1-1-2005

A reliable and resource aware framework for data dissemination in wireless sensor networks

Vasu Jolly
University of Nevada, Las Vegas

Repository Citation
Jolly, Vasu, "A reliable and resource aware framework for data dissemination in wireless sensor networks" (2005). *UNLV Retrospective Theses & Dissertations*. 2636.
http://dx.doi.org/10.25669/u2ow-zl7j

A RELIABLE AND RESOURCE AWARE FRAMEWORK FOR DATA DISSEMINATION IN WIRELESS SENSOR NETWORKS

by

Vasu Jolly

Bachelor of Engineering
Nagpur University
1999

Master of Science Degree in Electrical Engineering
University of Nevada Las Vegas
2002

A dissertation submitted in partial fulfillment
of the requirements for the

Doctor of Philosophy Degree in Electrical Engineering
Department of Electrical and Computer Engineering
Howard R. Hughes College of Engineering

Graduate College
University of Nevada Las Vegas
December 2005

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

UNLV
UNIVERSITY OF NEVADA LAS VEGAS

Dissertation Approval
The Graduate College
University of Nevada, Las Vegas

November 16, 20 05

The Dissertation prepared by

Vasu Jolly

Entitled

"A Reliable and Resource Aware Framework for Data Dissemination in Wireless Sensor Networks"

is approved in partial fulfillment of the requirements for the degree of

Ph.D. in Electrical Engineering

Shahram Latifi
Examination Committee Chair

Dean of the Graduate College

Examination Committee Member

Examination Committee Member

Graduate College Faculty Representative

ABSTRACT

A Reliable and Resource Aware Framework for Data Dissemination in Wireless Sensor Networks

by

Vasu Jolly
Shahram Latifi, PhD., P.E., Examination Committee Chair
Professor of Electrical & Computer Engineering
University of Nevada, Las Vegas

Distinctive from traditional wireless ad hoc networks, wireless sensor networks (WSN) comprise a large number of low-cost miniaturized nodes each acting autonomously and equipped with short-range wireless communication mechanism, limited memory, processing power, and a physical sensing capability. Since sensor networks are resource constrained in terms of power, bandwidth and computational capability, an optimal system design radically changes the performance of the sensor network. Here, a comprehensive information dissemination scheme for wireless sensor networks is performed. Two main research issues are considered: (1) a collaborative flow of information packet/s from the source to sink and (2) energy efficiency of the sensor nodes and the entire system. For the first issue, we designed and evaluated a reactive and on-demand routing paradigm for distributed sensing applications. We name this scheme as IDLF-Information Dissemination via Label Forwarding. IDLF incorporates point to point data transmission where the source initiates the routing scheme and disseminates the information toward the sink (destination) node. Prior to transmission of actual data packet/s, a data tunnel is formed followed by the source node issuing small label information to its neighbors locally. These labels are in turn disseminated in the network. By using small size labels, IDLF avoids generation of unnecessary network traffic and transmission of duplicate packets to nodes. To study the impact of node failures and to improve the reliability of the network, we developed another scheme which is an extension to IDLF. This new scheme, RM-IDLF - Reliable Multipath Information dissemination by Label Forwarding, employ an alternate disjoint path. This alternate path scheme (RM-IDLF) may have a higher path cost in terms of energy consumption, but is more reliable in terms of data packet delivery to sink than the single path scheme (IDLF). In the latter scheme, the protocol establishes multiple (alternate) disjoint path/s from source to destination with negligible control overhead to balance load due to heavy data traffic among intermediate nodes from source to the destination. Another point of interest in this framework is the study of trade-offs between the achieved routing reliability using multiple disjoint path routing and extra energy consumption due to the use of additional path/s. Also, the effect of the failed nodes on the network performance is evaluated within the sensor system.

Performance of the label dissemination scheme is evaluated and compared with the classic flooding and SPIN. For the second issue, we proposed discrete energy efficient schemes, which are incorporated in the system in conjunction with RM–IDLF. Setting up a battery threshold ensures that data packets will not be dropped after the sensor node's battery level falls below the threshold value. Minimum transmission around the sink prevents fast energy dissipation of the neighboring nodes to the sink. Finally, directional forwarding is applied to RM-IDLF. In directional forwarding, the sensor nodes narrow the range of broadcasting data packets by restricting communication only to the nodes lying in the direction toward sink/s. A C++ simulator is implemented to validate the design and to study the performance of the wireless sensor network.

TABLE OF CONTENTS

ABSTRACT iii

LIST OF FIGURES ix

LIST OF TABLES xii

ACKNOWLEDGEMENTS xiii

CHAPTER 1 INTRODUCTION 1
  1.1 Wireless Micro- Sensor Networks 2
  1.2 Sensor Node Architecture 4
    1.2.1 SMART DUST 5
  1.3 WSN Protocol Stack 9
  1.4 Wireless Sensor Networks and Traditional Ad Hoc Networks 11
  1.5 Motivation and Design Challenges in Wireless Sensor Network 12
  1.6 Wireless Sensor Network Applications 18
  1.7. Contribution and Scope of the Dissertation 22

CHAPTER 2 ROUTING MANAGEMENT IN WIRELESS SENSOR NETWORKS .. 24
  2.1 Routing Techniques in Wireless Sensor Networks 25
    2.1.1 Routing Models 25
    2.1.2 Protocol Assessment 28
    2.1.3 Routing methodology in Wireless sensor networks 29
  2.2 Flat Routing 30
    2.2.1 Flooding 31
    2.2.2 SPIN- Sensor Protocols for Information via Negotiation 33
    2.2.3 Directed Diffusion 37
    2.2.4 Rumor Routing 39
    2.2.5 COUGAR 40
    2.2.6 ACQUIRE 41
  2.3 Hierarchical Routing 42
    2.3.1 Low Energy Adaptive Clustering Hierarchy- LEACH 42
    2.3.2 Power-Efficient Gathering in Sensor Information Systems-PEGASIS 45
    2.3.3 Power concerned routing 46
    2.3.4 Threshold Sensitive Energy Efficient Sensor Network Protocol 53
    2.3.5 Self-Organizing Protocol 54
  2.3 Location Based Routing Protocols 57
    2.4.1 Geographic Adaptive Fidelity 57
    2.4.2 Minimum Energy Communication Network 60

2.4.3 Geographic and Energy Aware Routing (GEAR) 61
2.4 Multipath Routing Protocols 62
2.5 Negotiation Based Protocols 64
2.6 Coherent and Non Coherent Routing 65
2.7 QOS Based Routing 66
2.8 Open Issues in Sensor Network Routing 67
2.9 Conclusions 68

CHAPTER 3 RELIABILITY IN WIRELESS SENSOR NETWORKS 71
3.1 Fault Tolerance Techniques in Wireless Sensor Networks 72
3.2. Fault Tolerance and Multisensor Fusion 76
3.2.1 Modeling Single-Type Spares 79
3.2.2 Modeling Pooled Spares 81
3.2.3 Reliability versus Cost 84
3.2.4 Multifusion Sensor Networks 87
3.3. Conclusions 89

CHAPTER 4 ENERGY EFFICIENT INFORMATION DISSEMINATION IN WIRELESS SENSOR NETWORKS 90
4.1. Motivation of Current Research 90
4.2 Sensor Network Topology 95
4.3 Preliminaries 96
4.4 Information Dissemination by Label Forwarding 101
4.4.1 Assumptions 101
4.4.2. Description of IDLF Algorithm 103
4.5 IDLF- Directional Forwarding Model 111
4.6 Energy Management Proposal 115
4.6.1 Minimum Transmission around the Sink 116
4.6.2 Battery Threshold Value 117
4.7 Reliable Multipath Information Dissemination 119
by Label Forwarding (RM-IDLF) 119
4.7.1 Background 119
4.7.2 RM-IDLF Explanation 123

CHAPTER 5 PERFORMANCE EVALUATION OF THE DATA DISSEMINATION FRAMEWORK 130
5.1 Simulation scheme 130
5.2 Performance Assessment –IDLF 132
5.3 IDLF Evaluation 133
5.3.1 Consumed Energy over Time 133
5.3.2 Data transmission over time 137
5.3.3 Energy Consumption per Data Packet 141
5.4 Energy Management 143
5.4.1 Minimum Transmission around the Sink 144
5.4.2 Battery Threshold Value 146
5.5 Faults in Sensor Nodes 155

5.6 Performance Assessment - RM-IDLF 156
  5.6.1 RM-IDLF Evaluation 157
  5.6.2 Comparison of Label Dissemination Scheme with Flooding and SPIN 180

CHAPTER 6 CONCLUDING NOTE 184
  6.1 Conclusions 184
  6.2 Future Directions 187

APPENDIX 189
  Initialization of Algorithm 189
  Pseudo code for IDLF scheme 191
  Pseudo Code for the RM-IDLF 193

BIBLIOGRAPHY 197

VITA 214

LIST OF FIGURES

Figure1.1 Wireless sensor network ... 3
Figure 1.2 Architecture of a Sensor Node ... 4
Figure 1.3 UCB, SMART DUST Project [reproduced in consent with Seth Hollar] ... 8
Figure 1.4 Protocol Stack ... 10
Figure 1.5 (a) Miniature Wiring (b) Nose-on-a-chip sensor ... 19
Figure 1.6 Real time security using sensor networks ... 21
Figure 2.1 (a) Single hop routing model (b) Multi-hop routing model ... 26
Figure 2.2 Disparities in power spending ... 27
Figure 2.3 Cluster based hierarchical routing model ... 28
Figure 2.4 Classification of Routing Protocols [59] ... 30
Figure 2.5 (a) Flooding Techniques ... 32
Figure 2.5 (b) Implosion Problem ... 32
Figure 2.5 (c) The Overlap Problem associated with Flooding ... 33
Figure 2.6 (a) Observer / Source node acquiring Data, (b) Receiver sending ADV message, (c) Desired nodes sending REQ message and (d) Source sending the Data to selective neighbors ... 35
Figure2.7 (a) Directed diffusion operation - Diffusing interests ... 38
Figure2.7 (b) Directed diffusion operation- Setting up gradients ... 38
Figure2.7 (c) Directed diffusion operation - Path reinforcement ... 39
Figure 2.8 Example of virtual grid in GAF ... 59
Figure 2.9 State transitions in GAF ... 59
Figure 2.10 Relay region of transmit-relay node pair (i, r) ... 61
Figure 3.1 Marzullo Model ... 73
Figure 3.2 Iyengar Model ... 74
Figure 3.3 RBD diagram for two-single type spares ... 79
Figure 3.4 Markov model for two-single type spares ... 80
Figure 3.5 Markov model for two-pooled type spares ... 81
Figure 3.6 RBD diagram for three-pooled type spares ... 82
Figure 3.7 Markov model for three-pooled type spares ... 83
Figure 3.8 (c), (d) Reliability values for $\lambda = 0.05$, 0.10 failures per 10000 seconds ... 86
Figure 3.9 MTTF versus Cost for C=4, c_1=0.5 and c_2=0.75 ... 87
Figure 4.1 Topology of Sensor Network ... 96
Figure 4.2 Transmission range R1 and R2 ... 98
Figure 4.3 Example of Label Information ... 100
Figure 4.4 Label Propagation to immediate neighbors ... 100
Figure 4.5 Label Propagation –Implosion ... 100
Figure 4.6 Label Propagation – Path Rejection to avoid implosion ... 106
Figure 4.7 Label Propagation – Distribution over the network ... 107

Figure 4.8 Promulgation of labels over the network 107
Figure 4.9 Label path/s from Source to Sink 109
Figure 4.10 Request packet pursues the trace, on which the label traversed from the source to the sink. 109
Figure 4.11 Data Transmission 110
Figure 4.12 Label Propagation in all directions 111
Figure 4.13 Restricting Propagation of Labels in all directions 112
Figure 4.14 Directional flooding 112
Figure 4.15 Probabilistic analysis of simple label flooding 113
Figure 4.16 Probabilistic analysis of Directional Forwarding of labels 114
Figure 4.17 Disjoint Multipath Network 123
Figure 4.18 Braided Multipath Network 123
Figure 4.19 Label Transmission Path- RMIDLF 123
Figure 4.20 Label Propagation Path 126
Figure 4.21 Creation of a new Label Path 127
Figure 4.22 Creation of a new Label Path 127
Figure 4.23 Creation of Disjoint Path 128
Figure 4.24 No outlet condition 128
Figure 4.25 Node failure in path 1 128
Figure 4.26 Node failure in path 2 128
Figure 5.1 Sensor network design 131
Figure 5.2 Energy Consumed by entire sensor network over time - 5 nodes 134
Figure 5.3 Energy Consumed by entire sensor network over time - 10 nodes 134
Figure 5.4 Energy Consumed by entire sensor network over time - 15 nodes 135
Figure 5.7 Energy Consumed by entire sensor network over time - 30 nodes 137
Figure 5.9 Number of Data Packets Delivered at Sink over Time- 10 nodes 138
Figure 5.10 Number of Data Packets Delivered at Sink over Time- 15 nodes 139
Figure 5.11 Number of Data Packets Delivered at Sink over Time- 20 nodes 140
Figure 5.12 Number of Data Packets Delivered at Sink over Time- 25 nodes 141
Figure 5.13 Number of Data Packets Delivered at Sink over Time- 30 nodes 141
Figure 5.14 Energy consumed per Data Packet 143
Figure 5.15 Average Sensor Node Energy Consumption by Hop Count 145
Figure 5.16 (a) Number of Data Packets Delivered vs. Number of Nodes 600 unit energy 147
Figure 5.16 (b)Number of Data Packets Delivered vs. Number of Nodes 700 unit energy 148
Figure 5.17 (a) Number of Data Packets Delivered vs. Number of Nodes 800 unit energy 149
Figure 5.17 (b) Number of Data Packets Delivered vs. Number of Nodes 900 unit energy 149
Figure 5.18 Energy consumption in sensor network - 600 unit energy 151
Figure 5.19 Energy consumption in sensor network - 700 unit energy 151
Figure 5.20 Energy consumption in sensor network - 800 unit energy 152
Figure 5.21 Request packet pursues the trace, on which the label traversed from the source to the sink- RM IDLF Energy consumption in sensor network 900 unit energy. 152

Figure 5.22 Percentage of Data Packets Dropped- 600 unit energy 153
Figure 5.23 Percentage of Data Packets Dropped -700 unit energy 154
Figure 5.24 Percentage of Data Packets Dropped - 800 unit energy 154
Figure 5.25 Percentage of Data Packets Dropped - 900 unit energy 155
Figure 5.26 Energy Consumed by entire sensor network over time – Fault Free 159
Figure 5.27 Percentage of Data Packets Delivered at Sink over Time- Fault Free 160
Figure 5.28 Energy Consumed by entire sensor network over time – (10 % Fault)...... 160
Figure 5.29 Percentage of Data Packets Delivered at Sink over Time– (10 % Fault)... 161
Figure 5.30 Energy Consumed by entire sensor network over time – (20% Fault)....... 162
Figure 5.31 Percentage of Data Packets Delivered at Sink over Time – (20% Fault).. 162
Figure 5.32 Energy Consumed by entire sensor network over time – (30% Fault)....... 162
Figure 5.33 Percentage of Data Packets Delivered at Sink over Time – (30% Fault).. 162
Figure 5.34 Energy Consumed by entire sensor network over time – (40% Fault)....... 164
Figure 5.35 Percentage of Data Packets Delivered at Sink over Time – (40% Fault)... 164
Figure 5.36 Energy Consumed by entire sensor network over time – (40% Fault)....... 164
Figure 5.37 Percentage of Data Packets Delivered at Sink over Time– (50% Fault).... 165
Figure 5.38 Cost Performance Graph –RM-IDLF 166
Figure 5.39 Consumed Energy vs. Number of nodes – (Fault Free) 169
Figure 5.40 Consumed Energy vs. Number of nodes – (10% Fault) 169
Figure 5.41 Consumed Energy vs. Number of nodes – (20% Fault) 170
Figure 5.42 Consumed Energy vs. Number of nodes – (30% Fault) 170
Figure 5.43 Consumed Energy vs. Number of nodes – (40% Fault) 171
Figure 5.44 Consumed Energy vs. Number of nodes – (50% Fault) 171
Figure 5.45 Percentage of Data Packets Reaching Sink vs. Number of nodes (Fault free) 173
Figure 5.46 Percentage of Data Packets Reaching Sink vs. Number of nodes(10% Fault) 173
Figure 5.47 Percentage of Data Packets Reaching Sink vs. Number of nodes (20% Fault) 174
Figure 5.48 Percentage of Data Packets Reaching Sink vs. Number of nodes (30% Fault) 174
Figure 5.49 Percentage of Data Packets Reaching Sink vs. Number of nodes (40% Fault) 174
Figure 5.50 Percentage of Data Packets Reaching Sink vs. Number of nodes (50% Fault) 174
Figure 5.51 Average time to reach to sink vs. number of nodes (Fault free) 177
Figure 5.52 Average time to reach to sink vs. number of nodes – (10% Fault) 178
Figure 5.53 Average time to reach to sink vs. number of nodes – (20% Fault) 178
Figure 5.55 Average time to reach to sink vs. number of nodes – (40% Fault) 179
Figure 5.56 Average time to reach to sink vs. number of nodes – (50% Fault) 180
Figure 5.57 Average Energy consumed vs. Fault percentage.. 182
Figure 5.58 Data Packets reaching sink vs. Fault percentage.. 182
Figure 5.59 Time to reach sink vs. Fault percentage 183

LIST OF TABLES

Table 2.1 Comparison among SPIN, LEACH and Directed Diffusion 45
Table 2.2 Hierarchical vs. Flat topologies routing 56
Table 2.3 Categorization and Assessment of Routing protocols in Wireless Sensor Networks 70
Table 3.1 Popularities of the intervals that make up the output interval 75
Table 4.1 Sensor Network Characteristics 102
Table 4.2 Battery Threshold Value 118
Table 5.1 Energy consumed by transmitting each data packet 142
Table 5.2 Average Energy consumed per node at different hop location from the source node 145
Table 5.3 Percentage of Extra Energy Consumed in RM-IDLF in comparison with IDLF 165
Table 5.4 Percentage of Extra Data Packets delivered in RM-IDLF in comparison with IDLF 166
Table 5.5 Energy Consumed in RM-IDLF in comparison with IDLF by varying the number of sensor nodes 168
Table 5.6 Percentage of data packets delivered in RM-IDLF in comparison with IDLF by varying the number of sensor nodes 172
Table 5.7 Average time to reach the sink in RM-IDLF in comparison with IDLF by varying the number of sensor nodes 177
Table 5.8 Energy to packet ratio for each algorithm 181

ACKNOWLEDGEMENTS

I am indebted to my supervisor and examination committee chair Dr. Shahram Latifi for his valuable guidance, encouragement, and support throughout my Ph.D. study in academic as well as other related aspects for career development. His exemplary works on research and commitment to ethics have been sources of inspiration making it a privilege to work with him. I owe him more than a student-teacher gratitude. I would like to thank Dr. Henry Selvaraj for serving as examination committee member. His helpful suggestions helped me refine my written work. I have been fortunate to have received the assistance of Dr. Emma Regentova. Her active participation in my research and extensive discussions benefited me in my research. I would like to thank Dr. Sahjendra Singh to serve as examination committee member in my dissertation. I extend my gratitude to Dr. Wolfgang Bein, Professor, School of Computer Science for serving as a graduate college faculty representative. It would be unjustified not to thank the help of my colleague, Mr. Naoto Kimura, for his unremitting help in the research. I benefited from his expertise. I would like to thank Doina Bein for her constant help during my studies.

