network. Hence, energy efficient defensive mechanism suits in WSN are the challenging tasks for the researchers.

iv. **Distributed Algorithm:** All the proposed and existing algorithms discussed here are the centralized algorithms, where sink is responsible to construct the data gathering tree. For the distributive nature of WSNs, it is the research challenge to construct an energy efficient distributive algorithm.

Deployment of WSN in different applications is an emerging and exciting area for the automation and improvement of next generation communications, monitoring and sensing activities. This emerging WSN technology, therefore, invests the technique for improving its performance and applicability managing its power resource efficiently. The proposed schemes are there a significant achievement towards these investigations and hence hasting the arrival of next generations WNS based applications.

Bibliography

[1] Agrawal P., "Energy efficient protocols for wireless systems," *IEEE International Symposium on Personal, Indoor, Mobile Radio Communications (PIMRC)*, vol. 2, pp. 564-569, 1998.

[2] Agre J. and Clare L., "An integrated architecture for cooperative sensing networks," *IEEE Computer*, vol. 33, no. 5, pp. 106-108, 2000.

[3] Akkaya K. and Younis M., "A survey on routing protocols for wireless sensor network," *Elsevier Ad-Hoc Network Journal*, vol. 3, pp. 235-349, 2005.

[4] Akyildiz F., Su W., Sankarasubramaniam Y. and Cayirci E., "A survey on sensor networks," *IEEE Communication Magazine*, vol. 40, no. 8, pp. 102-114, 2002.

[5] Azim A., Islam M. and Hossain B., "Cluster head selection following node admission in sensor networks," *International Conference on Electronics, Computer and Communication (ICECECE)*, pp. 844-847, 2008.

[6] Boukerche A. and Samarah S., "An efficient data extraction mechanism for mining associations rules from wireless sensor networks," *IEEE International Conference on Communications (ICC)*, pp. 3936-3941, 2007.

[7] Caraman D., Krus P. and Matt B., "Constrains and approaches for distribute sensor network security," *Technical Report*, vol. 0, no. 010, 2000.

[8] Chakrabarty K. and Iyengar S., "Scalable infrastructures for distribute sensor networks," *A Book by Springer publishers*, ISBN-10:1852339519, 2006.

[9] Chandrakasan A. and Brodersen R., "Low power digital CMOS design," *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 27, no. 4, pp. 473-484, 1992.

[10] Chandrakasan A., Amirtharajah R., Cho S., Goodman J., Konduri G., Kulik J., Rabiner W. and Wang A., "Design considerations for distributed microsensor systems," *Proceedings of the IEEE Custom Integrated Circuits*, pp. 279-286, 1999.

[11] Cormen T., Leiserson C., Rivest R. and Stein C., "Introduction to algorithms," *McGraw-Hill*, ISBN-10: 0262032937, 2001.

[12] Corner M., "Wireless-sensor networks find a fit in the unlicensed band," *Technical Writings in Electronics Design News*, 2006.

[13] Cost analysis of sensors, http://networks.cs.ucdavis.edu/`yick/research/SensorCost.html, last visited: 25th August 2011.

[14] eCos official website, http://ecos.sourceware.org/, last visited: 25th August 2011.

[15] Elson J., Bien S., Busek N. and Cerpa A., "EmStar: An environment for developing wireless embedded systems software," *Centre for Embedded Networked Sensing Technical Report (CENS)*, 2003.

[16] Epheremides A., "Energy concerns in wireless networks," *IEEE Wireless Communications*, vol. 9, no. 4, pp. 48-59, 2002.

[17] Goldsmith A. and Wicker. S., "Design challenges for energy-constrained ad hoc wireless networks," *IEEE Wireless Communications Magazine*, vol. 9, no. 4, pp. 8-27, 2002.

[18] Goldsmith A., "Wireless Communications", A *Book by Cambridge University Press*, ISBN-10: 0521837162, 2005.

[19] Hedetniemi S., Hedetniemi S. and Liestman A., "A survey of gossiping and broadcasting in communication networks," *IEEE Networks*, vol.18, no. 4, pp. 319-349, 1988.

[20] Heinzalman W., Sinha A. and Chadrakasan A., "Energy-scalable algorithm and protocols for wireless microsensor networks," *IEEE International Conference on Acoustics*, *Speech, Signal Processing (ICASSP)*, pp. 3722-3725, 2000.

[21] Heinzelman W., Chandrakasan A. and Balakrishnan H., "Energy-efficient routing protocols for wireless microsensor networks," *In Proc. of the 33rd Hawaii Int. Conf. on System Science (HICSS)* , vol. 2, pp. 1-10, 2000.