I am grateful to my wife, Manasi for her understanding, endless patience and encouragement at the time, when it was most required. Last but certainly not least, I am forever indebted to the love and care of my father who made me capable to accomplish his expectations.

CHAPTER 1

INTRODUCTION

Embedded systems technology has come a long way in the last few years. These systems have specific requirements and perform pre-defined tasks. The hardware-based applications of the past can be converted into sophisticated modules using embedded systems. Such systems generally use microcontrollers, or they may use custom-designed chips or both. The presence of ubiquitous computing in everyday life has been strongly felt due to seamless integration of discrete computing elements in the embedded systems [1]. They are used in maintaining the communication infrastructures, automobiles, machine tools, wide-range monitoring, space vehicles, cell phones etc. The uses are endless and the consumption of billions of microprocessors every year for numerous applications speaks for itself.

Advancements in CMOS integrated circuits and Bi CMOS micromachining has led to a positive expansion in micro-electro-mechanical system (MEMS) and wireless technology. These advances have abridged the size, energy requirements and the circuitry cost of the embedded systems. One such derivative of this momentum is wireless sensor networks. It consists of collection of sensor nodes deployed in a distributed fashion, within the given area for a specific set of application/s. These sensor nodes are minute, low power discrete devices that embed sensors and actuators with limited on board processing and ability to communicate over the wireless media (typically over an RF Channel). In this chapter we first describe the fundamental components, design requirements, challenges and limitations for such sensor networks. We highlight the use of wireless sensor networks over a broad range of applications. The thrust for designing routing protocols will be explained followed by the outline of the dissertation. The specific network architecture and related assumptions used in the rest of the dissertation are also introduced later in the chapter.

1.1 Wireless Micro- Sensor Networks

A wireless sensor network [2], [3], [4], [5] consists of a collection of minuscule wireless nodes with embedded sensors that are spatially diverse. Sensor nodes are appealing due to their autonomous ad hoc connectivity, ease of deployment and almost no dependency on any human intervention. The sensors are deployed in various environments and are capable of sensing and acquiring signals, processing signals, performing simple computational tasks and communicating with other nodes in a collaborative manner. The processing capability of each processing node is limited. Nevertheless, the coordination of information from a large subset of nodes makes it possible to measure a respective physical environment in great detail. The low cost of sensor nodes makes it possible to have a network of hundreds or thousands of these sensors nodes. One advantage of using the large number of nodes is to enhance the accuracy of the data retrieved and to make the system fault tolerant.

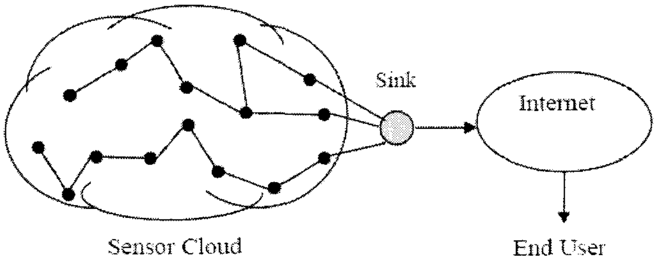


Figure1.1 Wireless sensor network

Wireless sensor network is a peer-to-peer multi-hop wireless network where data towards the sink is transmitted in a store and forward pattern via intermediary nodes, as shown in Figure 1.1. The sensor nodes are connected by radio frequency, infrared, or other medium without any physical wire connection. On detecting a physical phenomenon, a sensor node collects and processes the event data. The event data is then destined to sink node and traverses among other nodes in a wireless medium in a multi-hop fashion [6], [7]. Each sensor node has a radio range, which is referred to as the distance at which the signal strength remains above the minimum usable level for that particular node to transmit and receive. If two nodes, A and B cannot communicate directly, other nodes, located between those two nodes, transmit an information packet from the source node to the destination node. Information is received only by nodes within the radio range of the forwarding node in a wireless medium. Data communication in ad hoc networks uses intermediate nodes as routers. This can be well related to a single hop mobile network model that supports the needs of wireless communications by installing base stations as access points. Finally the data from sensor nodes is gathered by a sink node. Multiple sink nodes may exist in one wireless sensor network [8], [9].

Usually, the sink node is robust in terms of processing speed, battery capacity and memory size as compared to other sensor nodes in the network. End user can be connected to the sink using Internet or satellite to access the collected data.

1.2 Sensor Node Architecture

Sensor node includes a sensing module or a transducer, a small power unit, a microprocessor to process the sensed signals, a small memory unit to temporary hold sensor data, and a wireless interface to communicate with the other nodes (Fig 1.2). Depending on the application to be performed, a transducer translates a physical phenomenon to or from an electronic signal. Once the physical quantity has been sensed, the signal is fed to an A/D converter. The microprocessor takes this digital input and processes it, sending the ensuing data out to the network. Network communications are conducted by the interface block.

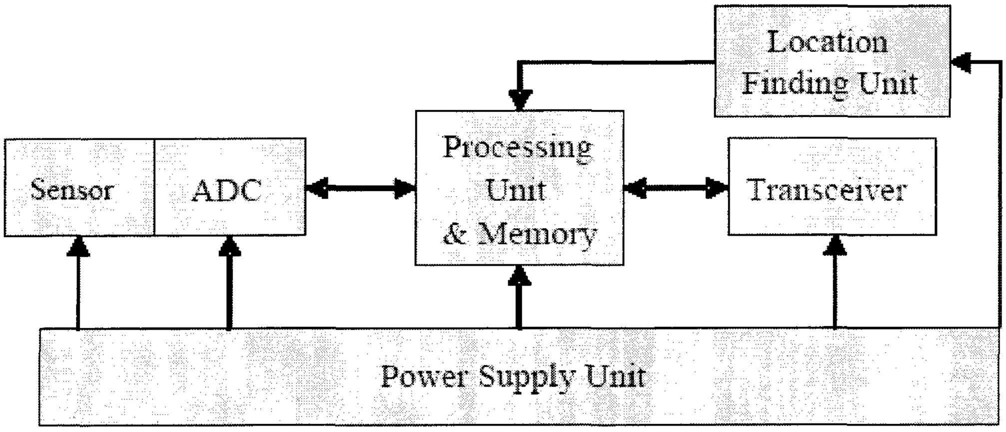


Figure 1.2 Architecture of a Sensor Node

Expected size of each sensor node is approximately 1 cm^3 in its volume and nearly 100 g in its weight. Its memory size, combining data and program memory, will be several tens of Mbytes. On the contrary, the sensor node currently available on the shelf [10] has its volume of 70 cm^3 with 128 KB of instruction memory, 4 KB of data RAM, and 512 KB of flash memory. The processor is operated at 4 MHz.

1.2.1 SMART DUST

Smart dust was envisaged in 1998 by Kris Pister of the UC Berkeley [11], [12]. Smart dust sets out to build a device with a sensor, communication device, and small computer integrated into a single package [14]. The Defense Advanced Research Projects Agency (DARPA)[1] funded the project, setting as a goal the demonstration *"that a complete sensor/communication system can be integrated into a cubic millimeter package"* [13]. "Smart dust" device is a tiny wireless micro-electro-mechanical sensors (MEMS) that can detect everything from light to vibrations. Recent innovations in fabrication techniques are leading these sensors "motes" to emerge to the size of a grain of sand. Each mote encapsulates sensors, computing circuits, power supply and a bidirectional wireless communication capability.

The goal of the Smart Dust project [15], [16], [17], [18] is to build a self-contained, millimeter-scale sensing and communication platform for a massively distributed sensor network. Figure 1.3 (a-h) illustrates some of the motes designed under smart dust project. The RF mote [19], [20], Fig.1.3 (a) was designed by Seth Hollar at UC Berkeley in the early part of year 1999. Since then, various projects have been

---

[1] The Defense Advanced Research Projects Agency (DARPA) is the central research and development organization for the Department of Defense (DoD) It manages and directs selected basic and applied research and development projects for DoD, and pursues research and technology where success may provide dramatic advances for traditional military roles and missions..

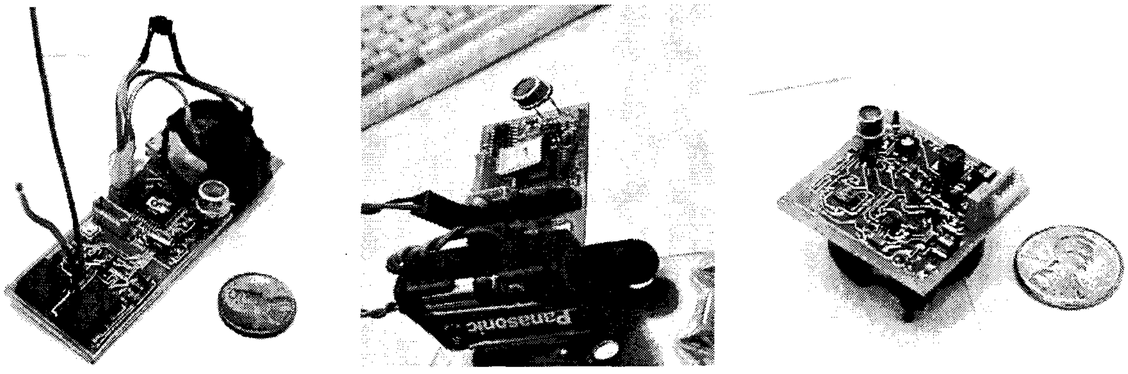
undertaken to test communication protocols for distributed sensor networks with the RF motes. It consisted of an Atmel AT90LS8535 processor, a 916 MHz RF transceiver and 5 sensors (temperature, light, barometric pressure, a 2 axis accelerometer and a 2 axis magnetometer). It operated on a 3V lithium coin cell battery that could support a mote for 5 days of continuous operation or 1.5 years at a duty cycle of 1%. The mote used a single radio carrier frequency to transmit data.

The RF mote had a communication range of about 5 - 30m at a rate of 5Kbps depending on other physical conditions. The Laser Mote, [19], [20], Fig.1.3 (b) has been used to reveal the long range communication abilities in a small silicon package. It uses an active laser communication to send sensor data over long distances. A Laser module acting as a transmitter from a laser pointer needs to be manually pointed towards the receiver. These motes can only send data back to a base station as they have no receiver module on board. The mote runs on 2 AA batteries and contains humidity, light, temperature and pressure sensors. Matthew Last et al [21] have demonstrated 21 Km one way communication from San Francisco to Berkeley. A CCD camera linked to a laptop computer was used as the receiver. However due to the slow speed of the camera data was sent at extremely low data rates but with commercial high speed camera data rates in excess of 1 Kbps are possible.

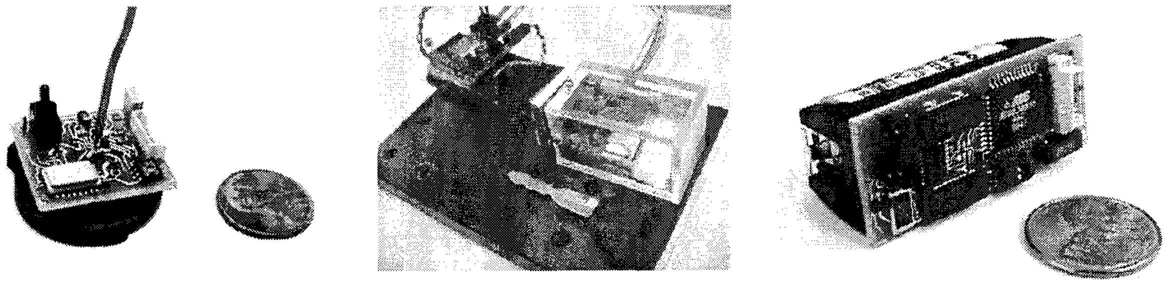
The Corner Cube Reflector (CCR) [19], [20], Fig 1.3 (c) is a MEMS device that allows for passive laser communications. It was designed at UC Berkeley by Seth Hollar and Farrah Santoso. The mote outfitted with a temperature sensor and a corner cube reflector (CCR) module allows passive laser communication. Initially an interrogator must project a swerve laser beam in the direction of the motes. This beam contains instructions to be performed by the motes. The motes receive this signal and by modulating and reflecting the beam back to the interrogator the mote can send back data. The communication range is a function of the laser beams intensity. The device is being used as a test platform to exhibit the communication algorithms that will ultimately be implemented on SMART DUST.

The Mini mote, [19], [20], Fig 1.3 (d) was designed by Seth Hollar and Christina Adela at UC Berkeley. It is a miniaturized edition of the RF Mote. Smaller size and simpler circuit design makes it cheaper and easier to handle. It has an Atmel AT90S2313 processor and an on-board temperature sensor. It can communicate via a radio link at 10 Kbps over a distance of 20m depending on the physical conditions. MALT (*M*otorized *A*ctive *L*aser *T*ransceiver), [21].Fig 1.3 (e) is designed by Sean Hubert. It was built in order to demonstrate steerable laser beam communication. MALT uses two linear actuators to tilt a plate with a mirror attached. Laser macro mote board drives the motors and laser, and collects data from the onboard light sensor.

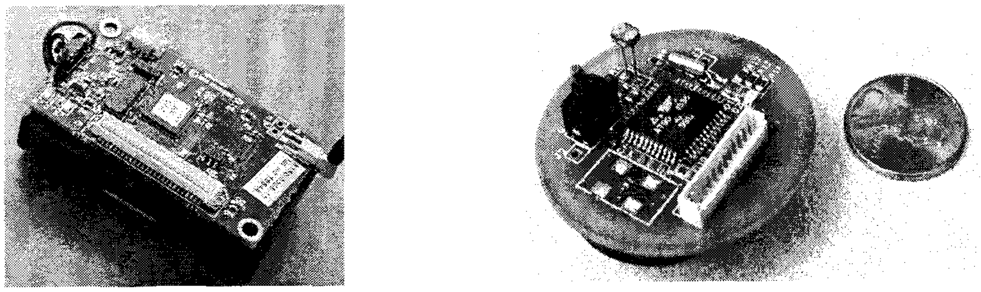
IrDA, [19], [20], Fig 1.3 (f) designed by Brett Warneke is designed to interface with other IrDA (Infrared Data Association) compliant devices, e.g. Palm Pilot. The ubiquity of the IrDA standard allows these motes direct communication with commercial technology, like Bluetooth. The Mica mote [22], Fig 1.3 (g) is designed by Crossbow in the United States. Mainly used for research and development of low power wireless sensor networks, it contains an Atmel Atmega 128L processor which is capable of running at 4 MHz. The device has a battery life of one year depending on the applications.



(a) RF MOTES (b) LASER MOTES (c) CCR MOTE



(d) MINI MOTE (e) MALT (f) IrDA



(g) MICA (h) weC

Figure 1.3 UCB, SMART DUST Project [reproduced in consent with Seth Hollar]

The mote is able to communicate with the sensor network via a radio link which operates on the 916 or 433 MHz bands and can carry data at 40 Kbps over distances of up to 100 feet. Figure 1.3 (g) illustrates weC [19], [20], Courtesy, Seth Hollar and James McLurkin at UC Berkeley. It is an improvement over the Mini mote which has a number of additions and a slightly larger size. weC is equipped with temperature and light sensors as well as an integrated PCB antenna to improve the motes communication performance. weC mote can be reprogrammed wirelessly. Currently, weC mote is being used in an EECS graduate class at Berkeley.

1.3 WSN Protocol Stack

The sensor network protocol stack is similar to the traditional protocol stack. In addition, this protocol stack blends (a) energy and routing awareness, (b) integrates data with networking protocols, (c) communicates energy efficiently through the wireless medium, and (d) encourages cooperative efforts of sensor nodes. [23], [24] The protocol stack in a sensor communication network has five layers: application, transport, network, data link, and physical layer (Fig 1.4) and three planes: power management plane, mobility management plane, and task management plane. Different application software can be developed and used on the application layer, depending upon the sensing task. The physical layer controls the transmission of data packets over the communication channel, which includes selecting a frequency, modulation, and demodulation. The minimum output power required to transmit over a distance "d" is proportional to "d" to the power of "n", where "n" varies from 2 to 4 and is closer to four when the antenna is close to the ground as is typical in wireless sensor networks. The main task of the data link layer is to make the data transmitted or received free from any errors. Data Link Layer helps in multiplexing data streams, data frame detection, medium access and error control. A wireless sensor network must have a dedicated MAC protocol to address the issues of energy conservation and data-centric routing. MAC protocol should satisfy two requirements in sensor network operation, (a) to establish a network infrastructure, which includes establishing communication links among possibly hundreds of nodes, and providing the network with self-organizing capabilities and (b) to share communication resources among all the nodes. Traditional MAC protocols do not meet these two goals because energy constraint is not prevalent in wired networks. Also, wireless sensor networks have no centralized control.

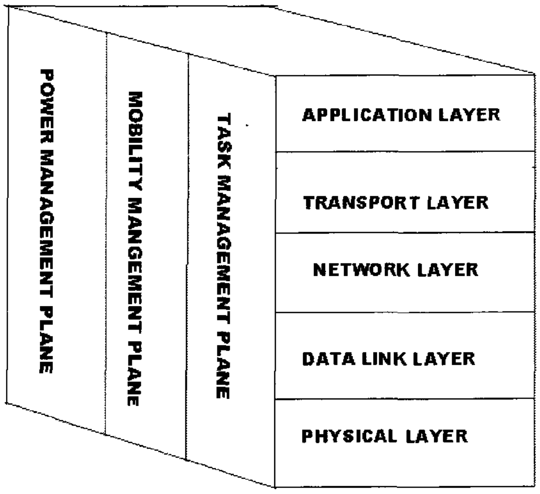


Figure 1.4: Protocol Stack

The MAC protocols proposed for the sensor networks are sensor-MAC (SMAC) [25], TRAMA (traffic-adaptive medium access protocol) [26], Etiquette Protocol [27], and CSMA for Sensor Networks [28]. The network layer is designed to perform routing among sensor nodes. The transport layer controls the flow of data. It manages each packet of data to arrive at the destination correctly. Finally, the application layer manages to handle application software, which will be varied depending on the task of the sensor network. The Power Management Plane is responsible for minimizing power consumption and may turn off functionality to preserve energy. The mobility management plane keeps track of the movement of sensor nodes. It detects and registers movement of nodes so a data route to the sink is always maintained. The task management plane decides which node(s) will be activated to carry out the sensing tasks among neighboring nodes. It ensures that the required nodes are assigned to do a task while the rest can focus their respective power on routing and data aggregation.

1.4 Wireless Sensor Networks and Traditional Ad Hoc Networks

Sensor networks are significantly different from the traditional ad hoc networks. One conspicuous difference is a stern constraint on power, computation, storage and bandwidth requirements. The nodes are deployed in an unattended arrangement and may not have a renewable energy source (except solar nodes, which is gaining popularity recently). It becomes imperative for nodes to stay alive on small, limited energy and communicate through a wireless communication channel. To prolong the lifetime of the sensor networks, the power aware algorithms and routing protocols are designed [29], [30], [31]. Output parameters like response delay, accuracy of the received information and network performance often take back seat over the energy efficient design. In case of traditional ad hoc networks such as cellular systems, the energy of the mobile nodes can be replenished. The number of sensor nodes in a sensor network can be several orders of magnitude larger than the nodes in an ad hoc network. Also the topology of a sensor network changes very frequently and the end user has a little control over the topology, unlike ad hoc networks. Due to the large amount of transmission overhead, sensor nodes within a sensor cloud do not use a global identification. In sensor networks, the transmission power levels are kept low. This is highly preferred in stealthy operations [23]. Compared to the traditional ad hoc communication, there is far less noise in signal propagation in sensor networks. This is mainly due to low power multi-hop communication pattern among the neighboring nodes.

1.5 Motivation and Design Challenges in Wireless Sensor Network

One of the main design goals of WSNs is to extend the lifetime of the network and utilize the resources efficiently. The motivation for employing aggressive power management techniques arises from the fact that sensor nodes are irreplaceable. Furthermore, there is no control on the topology of the sensor nodes, thus random deployment of the sensor leaves designer with acute design challenges. An extensive collaboration of sensor nodes is needed to execute high level sensing. Additionally, sensor networks should be highly reliable and fault tolerant for decisive applications. This section discusses about the way sensor networks should perform, the design challenges and the limitations.

- Wireless sensor networks need to be robust. The setting up of the network should be easy. The operating system of the sensor nodes must not be complex unlike conventional computers.
- Sensor networks should be application specific [32], which means some of the nodes, in addition to routing the data packets, performs application specific tasks to accommodate wide variety of applications
- The sensor nodes need to be densely deployed in a magnitude much greater than conventional ad hoc networks [33], [34]. All sensor nodes in a network are

broadly divided into different divisions or subsets each of which provides a blueprint of vital sensing. Different divisions take turns to being repeatedly switched on and off depending on a specific duty cycle. Thus, the remaining nodes in a dormant subset will remain asleep until the next desired action is chosen and the pattern is shifted. It is crucial to accurately estimate the availability of power for a chosen time interval. Nodes can be switched on and off, by choosing a specific duty cycle and a random phase difference. In terms of power consumption, operation of a wireless sensor node can be divided into three parts: sensing, processing, and transmission. Among those three operations, it is known that the most power consuming task is data transmission [35]. Approximately, 80% of power consumed in each sensor node is used for data transmission.

- Scalability is one of the most important factors governing the effectiveness of sensor networks. It is the ability of a network to adjust or maintain its performance as the size of the network increases. Diverse applications require using large networks, yet the performance of an ad hoc sensor network tends to degrade as the number of mobile nodes increases [36], [37]. Therefore there is a need for more scalable solutions.
- In a sensor network, it is crucial that the collected data is safely delivered to a desired destination. For conventional wired sensor networks, the flow of data packets and conditions of sensor nodes are usually monitored and controlled by centralized units. On the contrary, Wireless sensor networks are not equipped with centralized controlling unit for monitoring the entire network. One of the

primary reasons for delivery failure is the limited battery power in the sensor node. In sensor networks, information packets are disseminated hop-by-hop with each sensor node having limited information about its immediate neighbor. To minimize the transmission overhead, the neighboring nodes do not exchange their status information with each other; hence they are unaware of the battery status of next hop node. Therefore, it may occur that while a node is transmitting an information packet to its next-hop neighbor, the neighbor node runs out of battery, or the information sending node, itself, runs out of battery.

- It is often essential for the sensor communication to meet real-time constraints. In combat surveillance systems [45], communication delays within sensing and actuating loops directly influence the quality of enemy tracking. Due to the type of the wireless communication and unpredictable traffic patterns, it is infeasible to ensure hard real-time constrains, however, research that provides probabilistic guarantee for timing constraints is crucial.
- With the large amount of sensor nodes deployed for a single application, there is absolutely no control over the topology of the network. This, coupled with inaccessibility of human intervention strongly necessitates the ability of sensor networks to self organize. Neighboring sensors nodes therefore need to be able to self organize into sub network, and route the data and messages accordingly. For example data aggregation [38] is a self organization scheme at a higher, more abstract layer of functionality. Therefore, the network must be able to occasionally reconfigure itself so that it can continue to function. Individual

nodes may become disconnected from the rest of the network, but an extreme degree of connectivity must be maintained.

- In-network processing within the sensor nodes requires additional complexity to the sensor nodes and designing in-networking processing protocols with minimum overhead is a major concern. Traditional networks provide finest forwarding where the loss due to the bit error and buffer overflow is much lower as compared to the one in wireless networks. With different assumptions, such design philosophy is hard to hold in sensor networks. As a result, the wireless sensor networks should be highly reliable.
- Data centric processing is a basic consideration in the design of a sensor network. Sensor nodes are not assigned any global identifications like an IP address for the computers; instead, sensor nodes and the data are acknowledged through their respective (1) Contents (2) Location and (3) Constraints. Unlike conventional networks, maintaining a node addressing table, results in a large overhead. Instead the data queries are exchanged. For example in the task of monitoring a building's entrance, the request to gather the *"The total number of people going out"* is more appropriate than the request to *"gather the readings from the nodes A, B, C & D"*.
- Sensor network applications are driven by physical events, such as fire, inclement weather subsequently taking an unpredictable pattern. Node failures are common due to the sheer number of sensor nodes and the hostile environment. The radio media shared by densely deployed nodes is subject to heavy congestion and jamming. Further the communication become highly unpredictable due to low

bandwidth, high bit error ratio, and asymmetric channel. This affects the quality of service in the operation of sensor network.

- Sensor networks are susceptible to all kinds of attacks, such as eavesdropping, jamming and hacking. With constrained available resource, it is impracticable to deal with all possible security threats [48]; however some measures for expected attack must be dealt with while designing the sensor networks.
- Wireless Sensor Network (WSN) is transforming into a multi-service medium leading to the convergence of voice, video and data communications. Each type of service has a particular constraint and it has to be satisfied for the communication to be effective. Adding these with restricted power supply and dynamic hostile setting, many networks are established by distributing sensors over the area of interest. This strongly suggests that that these disposable sensor nodes have to be fairly cheaper in price.
- While in operation, some sensor nodes can fail due to lack of power, external damage, intrusion or environmental interference. The malfunction of sensor nodes should not affect the overall task of the sensor network. MAC layer and routing protocols must adjust the configuration of new links and routes to the data collection nodes. Dynamic regulation of transmission power and signaling rates on the existing nodes is required to reduce energy consumption. Furthermore the packets need to be rerouted through sections of the network where more energy is available. Multiple level of redundancy is needed in a fault-tolerant sensor network.

- Sensor nodes may be deployed in harsh environments such as disaster areas, a battle field, or ocean bed etc. Sensor network topology is prone to frequent change after the deployment [39]. Therefore the sensor nodes should be prone to adjusting itself according to the harsh physical and environmental challenges.
- Sensor networks should be adaptive to changes in network connectivity due to node failure. In a multi hop network, each node plays as a dual task of sender and receiver. Faulty sensor nodes can cause considerable topological changes and may call for re- routing information. The moment data is sensed, it should be delivered within a certain time span, before it becomes redundant. The transmission latency should be optimally designed for each application.
- Upon deployment of the sensor nodes in a network, practically all the nodes have an identical initial energy. Within a sensor cloud, variation in the rate of consuming power by each node depends on the various factors such as event sensing rate, distance from sink node, and location of each node relative to other nodes. This disparity in energy consumption in wireless sensor network causes an imbalance of node power status resulting in diminishing overall network lifetime. Sensor nodes should not be totally dependent on few fading energy nodes at the end.
- Position awareness is an important aspect of sensor nodes because the information collection depends on relative position between source and the sink nodes. Currently, due to hardware and bandwidth constraints, it is not possible to use global positioning system GPS for this purpose. Triangulation based methods [40],

where sensor nodes predict their position using Pico-radio strength from the known points.

1.6 Wireless Sensor Network Applications

The design and dimension of the wireless sensor networks highly depends on the kind of application for which the sensor nodes are used. They offer unprecedented prospect for a broad spectrum of applications, e.g. environmental and habitat monitoring, observing temperature, humidity, and barometric pressure of certain areas, equipment diagnoses, disaster management, and traffic control [6]. Varying range of sensor network applications made it feasible to design and tailor a network in such a way that it caters to the specific requirements of the application. There are some applications which require a continuous data updating from the network, like pressure reading, video monitoring etc. On the contrary, there are some applications where the network is inactive for a long period of time. As soon as an event occurs, the sensor nodes become active and data transfer is initiated [35] e.g., in earthquake monitoring system, the traffic is delay sensitive and bursty unlike video monitoring. There is a considerable set of different issues in designing the two above mentioned networks. Sensor networks can be classified into two main subsets as per their respective applications (a) Data gathering applications (b) Event detection applications [5], [35].

Culler et al classifies all these applications into three separate types [2]. The first type of applications monitor space e.g. applications like environment monitoring, agricultural monitoring, climate control, surveillance and smart alarms. The second type monitors objects, such as structural monitoring, equipment maintenance, asset tracking, and medical diagnostics. The third category monitors the interactions among things and the adjoining objects. It includes disaster management, wildlife habitat, ubiquitous computing environments, health care and mechanical process flow.

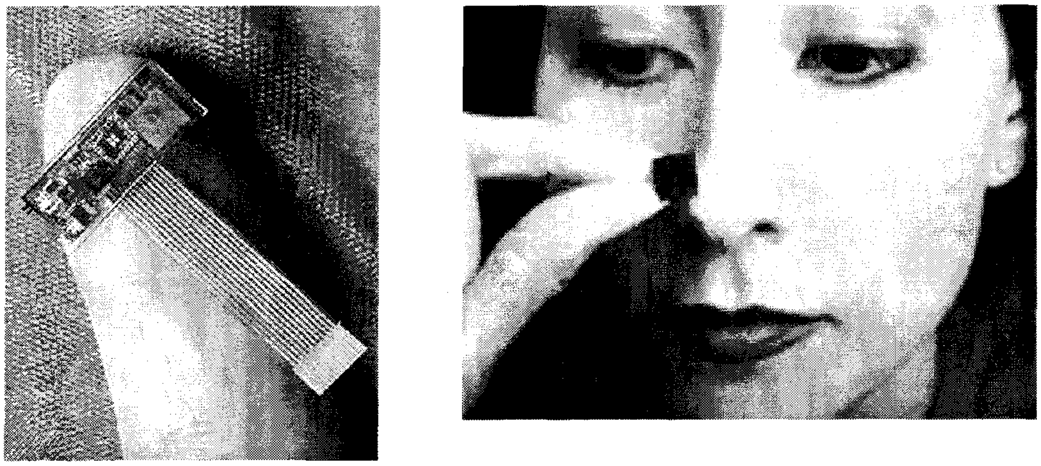


Figure 1.5 (a) Miniature Wiring (b) Nose-on-a-chip sensor

(Photo reproduced in consent with Oak Ridge National Laboratory Oak Ridge TN.)

Fig 1.5 (a) illustrates wiring on miniature sensors, making it evident that wireless technology is becoming more and more important as the sensors shrink. The wireless sensor nose-on-a-chip, fig 1.5 (b) is a MEMS-based sensor developed at Oak Ridge National Laboratory. It can detect 400 species of gases and transmit a signal indicating the level to a central control station. Military sensor networks are capable of detecting enemy movements, possible radiations and explosions. Therefore, large area surveillance and target detection applications are primary areas of defense research. These systems can use assorted collections of sensors to survey and report on various dynamic properties of the topography in a timely manner. An analogous solution utilizing traditional, wired technology either would be too costly or would produce long delays associated with trenching for fiber. Moreover, re-deployment or repositioning can be done quickly, and adding new nodes to an existing network is swift and effortless. This potential provides unmatched safety and recognition on the battlefield and helps reduce casualties even in the most traumatic situations. Brennan et al, developed a sensor array for radiation detection [44] using a massive amount of handy sensors to form an array. The gamma counts received indicate the sensor network approach provides higher sensitivity than traditional portal sensor. It is also portable and much cheaper. An urban shooter localization system is proposed by Matori et al [45]. An acoustic model of multiple sensors can detect the location of the shooter by generating the accuracy of 1 meter using 60 sensors.

Environmental sensor networks detect and monitor environmental changes and are deployed mostly to monitor weather changes, thunderstorms or wind motions. There is a robust challenge involved in designing a sensor network sustainable in harsh environment. GlacsWeb project [46] observes glacial ecosystem using embedded probe positioned inside the glacier and the base station is on the surface. Burrell et al designed the vineyard computing project [47] for agricultural monitoring to pull out the physical information in the vineyard set up. The "data mule" system consists of sensors to record temperature, humidity and weather. In addition, smart shovels trace workers activity. The collective data is then analyzed to provide suggestions on the performance and production optimization. Homeland Security has been a vital concern for federal and state governments. In recent years, the uninterrupted monitoring of public places of strategic importance has been very critical. Sensor networks deployed at vulnerable places, equip law enforcement agencies with an ability to synchronize data from other security systems with video images.