[22] Heinzelman W., Kulik J. and Balakrishnan H., "Adaptive protocols for information dissemination in wireless sensor networks," *ACM International Conference on Mobile Computing and Networking (MobiCom)*, pp. 174-185, 1999.

[23] Heinzelman W.B., Chandrakasan A. P. and Balakrishnan H., "An application-specific protocol architecture for wireless micro sensor networks," *IEEE Trans. on Wireless Communications,* vol. 1, no. 4, pp. 660-670, 2002.

[24] Heinzelman WR., Chandrakasan A. and Balakrishnan H., "Energy-efficient communication protocol for wireless micro-sensor networks," *In: Proc. of the Hawaii Int'l Conf. on System Sciences*, vol. 2, pp. 1-10, 2000.

[25] HO T. and Korpeoglu I., "Power efficient data gathering and aggregation in wireless sensor networks," *ACM SIGMOD Record*, vol. 32, no. 4, pp. 66-71, 2003.

[26] Holland J.H., "Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control and artificial intelligence," *MIT press, Cambridge, MA, USA,* ISBN-10: 0268581116, 1992.

[27] IEEE 1451, http://ieeexplore.ieee.org/xpl/standards.jsp, last visited: 25th August 2011.

[28] Hu T., Jhonson D. and Perrig A., "Secure efficient distance vector routing in mobile wireless ad hoc networks," *Proceeding 4th IEEE Workshop on Mobile Computing Systems and Applications*, pp. 3-13, 2002.

[29] Hu Yih-Chun, Perrig Adrin and Johnson Devid B., "ARIADNE: A secure on-demand routing protocol for ad hoc networks," *International Conference on Mobile Networking and Computing*, vol 11, no. 1-2, pp. 12-23, 2005.

[30] Hu. Y. and Jhonson D., "Implicit source routing in on-demand ad hoc network routing," *International Symposium on Mobile ad Hoc Networking and Computing*, pp. 1-10, 2001.

[31] Hwang S., Jin G., Shin C. and Kim B., "Energy-aware data gathering in wireless sensor networks," *IEEE Consumer Communications and Networking Conference (CCNC),* pp.1- 4, 2009.

[32] IEEE 802 family, http://www.ieee802.org/15/, last visited: 25th August 2011.

[33] Intanagonwiwat C., Govindan R. and Estrin D., "Directed Diffusion: A scalable and robust communication paradigm for sensor networks," *International Conference on Mobile Computing and Networking (MobiCom)*, pp. 56-67, 2000.

[34] Jelena M., Shairmina Shafi and Volislav B., "Admission control in 802.15.4 beacon enabled clusters," *International Conference on Wireless Communication and Mobile Computing (MobiCom)*, pp. 1269-1264, 2006.

[35] Jhonson D. and Maltz D., "Dynamic source routing in ad hoc wireless networks," *A chapter of Mobile Computing book by Kluwer Academic Publishers*, pp. 153-181, 1996.

[36] J-sim, Sensor network simulator, http://www.j-sim.org, last visited: 25th August 2011.

[37] Ramanan k. and Baburaj E., "Data gathering algorithm for wireless sensor networks: a survey," *International Journal of Ad-hoc, Sensor & Ubiquitous Computing (IJASUC),* vol. 1, no. 4, 2010.

[38] Kahn J., Katz R. and Pister K., "Emerging challenges: mobile networking for smart dust," *Journal of communication Networks*, vol. 2, no. 3, pp. 188-196, 2000.

[39] Kwon S., Kim J. and Kim C., "An efficient tree structure for delay sensitive data gathering in wireless sensor networks," *In: Pro of 22nd IEEE International Conference on Advance Information Networking and Applications (AINA)*, pp. 738-743, 2008.

[40] Ledeczi A., Maroti M. and BartokI., "Simple NEST application simulator," http:// www. cse.msu.edu/~arumugam/presentations/researchgrp/siesta.ppt, last visited: 25th August 2011.

[41] Levis P., Lee N., Welsh M. and Culler D., "TOSSIM: Accurate and scalable simulation of entire TinyOS application," *Proceeding of the 1st International Conference on Embedded Networked Sensor Systems*, pp. 126-137, 2003.

[42] Li L. and Halpern J., "Minimum-energy mobile wireless networks revisited," *IEEE International Conference on Communication (ICC)*, vol. 1, pp. 278-283, 2006.

[43] Liang J., Wang J. and Chen J., "A delay-constrain and maximum lifetime data gathering algorithm for wireless sensor networks," *Fifth International Conference on Mobile Ad-hoc and Sensor Networks*, pp. 148-155, 2009.

[44] Xiaoyuan Wang, Qing Zhang, Weiwei Sun, Wei Wang and Baile Shi; "A coverage-based maximum lifetime data gathering algorithm in sensor networks," *Seventh International Conference on Mobile Data Management (MDM),* pp. 33, 2006.

[45] Lindsey S. and Raghavendra CS., "PEGASIS: Power efficient gathering in sensor information systems," *In: Proc. of the IEEE Aerospace Conference*, vol 3, pp.1-6, 2006.

[46] Lindsey S., Raghavendra C. and Sivalingam K. M., "Data gathering algorithms in sensor networks using energy metrics," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 9, pp. 924 – 935, 2002.

[47] Liu Y., Ngan H. and Ni M., "Power-aware node deployment in wireless sensor networks," *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC)*, vol. 1, pp. 128-135, 2006.

[48] Luprano J., Sola J., Dasen S., Koller J.M. and Chetelat O., "Combination of body sensor networks and on-body signal processing algorithms," *International Workshop on Wearable and Implantable Body Sensor Networks*, pp. 75-79, 2006.

[49] Madden S., Franklin M., Hellerstrin J., and Hong W., "TinyDB: An inquisitional query processing system for sensor networks," *ACM transactions on Database Systems (TODS)*, vol. 30, no. 1, pp. 122-173, 2005.

[50] Mazumdar R., Mason L. and Douligeris C., "Fairness in network optimal flow control: Optimality of product forms," *IEEE transactions on communications*, vol. 39, no. 5, pp. 775-782, 1991.

[51] Meghanathan N., "An algorithm to determine energy-aware maximal life nodes data gathering tree for wireless sensor networks," *Journal of Theoretical and Applied Information Technology(JATIT)*, vol. 15, no. 2, pp. 96-107, 2010.

[52] Misic J., Shafi S. and Misic V., "Real-time admission control in 802.15.4 sensor clusters," *International Journal of Sensor Networks(IJSNET)*, vol. 1, no. 1, pp. 34-40, 2006.

[53] Nitesh S., Gene T. and Jeong Y., "Efficient node admission for short-lived mobile ad hoc networks," *IEEE International Conference on Network Protocols (ICNP)*, pp. 269-278, 2005.

[54] Ns-2 simulator, http://www.isi.edu/nsnam/ns/, last visited: 25th August 2011.

[55] Park S., Savvides A. and Srivastava M.B., "SensorSim: A simulation framework for 56sensor networks," *ACM International Workshop on Modelling, Analysis and Simulation of Wireless and Mobile Systems (ACM-MSWiM)*, pp. 104-111, 2000.

[56] Perkins C. and Royer E., "Ad hoc on-demand distance vector routing," *IEEE Workshop on Mobile Computing Systems and Applications(WMCSA)*, pp. 90-100, 1999.

[57] Perosinol A. and Staiano A., "A neuro-fuzzy approach for sensor network data cleaning," *International Conference on Knowledge-Based Intelligent Information and Engineering Systems*, pp. 140-147, 2007.

[58] Perrig A., Stankovic J. and Wagner D., "Security in wireless sensor networks," *Communications of the ACM-Wireless Sensor Networks (CACM)*, vol. 47, no. 6, pp. 53-57, 2004.

[59] Perrone L.F. and Nicol D.M., "A scalable simulator for TinyOS applications," *Proceeding of the Winter Simulations Conference (WSC)*, pp. 679-687, 2002.

[60] Products, http:/www.xbow.com/Products/Wireless_Sensor_Networks.html, last visited: 25th August 2011.

[61] Puthenpurayil S., Ruirui G., Bhattacharyya S., "Energy-aware data compression for wireless sensor networks," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, pp. 45-48, 2007.

[62] Rabaey J., Ammer J., Silva J. and Patel D., "PicoRadio: Ad-hoc wireless networking of ubiquitous low-energy sensor/monitor nodes," *IEEE computer society workshop on VLSI*, pp. 9-12, 2000.

[63] Rodoplu V. and Meng T. H., "Minimum energy mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1333-1344, 1999.

[64] Schurgers, C. and Srivastava M., "Energy efficient routing in wireless sensor networks," *IEEE Military Communications Conference*, vol.1, pp. 357-361, 2001.

[65] Sharaf M. Beaver J., Labrinidis R. and Chrysanthis P., "TiNA: A scheme for temporal coherency-aware in network aggregation," *ACM International Workshop on Data Engineering for Wireless and Mobile access*, pp. 69-76, 2003.