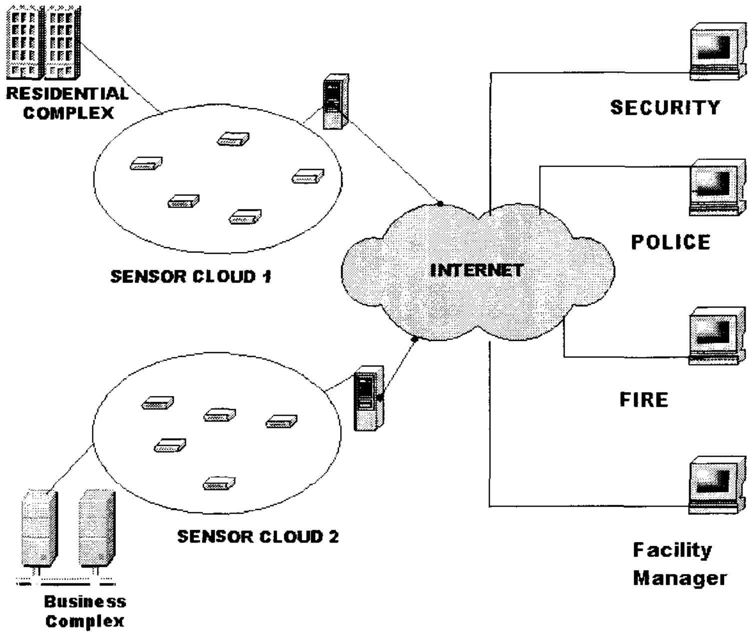


Figure 1.6 Real time security using sensor networks

Figure 1.6 depicts a real time association of security provisions with active sensor networks deployed at business as well as residential complexes. Information generated from these stand-alone, discrete components provides an ability to retrieve and view important data pertaining to a specific entry from a facial recognition system, or an access control system. Sensor networks can also be deployed at building walls to determine the wear and tear. Wireless parking lot networks can monitor the free spaces, issuing parking tickets, monitoring illegal activities etc. These networks improve emergency response and management activities.

1.7. Contribution and Scope of the Dissertation

This dissertation primarily addresses several issues related to designing a reliable and energy efficient schemes for extending the lifetime and fault tolerance of the network. Since sensor networks are resource constrained in terms of power, bandwidth and computational capability, an optimal system design radically changes the performance of the sensor network. In this research, a comprehensive information dissemination scheme for wireless sensor networks is proposed. Two main research issues are considered: (1) a collaborative flow of information packet/s from the source to sink and (2) energy efficiency of the sensor nodes and the entire system. For the first issue, a new scheme called Reliable Multi-path Information Dissemination via Label Forwarding (RM-IDLF), which is a reactive and on-demand routing paradigm for distributed sensing applications, is designed and evaluated. RM-IDLF incorporates point to point data transmission where the source initiates the routing scheme and disseminates the information toward the sink (destination) node. Prior to transmission of actual data packet/s, a data tunnel is formed followed by the source node issuing small label information to its neighbors locally. These labels are in turn disseminated in the network. By using small size labels, RM-IDLF avoids generation of unnecessary network traffic and transmission of duplicate packets to nodes. We study the trade-offs between the achieved routing reliability using multiple disjoint path routing and extra energy consumption due to the use of additional path/s. The effect of the failed nodes on the network performance is evaluated within the sensor system. Performance of the scheme is evaluated and compared with the classic flooding and SPIN [41], [42], [43].

For the second issue, we proposed discrete energy efficient schemes, which are incorporated in the system in conjunction with RM–IDLF. Setting up a battery threshold ensures that data packets will not be dropped after the sensor node's battery level falls below the threshold value. Minimum transmission around the sink prevents fast energy dissipation of the neighboring nodes to the sink. Finally, directional forwarding is applied to RM-IDLF. In directional forwarding, the sensor nodes narrow the range of broadcasting data packets by restricting communication only to the nodes lying in the direction toward sink/s. The rest of the thesis is organized as follows. Chapter 2 provides an overview of the related work on routing protocols in wireless sensor networks. In this chapter we motivate the need of an energy aware routing infrastructure and explain relevant design issues to be considered for building a routing framework for sensor networks. Chapter 3 explains fault tolerance techniques in general and explores the reliability issues in multifusion sensor networks. Chapter 4 provides a detailed discussion of the current information dissemination scheme RM-IDLF. Analytical and simulation results for the algorithms proposed are included in Chapter 5. The comparison of the current scheme with existing routing scheme is also conducted. We demonstrate the effectiveness of the new scheme in improving network lifetime and overall network reliability. Finally, Chapter 6 provides the conclusion of this dissertation and directions for future work.

CHAPTER 2

ROUTING MANAGEMENT IN WIRELESS SENSOR NETWORKS

Sensor networks are broadly deployed to sense, examine and manage the physical environment/s from remote locations. The precision of the information exchange is greatly enhanced with the alliance of sensor nodes and the reliable routing of the sensed data. The functionality of the routing protocols might vary depending on the sensor network architecture and the application. A daunting challenge in the design of a reliable wireless sensor network is to augment its lifetime in terms of energy and information efficiency. Therefore, it is desired to save energy of the sensor nodes while routing query responses back to the sink node. This may either be accomplished by cutting down the number of nodes or incorporating sleep periods, when nodes are not participating in transmitting data on the path ([49], [50]). In terms of power expenditure, operation of a sensor node can be categorized in three phases: sensing, processing, and transmission. Among these three phases, it is known that the most power consuming task is data transmission. Approximately, 80% of power consumed in each sensor node is used for data transmission. Energy-aware routing algorithms [51], [52], [53] discuss reducing the consumption of battery-power at different nodes. Another concern is the narrow computing power of the sensor nodes and the limited bandwidth [41] of the connecting nodes, which deter the communication of sensor nodes within the Wireless sensor cloud. Other challenging design requirements are lack of a centralized awareness of the network topology, scalability due to large network size and fault tolerance due to frequent failure of nodes. An optimal objective is to design routing schemes which (a) minimize energy requirements at each node to transfer individual packets and (b) maximize the operational lifetime of scalable networks. This chapter sets up the foundation of the research work proposed in this dissertation and presents a comprehensive investigation of different routing schemes. Depending on the sensor applications, the design challenge, advantage and performance concern for each routing protocols is also revealed.

2.1 Routing Techniques in Wireless Sensor Networks

In this section, we review the state-of-the-art routing techniques for wireless sensor networks. The routing scheme for wireless sensor networks has to be straightforward and simple which does not expend much computation power and memory, and eventually minimize communication among nodes to save its power.

2.1.1 Routing Models

Routing protocols may be classified into one of the ensuing three models [54] (a) single hop model, (b) multi-hop model and (c) Cluster-based hierarchical model. We will discuss each model briefly and further classify the protocol based on network structure and protocol operation in the subsequent sections. Single hop model is the simplest model and act as a direct communication model. In this model, fig 2.1 (a), all the nodes travel one hop to reach to a base station or the sink node. This kind of single hop transmission is highly unrealistic in the real world. The transmission range of each node and the energy consumed plays a crucial role in defining the sensor network. The multi-hop model supports the collaborative effort of several nodes within the sensor cloud, fig 2.1 (b).

Each sensor node has a radio range, which is referred to as the distance at which the signal strength remains above the minimum usable level for that particular node to transmit and receive. If two nodes cannot communicate directly, the nodes positioned between those two nodes, transmit an information packet from the source node to the destination node. Information is received only by nodes within the radio range of the forwarding node in a wireless medium. In view of efficient energy consumption, this model follows more practically feasible approach and is employed by [41], [55], [56], [57], [58]. The multi hop model uses the data aggregation techniques.

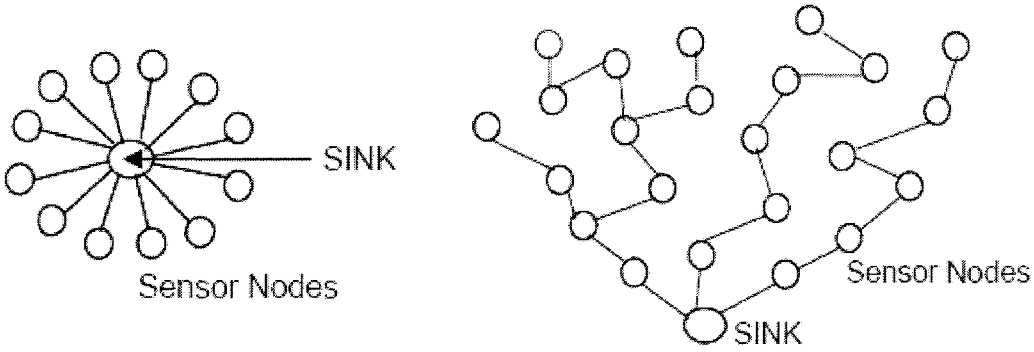


Figure 2.1 (a) Single hop routing model (b) Multi-hop routing model

Within a sensor cloud, variation in the rate of consuming power by each node depends on factors such as event sensing rate, distance from sink node, and location of each node relative to other nodes. This disparity in energy consumption in wireless sensor network causes an imbalance of node power status (figure 2.2) resulting in diminishing overall network lifetime. If the sink node is at one fixed location, information packets gather from the entire network to one fixed sink. This result in denser information traffic around the nodes in vicinity of the sink, as compared with the nodes placed farther from the sink. Hence, the nodes close to the sink will exhaust energy at the faster pace. If the nodes around the sink drain their energy, the sink is isolated from the entire sensor network, thereby making the data collection impossible. The segregation of the sink node from entire network is called self induced black hole effect [54]. To avoid isolation of sink node from the network, it is necessary to adopt an energy conservation heuristic on nodes located around the sink.

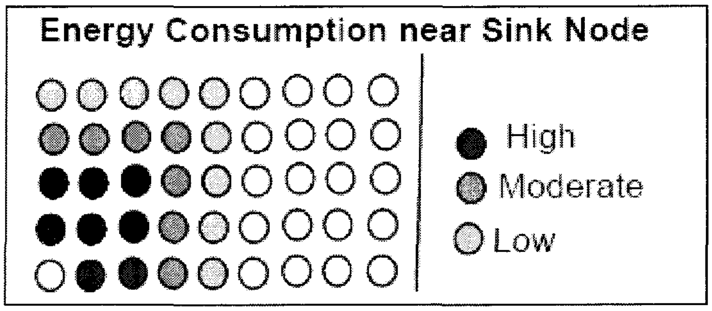


Figure 2.2 Disparities in power spending

In the cluster based model, fig 2.3, the network is divided into clusters comprising of "X" amount of nodes. Cluster head, which is master node, within each respective cluster is responsible for routing the information to the other cluster head. Data is first aggregated within the cluster and then from cluster to cluster. As the data packets moves from one cluster to another, it covers larger distances. This results in very low data latency as compared to multi hop model and single hop model respectively. However this model has a drawback. As the distance between clustering levels increases, the energy spent grows proportional by the square of distance. This definitely increases the energy consumption of the sensor network.

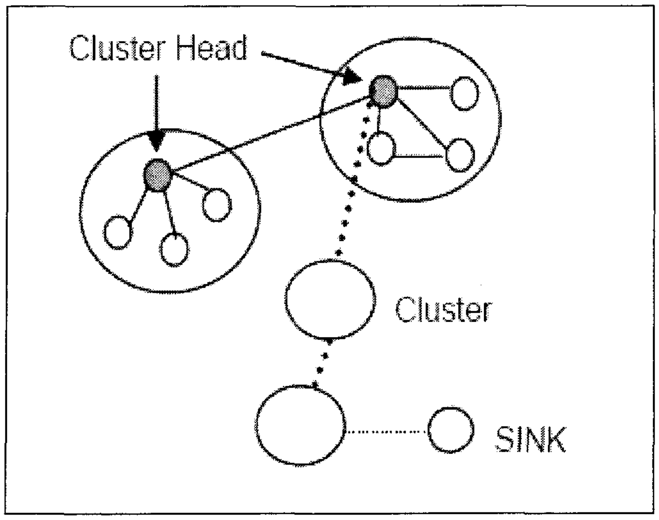


Figure 2.3 Cluster based hierarchical routing model

2.1.2 Protocol Assessment

In traditional networks, the focus is on Quality of Service (QoS). In wireless sensor networks QoS requirements can be relaxed to preserve energy and the network lifetime. At each layer of protocol stack, steps must be taken to (a) save energy, (b) allow sensor nodes to reconfigure, and (c) update their respective tasks according to the resources available. The simulation test bed should be as simple as possible. Diverse environmental conditions need to be implemented in analyses and simulations [54]. Some of the parameters used to evaluate the routing performance are (a) Energy consumption, both with respect to an individual node and the whole network (b) Simulation time and latency, (c) success rate of the data packets reaching the sink, (d) network size, and finally (e) fault tolerance capability of the entire network. In addition, the routing protocol should incorporate some kind of security to evade vulnerability from adversaries.

2.1.3 Routing Methodology in Wireless Sensor Networks

In WSNs, discovering the routes and then sustaining them is practically insignificant because of the energy constraints and sudden node failures. First of all, there is no control on topology of the nodes within a sensor network cloud. Secondly the unpredictable topological change makes it impossible to stick to a fixed routing strategy. Some well known routing plans such as data-centric methods, in-network processing, clustering, data diffusion, data aggregation and energy aware methods are proposed in the literature to cater the requirements for wireless sensor networks. Figure 2.4 illustrates the lineage of the routing protocols in wireless sensor networks. Broad classification is on the basis of

- Network structure and
- Protocol operation

Based on the network structure, the protocols are further classified as flat-network routing, hierarchical-based routing, and location-based routing. Flat routing protocols distribute information as needed to any reachable sensor node within the sensor cloud. No effort is made to organize the network or its traffic, only to discover the best route hop by hop to a destination by any path. All nodes are assigned uniform functionality. In hierarchical-network routing, nodes play different roles in the network. These protocols often group sensor nodes together by functionality and merge them into a hierarchy. Location-based routing uses the physical position of a sensor node in the network to route packets to that node. If a node changes location, the connection to that node will be broken and another route is required to establish its new location. Based on their operation, the protocols are further classified into Query based, negotiation based, multipath based, QoS based and coherent routing. The protocols categorized here often overlap on top of each other. We will explain each relevant protocol on the basis of network structure and protocol operation respectively.

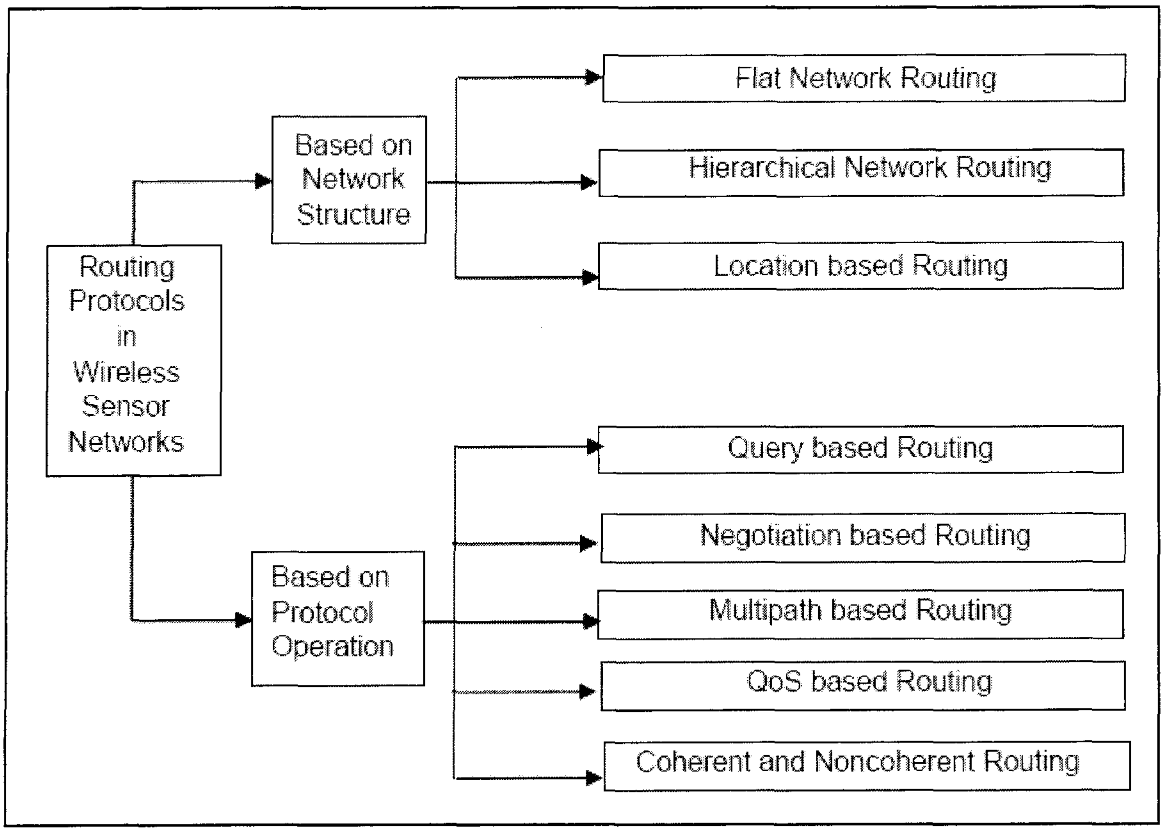


Figure 2.4 Classification of Routing Protocols [59]

2.2 Flat Routing

In flat network architecture all the sensor nodes are equal and connections between nodes are set up in short distance to establish the radio communication. Route discovery can be carried out in sensor networks using flooding or broadcasting which do not involve topology maintenance. In this section, we will provide an in-depth knowledge of the flat routing based protocols.

2.2.1 Flooding

Flooding is an example of simple flat routing scheme. When a sensor node receives a data packet, it stores the data and broadcasts it to its neighboring nodes. This process repeats until the information reaches all the sensor nodes in an entire network. To perform flooding, figure 2.5 (a), sensor nodes do not need any knowledge of the network configuration. Sensor nodes distinguish each data packet, while receiving or transmitting a data packet. This will save the limited memory space of each node. Since flooding does not require any complicated routing algorithms, it can be easily implemented for sensor networks. However, there are some shortcomings in this scheme which dissipate the limited resources of the sensor nodes [41].

One such problem in classic flooding is implosion. Implosion occurs, when a data receiving node broadcast the data packet to its all neighboring nodes, irrespective of whether the neighboring node already has the same data or not. Figure 2.5 (b) illustrates implosion; here node D floods the information to its neighboring nodes E and F respectively. Node H being the neighbor to both E and F, gets the same copy of the information from both E and F. Due to indiscriminate transmission of data, sensor nodes in this scheme expend limited transmission energy and bandwidth. Another problem associated with flooding is overlap. This situation occurs when multiple nodes observe the same sensor region, they generate the overlapping data. The sensor nodes within its neighborhood receive the multiple copies of the same packet containing the same information. Overlap, like implosion expends transmission power and bandwidth. Figure

2.5 (c) shows that J and K collect overlapped information and flood to their respective common neighbor/s, here node L. In flooding, the overlap problem is more difficult to solve than the implosion problem, because implosion is a function of network topology, whereas overlap is a function of both topology and observed data [41]. Additionally, in classic flooding, the sensor nodes are not resource aware, i.e. sensor nodes do not update their activity status according to the energy constraints at any given time. These shortcomings reduce the battery life of sensor node and therefore shorten the entire network life span.

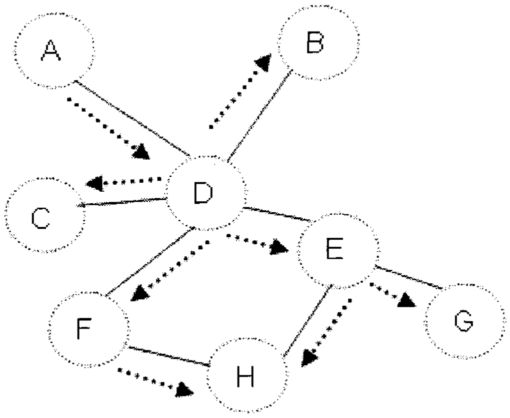


Figure 2.5 (a) Flooding Techniques

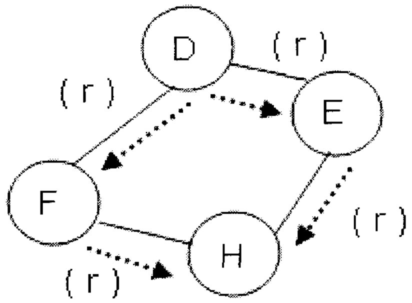


Figure 2.5 (b) Implosion Problem

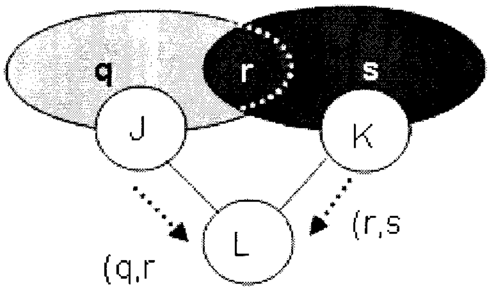


Figure 2.5 (c) The Overlap Problem associated with Flooding

Nonetheless, Flooding, due to its easier implementation and simple design has been investigated intensively to overcome the abovementioned shortcomings. In [43] and [60], each sensor node only needs to know a small portion of entire network configuration, which is the location information of its neighboring nodes, instead of the information of entire network topology. Gossiping is another by-product of flooding [42].When a node receives the data packet it selects a subset of its all neighboring nodes and transmits the data packet to the subset, instead of all neighboring nodes. This reduces the consumption of transmission energy.

2.2.2 SPIN- Sensor Protocols for Information via Negotiation

Another example of flat routing is negotiation based protocols. Sensor Protocols for Information via Negotiation- SPIN [41] intend to disseminate data towards the sink using negotiations. It is assumed that the source has an observed data meant to be transported to sink node. The source node advertises its data over the sensor network. Those nodes desiring the sensor data, request it from the source. For the negotiations, the information descriptors called "meta-data" are used. Upon sensing the information packet, figure 2.6, the observer or source node transmits a small advertisement packet (ADV) to its all neighboring nodes except the one from which the node receives the data packet. The ADV contains the information of actual data. Upon receiving the ADV, a neighboring node checks its local cache whether the node already has the same data or not. If the neighboring node already has the data, the ADV is rejected. If the node does not have the desired data, it sends a request message (REQ) to the receiving node. Then, the receiving node transmits the data packet (DATA) to the neighboring nodes, which request the data by sending the REQ message.The corresponding neighboring node then replicates this procedure with its neighbors. As a result, the entire sensor network will acquire a copy of the data. This guarantees that there is no redundant information sent throughout the network. The SPIN family of protocols includes several schemes with minor modifications on the actual proposal [41].

SPIN-1 includes negotiation before transmitting information to guarantee that only useful information will be transferred. It is a three-way handshake protocol, as mentioned above. SPIN-2 is a modification to SPIN-1 which in addition to a three- way handshake includes a resource awareness mechanism [59]. SPIN-2 works under resource constraint environment. Each sensor node has its own resource manager, which keeps account of the expended and remaining power. Before each transmission, the nodes examine their resource manager and curb on other energy expending activities to increase the lifetime of the node.

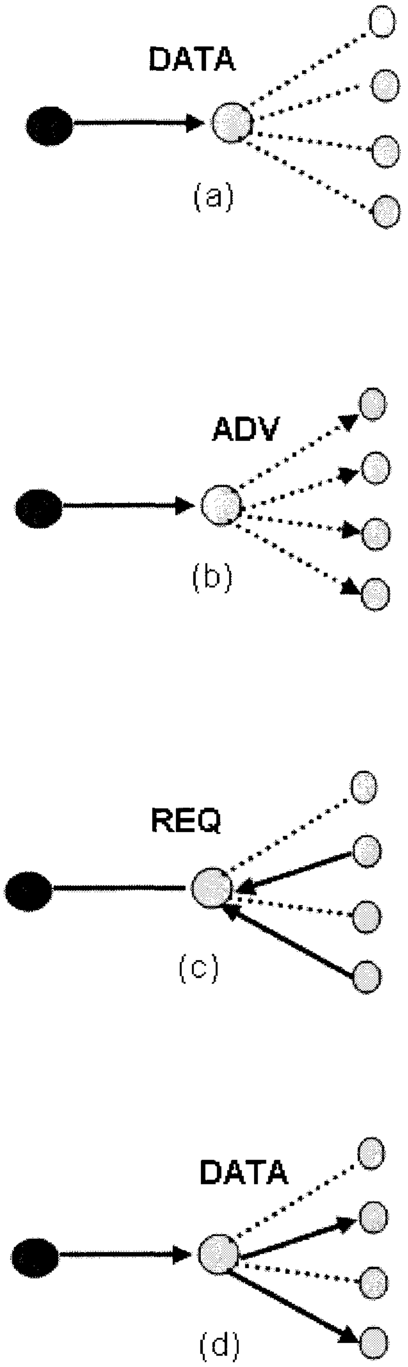


Figure 2.6 (a) Observer / Source node acquiring Data, (b) Receiver sending ADV message, (c) Desired nodes sending REQ message and (d) Source sending the Data to selective neighbors

SPIN-BC is developed for broadcast networks in which the sensor nodes use a single shared channel to communicate with each other. In SPIN-BC, the sensor node on receiving the ADV message does not send a REQ message instantaneously. Instead, it waits for a certain time before sending the REQ message. This is to avoid the redundant request for the same message. In SPIN-RL, each node keeps track of all the nodes from where it hears the advertisement. If it does not receive any requested data within a certain period of time, it sends out the request again. Similarly after transmitting the data message, sensor node waits for a certain period of time before responding to other requests for the same data message. Mainly SPIN-RL is used for the lossy channels. SPIN-PP has been developed to work with point-to-point communication. In SPIN-PP, two nodes can have a direct communication, without the need for intermediate nodes. It is a simple 3-way handshake scheme in which energy is not considered to be a constraint. [62]. In SPIN-EC the sensor nodes follow the 3-way handshake like SPIN-PP but there is an energy-conservation heuristic added to it. Sensor node contributes actively in the protocol only if it is above a certain energy threshold and believes it can complete all the other stages of the protocol. Performance evaluation of SPIN [41] demonstrates that SPIN is more energy-efficient than flooding or gossiping while distributing data at the same rate or faster than these protocols. However the SPIN suffers from the weakness [63] of transmitting all the data packets at the same Energy level and not using the distance to a neighbor to adjust the energy level. Besides a large overhead in broadcasting the data, energy consumption is a concern in SPIN. The motivation for developing the Label dissemination forwarding, IDLF and RM-IDLF schemes surfaced from the above-mentioned limitations of currently employed schemes.

2.2.3 Directed Diffusion

Directed diffusion, [57], [64] is a data-centric and application-aware routing scheme where all the information generated by sensor nodes is named by attribute-value pairs. This is a Sink-initiated reactive routing paradigm in which routes are established as they are requested. In data centric routing the data originating from different sources are combined with in-network aggregation by eliminating redundancy, minimizing the number of transmissions; thus saving network energy and prolonging its lifetime. Unlike conventional networks, maintaining a node addressing table, results in a large overhead. Instead the data queries are exchanged. For example in the task of monitoring a building's entrance, the request to gather the *"The total number of people going out"* is more appropriate than the request to *"gather the readings from the nodes W,X,Y,Z"*. The sink node requests data by broadcasting "interests" or sensing task. Interest specifies the sensing task; include type of sensing event, sensing area, duration of sensing task, and event transmission frequency. Figure 2.7 illustrates the operation of directed diffusion. The interest is disseminated throughout the network in a hop-by-hop manner. The query is initiated by the *sink* node and it broadcasts its interest message periodically to all of its neighbors. An interest cache is maintained by each node. When a node receives an interest, it stores the interest and also sets up a gradient toward the node, from which it received the interest. This process continues until gradients are established from the sources back to the sink. If a respective node has the requested data, which matches the received interest, the node sends back the data packet to the sink in multiple paths according to the gradients. On receiving the data packet at the sink, the reinforcement of the optimal path is initiated by the sink. The criterion for the selecting the optimal path highly depends on the application. It may be the shortest path or minimum energy consuming path, whichever suits the application. The best path is reinforced by the sink sending a new interest to the path.

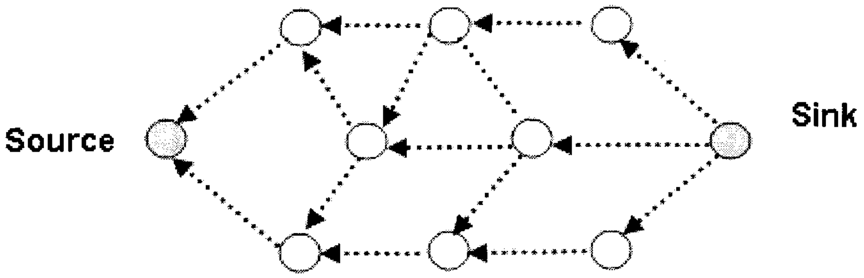


Figure2.7 (a) Directed diffusion operation - Diffusing interests

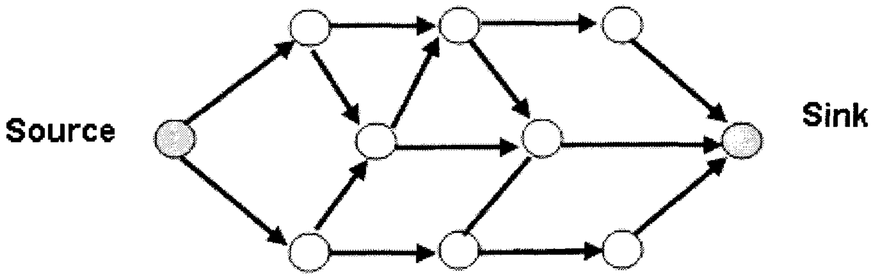


Figure2.7 (b) Directed diffusion operation- Setting up gradients

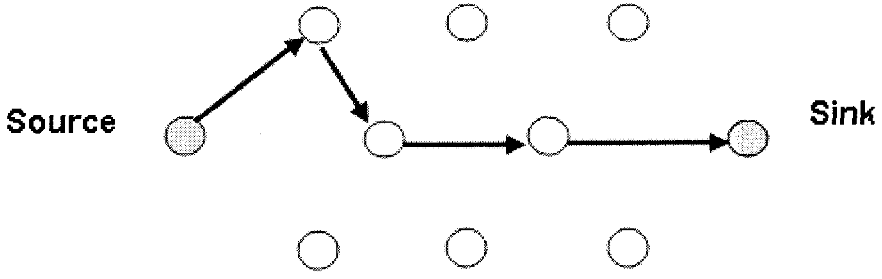


Figure2.7 (c) Directed diffusion operation - Path reinforcement

In flooding and SPIN, data collection is initiated by source nodes. In other words, source nodes start transmitting data whenever an event is observed. On the contrary, in directed diffusion data collection is initiated by sink nodes. Because of sink initiated data collection, directed diffusion can limit data flow. By doing so, it will reduce unnecessary data transmissions and thus energy consumption of sensor nodes will be reduced. Such type of information retrieval is appropriate for continual queries where requesting nodes do not anticipate data that fulfill a query for duration of time. Possibility of transmission overhead created by interests creates another disadvantage of using this scheme. When a sink broadcasts an interest, the sink does not know whether the data, which will match the interest, is obtainable or not. If the data is not available at that time, sink node can not collect any data at all. Furthermore, this makes it unsuitable for one-time queries, as it is not worth setting up gradients, which use the path only once. For instance, directed diffusion is not applicable data dissemination scheme for surveillance purpose because sensor nodes have to transmit data as soon as they detect abnormality.

2.2.4 Rumor Routing

Rumor routing [65] is a data centric scheme proposed for applications where geographic routing is not practically possible. Rumor routing directs the queries to the sensor nodes that have observed a specific event rather than flooding the entire sensor network to extract information about the occurring events. The rumor routing utilizes a set of long-lived agents which create paths that are directed towards the events they encounter. On detecting an event, sensor node adds it to its events table, and generates an agent. Agents travel within the sensor network to disseminate information about local events to distant nodes. When a node generates a query for an event, the nodes that know the route, may reply to the query by inspecting its respective event table. The agent has a lifetime of a specific amount of hops after which it dies. Any node creating a query will transmit the query if it has a route to the event, otherwise it will transmit it in a random direction. If the node discovers that the query did not reach the destination, then it will flood the network. The fewer the number of queries which flood, the less is the energy consumed. Unlike directed diffusion where data can be routed through multiple paths at low rates, rumor routing keeps only one path between source and destination. Simulation results [65] show that rumor routing can handle the node failure and achieve major energy savings when compared to flooding. Rumor routing performs well as long as the number of events is small. For the large number of events, the cost of generating and maintaining the event tables and agents results in a large overhead [59].

2.2.5 COUGAR

COUGAR [66] is a data- centric protocol and follows the directed diffusion model along the database approach. COUGAR utilizes in-network data aggregation for power saving. There exists an additional query layer between network and application layer. It abstracts the information generated by network in an update-only relational table. The attribute in this relational table is (a) details about the sensor node, for example its location, ID and (b) information collected from respective node (e.g., temperature, light). Sensor applications are often interested in summarized and consolidated data that are produced by aggregated queries rather than detailed data. In addition, sensor nodes choose a leader node to initiate the aggregation and transmit the information to the sink. Sink is liable for generating a query, which specifies the details about the data flow and in-network computation for the incoming query and sends it to the appropriate nodes. In COUGAR, sensor readings are treated like "virtual" relational database tables and a query language like SQL may be used to issue tasks to the WSN. COUGAR has some drawbacks [59]. First, inclusion of query layer on each sensor node puts an extra overhead on the nodes and the entire network in terms of power and memory storage. Second, a high level of synchronization among the sensor nodes is required to achieve the in-network data computation.