[66] Shih E., Cho S., Ickes N., Min R., Sinha A., Wang A. and Chandrakasan A., "Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks," *International Conference on Mobile Computing and Networking (MobiCom)*, pp. 272-287, 2001.

[67] SimJava: http://www.dcs.ed.ac.uk/home/hase/simjava, last visited: 24th August, 2011.

[68] Sohrabi K., Gao J., Ailawadhi V. and Pottie G. J., "Protocols for self-organizations of a wireless sensor network," *IEEE Personal Communications*, vol. 7, no. 5, pp. 16-27, 2000.

[69] Stark W., Wang H., Worthen A., Lafortune S. and Teneketzis D., "Low-energy wireless communication network design," *IEEE Wireless Communication Magazine*, vol. 9, no. 4, pp. 60-72, 2002.

[70] Stavros T. and Leandeos T., "Optimal deployment of large wireless sensor networks," *IEEE Transactions on Information Theory*, vol. 52, no. 7, pp. 2935-2953, 2006.

[71] Steven S., Borbash A., Ephremides A. and McGlynn M., "An asynchronous neighbor discovery algorithm for wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 7, pp. 998-1016, 2007.

[72] Sung-Min J., Young-Ju H. and Tai-Myoung C., "The concentric clustering scheme for efficient energy consumption in the PEGASIS," *9th IEEE International Conference on Advanced Communication Technology (ICACT),* vol. 1, pp. 260-265, 2007.

[73] Taherkordi A., Mohammadi R. and Eliassen F., "A communication-efficient distributed clustering algorithm for sensor networks," *IEEE International Conference on Advanced Information Networking and Applications (AINA)*, pp. 634-638, 2008.

[74] TinyOS Community Forum "An open-source OS for the networked sensor regime," http://www.tinyos.net/, last visited: 25th August 2011.

[75] UC Berkeley WEBS Projects, http://nescc.sourceforge.net/, last visited: 25th August 2011.

[76] uC/OS official website, http://micrium.com/page/home, last visited: 25th August 2011.

[77] Ulmer, "Wireless sensor probe networks- Sensor-Sim II," http.www.craigulmer.com/portfolio/unlocked/000919_sensorsimii/index.html, last visited: 25th August 2011.

[78] Virmani Deepali and Satbir J., "Centralize lifetime maximizing tree for wireless sensor networks," *International Journal of Computer and Electrical Engineering*," vol. 1, no. 5, pp. 529-534, 2009.

[79] Wang Z. and Liu Y., "Data gathering routing algorithm based on energy level in wireless sensor networks," *IEEE International Conference on Future Computer and Communications (ICFCC)*, vol. 2, pp. 160-164, 2010.

[80] WSN Design for Ultima-ted life, http://www.intechopen.com/source/pdfs/25371/InTech-Wsn_design_for_unlimited_lifetime.pdf, last visited: 20th December, 2011.

[81] Wolfgang Kreutzer, Jane Hopkins and Marcel van Mierlo, "SimJava- a framework for modelling queuing network in java," *Proceeding of the Winter Simulation Conference,* pp. 483-488, 1997.

[82] Woo A. and Culler D., "A transmission control scheme for media access in sensor networks," *International Conference on Mobile Computing and Networking (MobiCom)*, pp. 221-235, 2001.

[83] Wood A. and Stankovic J., "Denial of service in sensor networks," *IEEE Computer Communications*, vol. 35, no. 10, pp. 54-62, 2002.

[84] Wu Y., Fahmy S. and Shroff N.B., "On the construction of maximum lifetime data gathering tree in sensor networks: NP-completeness and approximation algorithm," *IEEE 27th Conference on Computer Communication (INFOCOM)*, pp. 356-360, 2008.

[85] Wu Y., Fahmy S. andShroff N.B., "A novel energy-efficient data gathering algorithm for wireless sensor networks," *8th Word Congress on Intelligent Control and Automation (WCICA)*, pp. 7016 – 7020, 2010.

[86] Younis O. and Fahmy S. 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.

[87] Yuen K., Li B. and Liang B., "Distributed minimum energy data gathering and aggregation in sensor networks," *IEEE International Conference on Communication (ICC)*, vol. 8, pp. 3536- 3542, 2006.

[88] Zhang Q., Xie Z., Sun W. and Shi B., "Tree structure based data gathering for maximum lifetime in wireless sensor networks," 7th*Asia-Pacific Web Conference,* pp. 513-522, 2005.

[89] Zhu Yi., Wu Wan., Pan J. and Tang Yi., "An energy-efficient data gathering algorithm to prolong lifetime of wireless sensor networks," *Computer Communications*, vol.33, pp. 639-667, 2010.

==================