2.2.6 ACQUIRE

ACQUIRE [67] stands for active query forwarding in sensor networks. It is a data-centric, application specific scheme for querying wireless sensor networks. In ACQUIRE an active query is passed through the sensor network. The intermediate sensor nodes use cached local information (within the look ahead of "d" hops) to partially resolve the query. When the query is resolved an entire response is sent directly back to the querying node. For the complex queries, directed diffusion may not be the right choice, because it uses flooding based query mechanism, which would expend energy. ACQUIRE can adjust the look ahead parameter "d" to offer an efficient querying [59]. When d is equal to the network diameter, ACQUIRE performs similar to flooding. If "d" is too small, the query has to travel more hops. ACQUIRE performs better than directed diffusion and the optimal ACQUIRE can lessen the energy consumption by more than 60% as compared to expanding the ring search.

2.3 Hierarchical Routing

Due to the nature of the applications supported by the sensor networks such as a range of estimations measuring temperature, pressure, humidity, seismic, thermal, acoustic, radar, noise levels etc, the sensor nodes need to be densely deployed in a magnitude much greater than conventional ad hoc networks [68]. In hierarchical routing, the nodes with the higher energy can be utilized to process and transmit the information. The low energy nodes can be assigned sensing in the proximity of an event. This routing uses the fact of division of labor, among the sensor nodes. Depending upon the remaining energy, the task to each node can be assigned accordingly. The formation of clusters within the sensor network, allows the sensor nodes to make the decision to choose the cluster leader. This enhances the network lifetime, energy efficiency and scalability of the sensor networks. According to [59], hierarchical routing consists of two layers where one layer is used to select cluster heads and the other layer is used for routing decision. This section explains some of the hierarchical routing schemes

2.3.1 Low Energy Adaptive Clustering Hierarchy- LEACH

LEACH is an energy conserving communication protocol [13] where all the nodes in the network are uniform and energy constrained. An end user can access the remotely monitored operation, where large numbers of nodes are involved. The nodes organize themselves into local clusters, with one node acting as the randomly selected local cluster-head. If the allocated cluster-heads are always fixed, then they would die quickly, ending the useful lifetime of all nodes belonging to those clusters. LEACH includes random alternation of the high-energy cluster-head nodes to enable the sensors to uniformly sustain the power. Sensors nominate themselves to be local cluster-heads at any given time with some probability. These cluster head nodes relay their status to the other sensors in the network. Each sensor node resolves which cluster to follow by choosing the cluster-head that requires the minimum communication energy. This allows the transceiver of each unassigned node to be turned off at all times except during it's transmit time, thus minimizing the energy dissipated in each sensor. LEACH operates in two phases (a) the initializing or set up phase, where the organization of clusters and selection of cluster heads takes place, and (b) steady state phase, where the actual data transfer takes place. During the set up phase a set of nodes, "p", nominate themselves as cluster heads respectively. A random number "r" between 0 and 1 is selected by the sensor node. If this random number is less than the threshold value, T (n), the respective node becomes the cluster head for the particular event. The calculation of threshold value T (n) is shown below, G is the set of nodes that were not accepted as cluster head in the last "1/p "events

$$T(n) = \frac{p}{1 - p(r \bmod (1/p))} \qquad \text{If } n \in G$$

Each nominated cluster head advertises to the rest of the nodes in the network about its status. After receiving the advertisement, the non-cluster head nodes decide as to which cluster they want to fit in. This assessment is based on the signal strength of the advertised message. The signal to noise ratio is compared from various cluster heads surrounding the node/s. The non cluster-head nodes notify the respective cluster-head/s about the decision to join the cluster. This notification takes place using CSMA MAC protocol.

On receiving all the messages from interested nodes, the cluster-head nodes generate a TDMA schedule and announce it to all the nodes within the cluster. In the steady state phase, cluster heads are aware of the schedule of each node transmitting the data during the allocated time slot. The sensor nodes start transmitting data to the cluster-heads. The cluster-head node receives all the data and aggregate the data by performing data fusion algorithms. The resulting information is then sent to the sink node. There exists an uncertainty regarding the strength of this protocol [59]. It is proposed that during the set up phase a set of nodes, "p", nominate themselves as cluster heads respectively. But the idea of uniformly distributing these cluster heads over the entire sensor network cloud is missing. The absence of uniform cluster heads in the sensor network can create the disparity in the rate of energy spending and in some cases may not even complete the communication from source to the sink node. Furthermore, the hypothesis of dynamic clustering can increase the burden of overhead. Secondly, LEACH protocol assumes that all the sensor nodes, irrespective of whether it is a cluster or not, consumes the same amount of energy.

Table 2.1 compares SPIN, LEACH and the Directed diffusion [59]. These three routing schemes are designed to so that collected data is disseminated efficiently in wireless sensor networks. However, due to in-network processing, directed diffusion shows a promising approach for energy efficient routing.

Table 2.1 Comparison among SPIN, LEACH and Directed Diffusion

	SPIN	LEACH	Directed Diffusion
Optimal Route	No	No	Yes
Network Lifetime	Good	Very Good	Good
Resource Awareness	Yes	Yes	Yes
Use of Meta-Data	Yes	No	Yes

2.3.2 Power-Efficient Gathering in Sensor Information Systems-PEGASIS

In PEGASIS [69], each sensor node forms a pattern so that each node will receive from and transmit to a close neighbor. Each node takes turn being the leader for transmission to the base station so that the average energy spent by each node per round is reduced. PEGASIS outdoes LEACH'S performance by (1) purging the overhead of dynamic cluster formation, (2) decreasing the distance non leader-nodes must transmit, (3) reducing the number of transmissions among all nodes, and (4) using only one transmission to the base station per round. Principal goals in the operation PEGASIS are (a) augment the lifetime of each sensor node by using collaborative techniques (b) reducing the bandwidth of communication by allowing the local coordination among neighboring sensor nodes. The performance evaluation in [69] shows that PEGASIS is able to enhance the sensor network lifetime twice as much as the network implementing LEACH protocol. In PEGASIS, this performance gain is attained through the exclusion of the overhead caused by dynamic cluster formation and through reducing the number of transmissions and reception by using data aggregation. Though PEGASIS outweighs the LEACH protocol, there still exists an uncertainty regarding the depth of this protocol [59]. There should be a dynamic topology adjustment in PEGASIS for the nodes to know energy status of its neighbors for routing its data. Secondly, PEGASIS presume that all the sensor nodes maintain a database with the location of all other nodes in the network, which increases the overhead. PEGASIS also assumes the communication of each sensor node with the sink directly, without the multihop routing.

2.3.3 Power Concerned Routing

Since a sensor network has limited bandwidth, it is necessary to minimize communication between sensor nodes. In terms of power consumption, operation of a wireless sensor node can be divided into three parts: sensing, processing, and transmission. Among those three operations, it is known that the most power consuming task is data transmission. Approximately, 80% of power consumed in each sensor node is used for data transmission. Energy-aware routing algorithms [51], [52], [53] discuss reducing the consumption of battery-power at the different nodes. Reference [70] explains energy management at the MAC layer using TDMA along with periodic listen and sleep to avoid energy wastage. The authors in [41] discuss about the narrow computing power of the sensor nodes and the limited bandwidth of the connecting nodes, which deter the communication of sensor nodes within the wireless sensor cloud. This section explains some power management techniques. They can be broadly classified as

1. Static power management, broadly applied at the (node) design time, aiming at different levels of system's hardware and software components and
2. Dynamic power management, applied at runtime. Dynamic power management takes into consideration the runtime events, to reduce power when the sensor nodes are idle or catering to trivial workloads.

2.3.3.1 Dynamic Voltage Scheduling (DVS)

Schemes such as auto shutdown and dynamic voltage scaling (DVS) have emerged as powerful methods for power-aware computing. In sensor networks, DVS plays an integral part in reducing the power consumed by a processor at each node during an active state. In a sensor node, the workload for a processor is not always constant; it varies over time [71]. Depending on the application involved and the processing speed, a node is either active or idle. This power optimization is realized by distributing workloads throughout the entire cycle of a processor. In other words, DVS minimizes the workload at a peak and spreads it during the idle times. Processor workload distribution can be accomplished by reducing processor frequency and voltage, which decreases the processing speed. One important point in designing a DVS system is that the processing speed has to be reduced without harming the efficiency of the entire network. Reducing only the frequency does not increase the processor power efficiency. By reducing frequency, the power consumption is decreased, but, the amount of task processed is also reduced. Because of the linear relationship between power consumption and task processing, the energy consumed by the task does not change. On the other hand, reducing the voltage applied to a processor by reducing the processing frequency, leads to a quadratic energy reduction [71]. Therefore, by changing the frequency and the voltage, the total power consumed per task can be reduced. One aspect of DVS is predicting future workloads. Since decisions to spread workload are based on the current and future workloads, the accuracy of future workload estimation can dramatically change the efficiency of DVS. Thus, it is crucial to develop a good algorithm, for predicting future workloads of nodes. Energy conservation is uniquely vital for embedded systems, such as obscured wireless sensors, which are deployed in applications where it is difficult to physically access sensors. Since the amount of power available to these systems is limited, it is considered a daunting challenge to minimize the energy consumption in order to broaden the life of the battery. In this section, we discuss the related power efficient routing schemes.

2.3.3.2 Minimum Total Transmission Power Routing (MTTPR)

MTTPR [50] protocol is an on-demand, reactive routing scheme which seeks an optimal path from a source to a destination node in mobile ad hoc networks only when such a path is needed. The objective of MTPR development was to design an algorithm for finding a minimum transmission power consumption path from a source to a destination in a power-constrained network. The basic idea is that if a shortest path between two nodes is employed to transmit a data packet, the power consumed by the transmission will be minimized, because radio transmission power is proportional to the distance. More specifically, the power consumed is directly proportional to “d^n“’ where “d “ is the distance between the two nodes and the value of “n” depends on “d”; namely n=2 for short distances and n=4 for long distance [72]. Since data packets in ad hoc networks are transmitted in a multihop manner, the total power required in transmitting between a source and a destination is the sum of the transmission power consumed by each hop between two nodes necessary for a packet to reach the destination node. Therefore, the total transmission power Pt, can be expressed as follows:

$$P_t = \sum_{i=0}^{D-1} P(n_i, n_{i+1})$$

where D is the total number of nodes in the route, and n_0 and n_D are the source and destination nodes respectively [50]. An optimal route is determined by minimizing the total transmission power P_t over all possible routes between a source and destination node. This can be achieved by applying a shortest path algorithm, such as Dijkstra algorithm. Because the value of "n" in "d^n" is determined by the distance between the two nodes, MTTPR protocol tends to select routes, which have more nodes, but with shorter distances for each hop.

By selecting a path between a source and destination node with many short-distance hops, the total transmission power efficiency will be optimal. However, another consideration of the MTTPR is propagation delay. Because of the MTTPR route selecting method and the proportionality of transmission power to the distance, more nodes are usually involved in delivering data packets. Since each node requires some processing time, each node contributes to the propagation delay. Therefore, the more nodes in the route, the longer the propagation delay. Further more, each node consumes power in processing data packets. To address this problem, the receiving power of a node was introduced in addition to the transmission power [50]. By considering both power consumption factors, propagation delay and the number of nodes included in an optimal path can be reduced. Other consideration of the MTTPR protocol is the energy state of each node. Once an optimal path is selected, it can be used to transmit data packets as long as the route remains connected. Since some nodes can consume all of their energy while other nodes consume very little, patches can get disconnected and the network become fragmented.

2.3.3.3 Min-Max Battery Cost Routing – MMBCR

This scheme [50] is also an on-demand reactive routing scheme. It selects an optimal data path based on the power remaining in each node. To measure how much a node is willing to transmit a data packet at any given time, "t", one proposed equation is

$$f_i^t = \frac{1}{C_i^t} \text{ ,}$$

where C_i^t is the battery capacity of node "i" at time "t". As the residual battery capacity decreases, a node is less willing to participate in transmitting data packets. This phenomenon is expressed by increasing the f-value. The battery cost of a route j, Rj, is defined as the maximum f value among nodes in the j-route.

$$R_j = \max f_i(C_i)$$

Hence, an optimal route in the MMBCR protocol is determined by finding a route having a minimum Rj value over the set A of all the possible routes j $\in$A between two nodes.

$$R_{(optimal)} = \min\{R_j \mid j \in A)\}$$

The MMBCR protocol is guaranteed to select a path, whose minimum power capacity node is a maximum. However, unlike the MTTPR protocol, MMBCR does not take into account the total transmission energy consumed by each data packet transmission. Therefore, the path selected by MMBCR is not necessarily the most energy efficient path.

2.3.3.4 Conditional Max-Min Battery Capacity Routing

CMMBCR protocol [50] is a routing scheme, which combines MTPR and MMBCR in an effort to maximize network power efficiency. CMMCR considers the best possible routing in terms of total transmission power and power consumption fairness over all routes in a network. In CMMBCR, the battery capacities of a node are divided into two states according to a threshold capacity value. There are three possible scenarios:

- all nodes have capacities above the threshold
- all nodes' have capacities below the threshold and
- Some capacities are above and some are below the threshold.

If the battery capacities of all nodes are above the threshold value, MTPR is used and CMMBCR selects a route with minimum total transmission power consumed per packet. Consequently, the power consumption of the whole network is minimized. On the other hand, if the battery capacities of all nodes are less than the threshold value, MMBCR is used, so that the lifetime of nodes with low capacity can be extended. In the third case, if there exists a route, between a source and a destination for which all nodes have capacities above the threshold value; the optimal route is selected by applying MTPR. If all possible routes from a source to a destination contain only nodes with capacities below the threshold value, a route is selected by applying MMBCR. One disadvantage of CMMBCR is that it does not allocate energy evenly throughout all nodes, as was expected [50]. Since the CMMBCR scheme is also a reactive routing scheme, a routing process is activated only when a route is needed for transmitting data packets. The power status of each node is not monitored continuously unlike proactive routing schemes which maintain routes periodically. Thus, after an optimal route is selected and as long as it is used for transmitting data packets, the power status of all nodes on the route is not monitored. This means that even if the power capacity of a node on a route is below the threshold level, it has to keep transmitting data packets as long as the route is active.

2.3.3.5 Modified Conditional Max-Min Battery Capacity Routing

In this scheme [50], two threshold values – selective-victim-search-zone (SVSZ) and forced-victim-search-zone (FVSZ) – are used in addition to the threshold value, γ, used by the conventional CMMBCR. The general idea of Modified-CMMBCR is as follows. The two constant values, SVSZ and FVSZ are applied to all nodes in a network, where SVSZ > FVSZ. On the other hand, γ is determined by a source node, so if a source applies a low γ value for one route, the route can be used despite having a low node capacity. A source node can change the threshold value depending on the data type transmitted. Also, each route can have a different γ value. Then, if the remaining power of node on a route becomes less than γ, a new route will be sought. Unless the remaining power of a node becomes less than both γ and SVSZ, all nodes continue transmitting data packets. In case a remaining power is less than SVSZ and greater than FVSZ, a source node receives a signal from a low power node to seek a new route, while the low power node continues to transmit data packets. Finally, if the remaining power of node is less than FVSZ, it sends a signal to a source node to seek a new route, and stops transmitting data packets. At this point, a node transmits data packets only when it is a source node. One advantage of this scheme over CMMBCR is that it reflects the power status of all nodes on a route during the data transmission state, so more power-aware routing can be achieved. In addition, since each source can determine the γ-value, a route can be selected according to the priority of data packets. For instance, if a source has a low γ value, more nodes participate in a selected route than if the source has a high γ-value, so a better route, which will be a shorter and have smaller propagation delay, can be selected. On the contrary, one disadvantage of this modified CMMBCR is the overhead created by transmitting control signals. When the remaining power of a node reaches SVSZ, FVSZ, or γ, it has to transmit a control signal to its source node to select another route. This will cause more control signals throughout the network as compared to MTPR, MMBCR, and CMBCR.

2.3.4 Threshold Sensitive Energy Efficient Sensor Network Protocol

TEEN [74] is a hierarchical protocol using data centric mechanism to route the data to sink. It is designed to be responsive to abrupt variations in the sensed physical attributes such as temperature, pressure etc. In TEEN, physical phenomenon is sensed constantly, but the actual data transmission is done sparingly. Clusters are formed and cluster heads are chosen. The cluster head sends two thresholds to the fellow nodes within the cluster. These two threshold values are (a) Hard Threshold, which is the threshold value of the sensed attribute and (b) Soft Threshold, is a small modification in the value of the sensed attribute that triggers the sensor node to switch on its transmitter and transmits to the respective cluster head. This way the sensor nodes transmit only when the sensed attribute is in the span of interest. The soft threshold lessens the number of transmissions that would have otherwise taken place without any change in the sensed attributes. To organize an effective data transmission, values for both soft and hard threshold can be attuned. TEEN protocol is a trade-offs between energy efficiency and data accuracy. This protocol is appropriate for time critical sensing applications, such as forest fires, sudden temperature increase etc. Downside of TEEN protocol is that if the updated threshold values do not reach the cluster head, the nodes cannot communicate and the information can never reach to the end user.

APTEEN [75], Adaptive Threshold sensitive Energy Efficient sensor Network protocol is an augmentation to TEEN. It is intended to acquire periodic data collections and is more receptive to time-critical events depending on the type of the application. In APTEEN, the cluster-heads broadcasts hard and soft thresholds, and the transmission schedules to all the nodes within the cluster. The node senses the environment constantly, and the sensor nodes which sense the physical data value beyond the hard threshold are allowed to transmit. The sensor node will transmit data only when the values of that attribute changes by an amount equal to or greater than the soft threshold [59]. In APTEEN, the count time is the maximum time period between two successive reports sent by the sensor node. If the sensor node does not send data beyond the count time, TDMA schedule is used and each node in the cluster is assigned a transmission slot. The performance evaluation of TEEN [74] and APTEEN [75] shows that both of them outperform LEACH. Performance of APTEEN in terms of network lifetime and energy dissipation is better than LEACH. On the negative feature of this scheme, is the added complexity required to execute the threshold functions and the count time. The problem of overhead on forming clusters at multiple levels and the method of implementing threshold-based functions still remains in APTEEN.

2.3.5 Self-Organizing Protocol

Self Organizing Protocol [76] is a protocol with self-organizing capabilities and taxonomy based on the sensor applications. The self organizing protocol architecture support heterogeneous sensors that can either be mobile or stationary. A subset of the sensor nodes probe the environment and forward the data to a selected set of nodes that acts as routers. Router nodes are stationary and form the backbone for communication.

The sink nodes are the robust nodes in terms of energy. The collected data is forwarded through the routers to sink node. The routing architecture is hierarchical where set of nodes are formed and merged when needed. In order to maintain fault tolerance, Local Markov Loops (LML) algorithm, which executes a random walk on spanning trees of a graph, is used in broadcasting.

The algorithm for self organizing the router nodes and creating the routing tables consists of four phases. (a) Discovery phase, where each sensor node, discover its respective neighbor/s. (b) Association phase, in this phase based on the grouping of each sensor node a hierarchy is formed. Each sensor node is allocated an address depending upon its position in the hierarchy. A routing table of size O (log N) is created for each sensor node. Broadcast trees that cover all the nodes are created. (c) Maintenance phase, in this phase each node notifies the neighbors about its respective energy level and routing table. Updating of routing tables and the energy levels of sensor nodes are made in the maintenance phase. Local markov loops are used to maintain the broadcast trees. (d) Self-reorganization phase, where the group reorganization is performed in case of node failures. There is a small cost of maintaining the routing tables in this protocol and performance evaluation shows that the energy consumed for broadcasting a message using self organization protocol is less than that consumed in SPIN [41] due to the broadcast trees utilized in the algorithm.

Table 2.2: Hierarchical vs. Flat topologies routing [59]

Flat Routing	Hierarchical routing
Contention-based scheduling	Reservation-based scheduling
Collision overhead present	Collisions avoided
Variable duty cycle by controlling sleep time of nodes	Reduced duty cycle due to periodic sleeping
Node on multi-hop path aggregates incoming data from neighbors	Data aggregation by cluster head
Routing is complex but optimal	Simple but non-optimal routing
Links formed on the fly without synchronization	Requires global and local synchronization
Routes formed only in regions that have data for transmission	Overhead of cluster formation throughout the network
Latency in waking up intermediate nodes and setting up the multipath	Lower latency as multiple hops network formed by cluster heads always available
Energy dissipation depends on traffic patterns	Energy dissipation is uniform
Energy dissipation adapts to traffic pattern	Energy dissipation cannot be controlled
Fairness not guaranteed	Fair channel allocation

This protocol, however, is not an on-demand protocol especially in the organization phase of algorithm, thereby causing an extra overhead. Secondly there is another drawback in forming hierarchy when there are many cuts in the network [59]. This will be expensive since network-cuts enhance the probability of employing reorganization phase. Table 2.2 compares the different aspects and issues of hierarchical routing and flat routing.

2.3 Location Based Routing Protocols

Wireless sensor networks are spatially deployed over a region depending on the application. There is no global addressing scheme for sensor networks like IP-addresses. In location based routing sensor nodes are addressed by means of their physical locations. The distance between neighboring nodes can be calculated on the basis of incoming signal strengths. Comparative coordinates of the neighboring nodes can be acquired by exchanging information between neighbors [40], [77], [78]. In location based scheme, some nodes go to sleep, in order to save the energy. The problem of designing sleep period schedules for each node in a localized manner was explained in [79]. If the location of the sensor nodes and the region to be sensed is known, a query can be diffused only to that specific region which will reduce the number of transmissions significantly. Initially a number of protocols from mobile ad hoc networks were employed on wireless sensor networks [81], [82], [83], [84], [85], [86]. These location-based protocols utilize the location information of ad-hoc nodes to achieve scalability in large-scale networks. However, many of these protocols are not applicable to sensor networks since they are not power aware. This Section discusses some relevant location aware routing protocols.

2.4.1 Geographic Adaptive Fidelity

GAF [81] is a power-aware location-based routing algorithm designed primarily for ad hoc networks, but can be applicable to wireless sensor networks too. GAF conserves energy by switching off unnecessary sensor nodes in the network without any effect on the level of routing fidelity. The sensor cloud is first divided into fixed zones and forms a virtual grid. Inside each zone, nodes poll resources with each other to play different roles. For example, one sensor node is elected by others to stay awake for a certain period of time and then they go to sleep. This node is responsible for monitoring and reporting data to the sink on behalf of the nodes in the zone [80]. Each sensor node uses its GPS-indicated position to associate itself with a spot in the virtual grid. Nodes related with the same point on the grid are considered equivalent in terms of the cost of packet routing. Such equivalence can be removed by keeping some nodes positioned in a particular grid area in sleeping state in order to save energy.

Figure 2.8 [81], an example of virtual grid in GAF is depicted. Node 1 can reach 2, 3 and 4 and nodes 2, 3, and 4 can reach 5. This shows that nodes 2, 3 and 4 are equivalent and two of them can sleep. In order to balance the load, each node change state from sleep to active mode. The three stages namely defined in GAF are (a) Discovery stage, this stage decide the neighbors within the grid, (b) active stage, which includes the active routing and (c) sleep stage, when the radio is turned off. The state transitions in GAF are depicted in Figure 2.9, redrawn from [81], [80].

In order to control the mobility, each sensor node in the grid estimates its respective leaving time from the grid and sends to its neighbor. In order to reliably route the data, the inactive or sleeping neighbors adjust their sleeping time accordingly. Before the departure time of the active node expires, the inactive node wake up and becomes active. GAF is implemented both for non-mobile sensor nodes (GAF-basic) and for mobile sensor nodes (GAF-mobility adaptation).

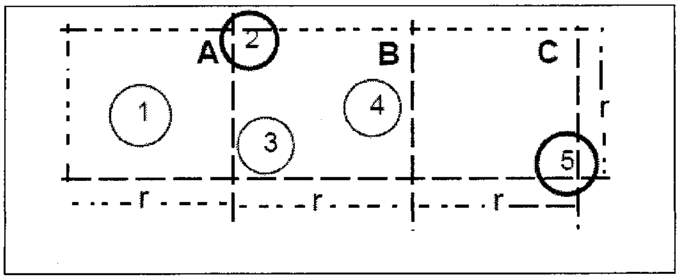


Figure 2.8 Example of virtual grid in GAF

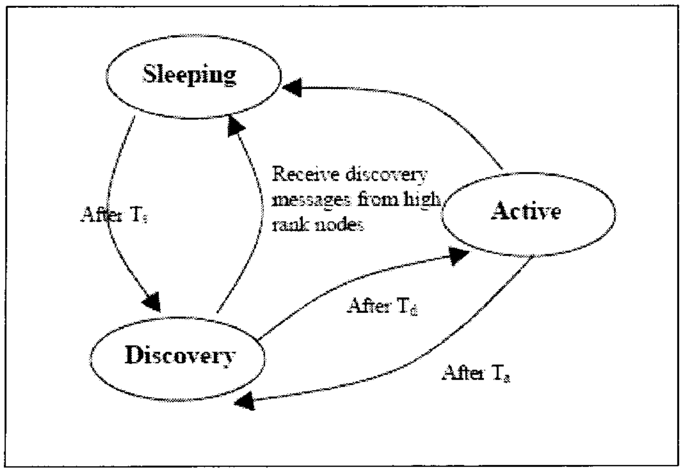


Figure 2.9 State transitions in GAF

GAF [81] assume that sensor nodes can identify their locations using GPS cards, which is inconceivable with the current technology. Performance evaluation of GAF shows that it performs reasonably well as a normal ad hoc routing protocol in terms of latency and packet loss. Besides it increases the lifetime of the network by saving energy. GAF may also be considered as a hierarchical protocol, where the clusters are based on geographic location [59]. For each particular grid area, a master node acts as the leader to transmit the data to subsequent nodes. It is worth mentioning that in GAF, the leader node does not do any data aggregation like other hierarchical protocols discussed earlier in this article.

2.4.2 Minimum Energy Communication Network

Minimum Energy Communication Network (MECN) [87] sets up and maintains a minimum energy network for wireless networks by utilizing low power GPS. The initial assumption of this protocol is for a mobile network, but it is applicable to the wireless sensor networks. MECN identifies a relay region for each sensor node. This relay region is a collection of the sensor nodes in a surrounding area, through which transmission is more energy efficient than the direct transmission. Figure 2.10 shows the relay region for a node pair (i, r) [87], [80]. The enclosure of a node “i” is then formed by taking the union of all the relay regions that node “i” can reach. The key proposal of MECN is to find a sub-network, which will have less number of nodes and require less power for transmission between any two particular source and destination pair. A localized search for each sensor node is performed considering its respective relay regions. This way the minimum power paths are found without taking into account all the nodes in the network. MECN protocol is self-reconfiguring and can dynamically adjust to node’s failure or the deployment of new nodes.

SMECN [88], Small minimum energy communication network is a realistic modification over the MECN. SMECN assumes possible obstacles between any pair of nodes unlike the assumption in MECN that each node can transmit to every other node. The sub-network constructed by SMECN for minimum energy relaying is smaller in terms of number of edges. As a result, the number of hops for transmissions will decrease. Simulation results show that SMECN uses less energy than MECN and maintenance cost of the links is less. However, finding a sub-network with smaller number of edges introduces more overhead in the algorithm.

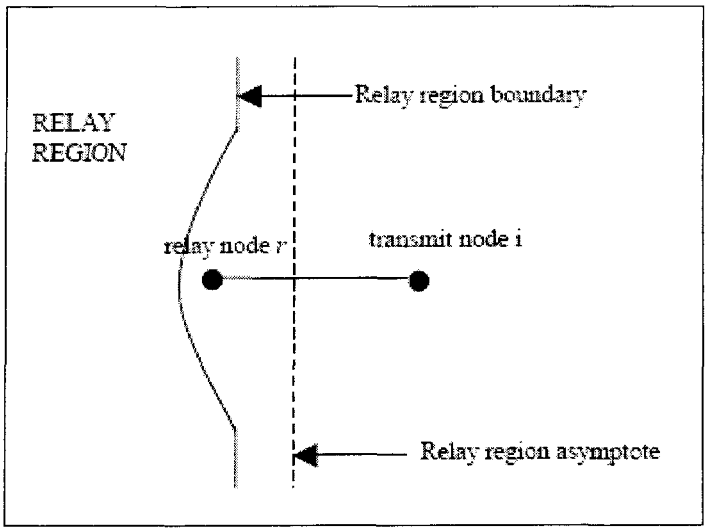


Figure 2.10 Relay region of transmit-relay node pair (i, r)

2.4.3 Geographic and Energy Aware Routing (GEAR)

GEAR, [89] discusses the utilization of geographic information while disseminating queries to suitable regions since data queries often include geographic attributes. GEAR uses an energy aware and geographically-informed neighbor select heuristics to route a data packet towards the sink region. This routing algorithm limits the number of interests in directed diffusion by only taking into account, a certain region instead of sending the interests to the entire network. Each sensor node in GEAR maintains an estimated cost and a learned cost of reaching the destination. The estimated cost is calculated by the combination of distance to the sink and the residual energy. The learned cost is the supplement of the estimated cost that accounts for routing around holes in the network. Formation of hole occurs when a node does not have any neighbor in the target region other than itself. In absence of the holes, the estimated cost is equal to the learned cost. The learned cost is transmitted one hop back every time a data packet reaches the sink so that route setup for next packet can be adjusted. There are two phases in the algorithm

- Forwarding packets towards the target region: On receiving the packet, sensor node make sure that there is at least one neighbor, which is closer to the target region than itself. If there is one neighbor, it is selected. If it's more than one, the nearest neighbor to the target region is selected as the next hop. If no neighbor is found, it is accounted as a hole. Then one of the neighbors is chosen to forward the packet based on the learning cost function.
- Forwarding the packets within the region: After the packet has reached the region, it can be disseminated by restricting flooding or recursive geographic forwarding. In high-density networks, recursive geographic flooding is more energy efficient than restricted flooding.

2.4 Multipath Routing Protocols

There is another subset of routing in wireless sensor networks called multipath routing. In this routing scheme, instead of one single path, the notion of multi paths available from source to sink is established. This scheme definitely enhances the network performance by improving the fault tolerance of the network. At the same time the multipath routing increases the overhead of maintaining the alternate paths. The fault tolerance of a protocol is measured by the possibility of existence of an alternate path between the source and the sink node, provided the initial path fails. Design of an optimal routing protocol must take into consideration the trade off between the network performance and fault tolerance capability. In this section we will discuss some multipath routing schemes. Disjoint paths are created to endure node/link failures. Disjoint multipaths are very resilient but at the same time they are extremely energy inefficient. [90], [91] propose an optimal algorithm to estimate the link disjoint paths in the network. An optimal algorithm for finding both node-disjoint and link-disjoint paths in the wireless network is proposed in [92]. Fault-tolerant clustering algorithm is proposed in [93] to detect the failure and to recover sensors from the failed gateway node. Fault-tolerant relay sensor node placement problem is studied in [94]. A polynomial time approximation algorithm is proposed to solve this problem. Using a set of sub-optimal paths occasionally to increase the lifetime of the network is proposed in [95]. These sub-optimal paths are chosen by the probability of amount of energy consumption in each path. The path with the most residual energy may be expensive for routing the data in the network. A balance must be made to minimize the total power consumed and the residual energy of the network. The authors in [96] proposed an algorithm in which the residual energy of the route is relaxed to enable it to select a more energy efficient path

The reliable multipath routing was proposed in [97]. This scheme is useful for delivering data in harsh physical environments. The tradeoff between the amount of data traffic and the reliability of the sensor network is analyzed. The data traffic increases by initializing several paths from the source to destination. This tradeoff is analyzed by using a redundancy function that is dependent on (a) degree of multipath and (b) failing probabilities of the available paths. Data packets are split into subpackets and then send to each sub packet through one of the available multipaths. The simulation proves that even if some of the sub packets were lost, the actual message can still be reconstructed. It has been concluded from this algorithm that for a given maximum node failure probability, using a higher multipath degree than a certain optimal value will rather augment the total probability of failure.

In Directed diffusion [64], when a path between a source and the sink fails, an alternative path should be identified. For this, Directed Diffusion basically reinitiates reinforcement by searching among other paths, which are sending data in lower rates. An extended version of directed diffusion [98], suggest initializing multiple paths in advance so that in case of a path failure, one of the alternative paths is chosen without probing for another one. There is an extra overhead involved in maintenance of these alternative paths. The alternate paths are kept alive by sending a low data rate message constantly

2.5 Negotiation Based Protocols

In negotiation based protocols, high level data descriptors or labels are incorporated within the sensor network. With the help of these data descriptors, sensor nodes negotiate with the neighboring nodes to eliminate redundant data transmissions. Exchange of communication between the sensor nodes depends on the resources available to each senor node within the network. SPIN [41] family of protocol is based on the continuous collaborative negotiation of sensor nodes. The SPIN protocols are designed to disseminate the data of one sensor to all other sensors assuming these sensors are potential base-stations. The key idea of negotiation based routing in WSNs is to hold back the superfluous information and avert redundant data from being sent to neighboring sensor node. This is accomplished by performing a series of negotiation messages before the real data transmission begins.

2.6 Coherent and Non Coherent Routing

In wireless sensor networks the processing of the data is required at the node level. The sensor nodes make a collaborative effort to process the data within the sensor network. The routing mechanism which initiates the data processing module is proposed in [99]. This mechanism is divided into two categories;

- Coherent Data Processing Based Routing: This category is an energy efficient mechanism where only the minimum processing is done by the sensor node. Time stamping, duplicate suppression etc are the tasks accomplished in minimum processing. After the minimum processing, the data is forwarded to the aggregators.
- Non Coherent Data processing based routing: In this category the, the sensor nodes locally process the actual data and then send to the other nodes for further processing. The nodes that perform further processing are called the aggregators. There are three phases of data processing in non-coherent routing. (a) Target detection, data collection, and preprocessing (b) Membership declaration, and (c) Central node election [59]. In target detection stage, an event is detected, its information collected and preprocessed. In the membership declaration phase, sensor node chooses to participate in a cooperative function and declare this intention to all neighbors. In the central node election stage, a central node is chosen to perform more refined information processing.

Additionally, single and multiple winner algorithms were proposed for non-coherent and coherent processing, respectively [99]. A single aggregator node is chosen for complex processing in the single winner algorithm (SWE). The selection of this node is established on the robustness of the sensor nodes in terms of energy and computational ability. By the end of the SWE process, a minimum-hop spanning tree completely covers the network. In multiple winner algorithm (MWE), when all the nodes send the data to the central aggregator node, this expends more energy. In this algorithm, limit the number of nodes that can send data to the central aggregator node. Each node maintains a record of up to "n" nodes, instead of only the best candidate node. This way each sensor node in the network has a set of minimum-energy paths to each source node (SN). Single winner algorithm is employed to find that node which yields the minimum energy consumption. This node can then operate as the central node for the coherent processing.

2.7 QOS Based Routing

Quality of Service enables the sensor network to provide better service to information flows. The performance of sensor network should be the balance between energy consumption and data quality. The network while delivering data to sink has to assure certain QoS metrics like latency, power, bandwidth etc. Sequential Assignment Routing (SAR) [99] takes into account the quality of service requirements in the sensor networks. It takes into account three factors (a) energy resources, (b) QoS on each path, and the (c) priority level of each data packet. SAR includes the multipath approach and localized path restoration. To create multiple paths from a source node, a tree is formed from the source node to the sink. The paths of the tree are formed in accordance to QoS metrics. At the end of this process, each sensor node will be part of multi-path tree. SAR algorithm takes into account the weighted QoS metric, which is the product of the (a) additive QoS metric and (b) weight coefficient associated with the priority level of the packet. Throughout the network lifetime, the objective of SAR algorithm is to minimize the average weighted QoS metric. A path re-computation is needed in case of node failure. SAR is a multipath routing scheme, which ensures fault-tolerance and easy recovery. But at the same time the protocol suffers from the overhead cost of maintaining the tables at each sensor node.

2.8 Open Issues in Sensor Network Routing

Sensor nodes are not assigned any global identifications like an IP address for the computers; instead, sensor nodes and the data are acknowledged through their respective contents, location and constraints. The data centric routing is generally followed in order to avoid the overhead of forming clusters. The naming schemes such as attribute-value pairs might not be adequate for complex queries and they are usually dependent on the application. Efficient standard naming scheme is one of the most appealing future research direction related to this category. Another interesting research issue regarding the formation of cluster heads is to optimize the latency and the energy consumption. According to [80], cluster formation and cluster-head communication are open issues for future research. The fusion among different clusters is also an interesting problem to explore. Protocols that employ the physical information and topological establishment of sensor nodes are classified as location-based. An optimized energy efficient solution to utilize the location information needs to be studied further. Quality of Service is another issue for the concentration of research. Real time applications such as signal processing, broadcasting video etc. demand an optimal balance between QoS requirements and energy efficiency. Another interesting issue for routing protocols is the consideration of node mobility. Most of the current protocols assume that the sensor nodes and the sink are stationary. However, there might be situations such as battle environments where the sink and possibly the sensors need to be mobile. In such cases, the frequent update of the position of the command node and the sensor nodes and the propagation of that information through the network may excessively drain the energy of nodes. New routing algorithms are needed in order to handle the overhead of mobility and topology changes in such energy constrained environment. Other possible future research for routing protocols includes the integration of sensor networks with wired networks (i.e. Internet). Most of the applications in security and environmental monitoring require the data collected from the sensor nodes to be transmitted to a server so that further analysis can be done. On the other hand, the requests from the user should be made to the sink through Internet. Since the routing requirements of each environment are different, further research is necessary for handling these kinds of situations.

2.9 Conclusions

Advances in sensor node architecture have made the large-scale deployment of sensor networks a reality. A variety of applications require sensor nodes to collect information over a continuous time period and forward to the sink directly or co-operating with other sensor nodes. The sensor networks, jointly with sensing devices, embedded processors, and communication components, uses an appropriate energy-efficient and fast routing strategy to deliver the data to the desired node. This chapter, besides setting the background for the proposed framework, also attempts to classify the key routing techniques used in sensor networks. Each routing technique is studied in terms of resource usage, efficiency, applicability and scalability and the most challenging research directions are outlined. Each of the routing schemes and algorithms has the common objective of trying to extend the lifetime of the sensor network. This chapter also focuses on the design tradeoffs between the energy consumption and fault tolerance in different routing scheme. There are some hybrid protocols that can be placed under more than one category. The summarize research results is shown in table 2.3 [59]. The Table compares different routing techniques according to many metrics.

Table 2.3: Categorization and Assessment of Routing protocols in Wireless Sensor Networks

	Classification	Mobility	Position Aware	Power Usage	Negotiation Based	Data Aggregation	Localization	QoS	State Complexity	Scalability	Multi path	Query Based
SPIN	Flat	Possible	No	Limited	Yes	Yes	No	No	Low	Limited	Yes	Yes
Directed Diffusion	Flat	Limited	No	Limited	Yes	Yes	Yes	No	Low	Limited	Yes	Yes
Rumor Routing	Flat	Very Limited	No	N/A	No	Yes	No	No	Low	Good	No	Yes
COUGAR	Flat	No	No	Limited	No	Yes	No	No	Low	Limited	No	Yes
ACQUIRE	Flat	Limited	No	N/A	No	Yes	No	No	Low	Limited	No	Yes
LEACH	Hierarchical	Fixed BS	No	Maximum	No	Yes	Yes	No	CH's	Good	No	No
TEEN & APTEEN	Hierarchical	Fixed BS	No	Maximum	No	Yes	Yes	No	CH's	Good	No	No
PEGASIS	Hierarchical	Fixed BS	No	Maximum	No	No	Yes	No	Low	Good	No	No
MECN & SMECN	Hierarchical	No	No	Maximum	No	No	No	No	Low	Low	No	No
SOP	Hierarchical	No	No	N/A	No	No	No	No	Low	Low	No	No
GAF	Location	Limited	No	Limited	No	No	No	No	Low	Good	No	No
GEAR	Location	Limited	No	Limited	No	No	No	No	Low	Limited	No	No
SAR	QoS	No	No	N/A	Yes	Yes	No	Yes	Moderate	Limited	No	Yes

CHAPTER 3

RELIABILITY IN WIRELESS SENSOR NETWORKS

Increasing computing and wireless communication capabilities will expand the role of the sensors from mere information dissemination to more demanding tasks as sensor fusion, classification and collaborative target tracking. Fault tolerance and reliability performs an exclusively vital role for embedded systems, such as obscured wireless sensors, which are deployed in some applications where it is difficult to access them physically. Due to their complex architecture and possible deployment in harsh environments, wireless sensor nodes and the entire network are exposed to a variety of malfunctioning. Ideally, a reliable output is obtained at the sink node with the help of a set of processors which assimilates information in a collaborative manner. The sensor architecture, network topologies, different integration techniques and heuristics should be robust and fault tolerant even in unfriendly environments. In wireless sensor networks from the perspective of fault-tolerance:

1. The quality of the output should not be affected adversely and despite of transient or random failure of nodes, the network must be capable to function.
2. There should be an appropriate integration of information in real-time, even when the sequential control at the nodes is not so perfect.
3. The protocols should dynamically adapt to changes in the network environment.

4. The network should be able to re-configure on loss of nodes, or failure of wireless links which is not unusual in a sensor network
5. Sensor network topology changes frequently, due to node failures, introduction of additional nodes, variations in sensor location etc. The network should be able to identify the most important types of faults, techniques for their discovery, and to ensure efficiency of fault resiliency methods.
6. Research on information security is still is in its infancy. Much of the work is directly taken from the wireless ad hoc networks. A methodological analysis needs to be performed in terms of primary threats and possible attacks to the proper functioning of sensor networks.

In this chapter, besides describing a comprehensive overview of fault tolerance techniques in general, we also explore the reliability issues in multifusion sensor networks. We present Markov models for the reliability using different types of sensors and spares that replace sensors when failed. We compare these models in terms of reliability, cost and MTTF (Mean-Time-To-Failure). We conclude by outlining the potential future research directions along several dimensions.

3.1 Fault Tolerance Techniques in Wireless Sensor Networks

Sensor nodes can have various reasons for failures e.g. physical damage, environmental interference, deficiency of power and an adversary's malicious attacks. Without fault tolerance, these failures can have a crucial effect on the functioning of the sensor network. Fault tolerance is the ability to sustain overall sensor network functionalities without any interruption despite sensor node failures [100], [101], [123].

In the current research literature there are several fault tolerance techniques for sensor networks, including fault models, self organization algorithms, design of reliability model, and availability of nodes to generate a robust performance.

The faulty sensor nodes can send incorrect information and can even be inconsistent when sending information to different sensors. This faulty behavior is referred as Byzantine [102]. In the presence of such faults, agreement needs to be performed for all the non faulty sensors to arrive at the same final decision. Numerous studies have been conducted on agreement and it is proven that to reach agreement in the presence of "m" Byzantine faulty sensors, the network must contain "$N \geq 3m + 1$" sensors. Value-fusion and Decision fusion [103] are the two distinctive approaches studied for achieving fault-tolerance in collaborative target detection algorithms. These approaches guarantee that when exchanging values, all the non faulty sensors obtain the same set of values and all the values sent by non faulty sensors are part of this set. Inconsistent values sent by faulty sensors are replaced by a majority vote or a default value.

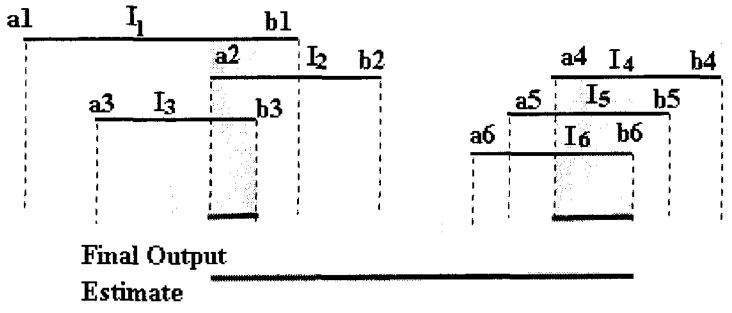
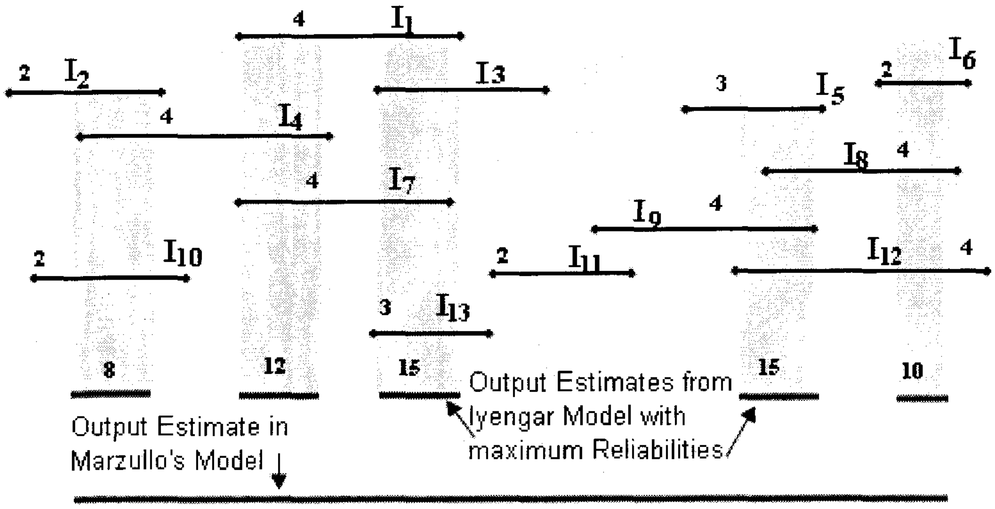


Figure 3.1: Marzullo Model

In Marzullo model [104], a processor receives inputs from several sensors whose outputs are connected intervals as shown in figure 3.1. A fault tolerant algorithm takes these discrete intervals as inputs and provides the output as a connected interval representing the sensor values. If there are "n" sensors, each providing certain output and measuring a certain physical value, the integration of these overlapped intervals will hold the correct or actual physical value in its interval. The wider this interval, the lesser is the accuracy of the processor output. If we assume that "f" sensors are faulty from a total of "n" sensors, it follows that at least "n-f" sensors are correct. Marzullo considers all possible non-empty (n-f) intersections of the "n" sensors and a sensor that doesn't belong to any of the (n-f) intersections is considered faulty. A correct sensor will overlap with at least (n-f-1) other correct sensors. The smallest connected interval containing all the (n-f) intersections is considered to be the output of the processor and we can conclusively say that this interval contains the actual physical value.



Comparison of output estimates from Marzullo's method and Iyengar model.The Shaded strip shows overlapping regions of three or more interval intersections, where n=13, f=10 and n-f =3

Figure 3.2 Iyengar Model

Table 3.1 Popularities of the intervals that make up the output interval

Interval	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}	I_{13}
Popularity	4	2	4	4	3	2	4	4	4	2	2	4	3

Iyengar's Model [105], [106] builds on the Marzullo's model but reduces the output interval estimate considerably. It explains that a sensor may wildly fail if there is no correlation between the actual physical value being measured and the interval estimate of the faulty sensor. Tame faults are those where the interval estimate lies significantly close to the correct value, even though the interval might not contain the actual physical value. It considers the case where the number of integrated sensors is large and most of the faults are tame and proved that the number of overlapping intervals is relatively large, as tamely-faulty sensors tend to overlap with the correct sensor estimates. Weights are assigned to each sensor overlap interval, based on the possibility of its containing the actual physical value. The maximum weight is set to that interval which is having the maximum probability of containing the physical value. The maximum weighted interval is taken to be the output estimate. According to [106], the reliability of the output estimate is a computation of the clustering of sensors around the maximum weighted interval and is addition of the popularities of the intervals that make up the output interval, (Table 3.1). Steps for algorithm [105], [106] are as follows:

1) Take all the (n-f) intersections to yield separate intersection intervals.
2) For each interval, count the number of intervals intersecting it having non empty intersections

3) Add these values to obtain the sum of the intervals involved with the formation of the weighted interval
4) Choose the maximum weight and call it "r". This choice is based on the estimation of the number of wildly faulty sensors and tame sensors
5) The higher the weight, the smaller the connected interval.
6) Assign I* as the integrated output estimate Now the integrated output estimate is much smaller than the entire overlapping estimate

With a very large number of sensors and by taking tamely faulty sensors into consideration, Iyengar's model reduces the output interval widths considerably compared to Marzullo's output interval estimate. Figure 3.2 shows the comparison of failure models where the width of the output interval estimate reduces significantly.

3.2. Fault Tolerance and Multisensor Fusion

Wireless Sensor Network (WSN) is transforming into a multi-service medium leading to the convergence of voice, video and data communications. Each type of service has a particular constraint and it has to be satisfied for the communication to be effective. For example a voice or video data is delay sensitive and has to be transmitted within a certain delay. So the service for each type of data needs to be met. Traditionally the current infrastructure only provided the best effort service, where the traffic is processed as quickly as possible, but there is no guarantee to the timeliness and assurance of actual delivery. This type of single service can no longer meet the need of the present day constraints. In [107], an interesting research regarding the fault tolerance aspects of a sensor network assumes that the nodes are either active or inactive with Bernoulli model.

In case that one or more sensors fail, other sensors of a different type can substitute their work, such that the fault goes undetected. This is called multimodal sensor fusion, and an interesting research of multimodal sensor fusion was done in [108]. The multimodal sensor fusion intrigues scientists in other disciplines, for example a still incompletely solved question is how we identify and deal with three dimensional objects while the eye retina works with only two dimensional patterns of light.

Given a network of multitype sensors, we study the aspects of fault tolerance of a multimodal sensor network [122]. We consider different models on achieving fault tolerance. The assumption of one failure at a time is not a strong assumption; two failures that happen at the same time can be consider consecutive, because we assume independent events. Another assumption made is that the failure of the components is independent of one another. There are cases of fault dependent events: the temperature raises suddenly, power fluctuations, etc, but we assume that any two faults are independent. As a result, any two events are disjoint in terms of probabilities.

Definition: The reliability function of a component at time t, $R(t)$ is a conditional probability that the component is operational at time t given that it was operational at time t_0. The unreliability of a system is $Q(t) = 1 - R(t)$. For any system, these conditions are generally true:

- Initially the system is functional at $t=0$: $R(0)=1$, $Q(0)=0$.
- Eventually the system will fail at $t=T$, $R(T)=0$, $Q(T)=1$.

The *reliability block diagram* (RBD) shows the dependence of the system reliability versus the reliability of each component. The Markov model for reliability of a system is based on two concepts: the possible states of the system, and the transitions between states. The failed state is annotated as F. The *reliability of the system* is defined to be as the probability of the system to be in any of those states except F; it is the probability of being in any state other than F (which is the sum of the probabilities of each state), or 1 – probability of the system to be in the F state. To measure the average time that each system operates before failing we consider the *Mean-Time-To-Failure (MTTF).*

Definition: MTTF is the expected value of the failure time

$$MTTF = \int_0^\infty t\frac{dQ(t)}{dt}dt = \int_0^\infty R(t)dt$$

$$MTTF = \int_0^\infty t\frac{dQ(t)}{dt}dt = \int_0^\infty R(t)dt$$

Definition: The *failure rate "λ"* is defined as the number of failures per time unit and is expressed as

$$\lambda = \frac{\frac{dR(t)}{dt}}{R(t)}$$

The spares can replace faulty components. We consider in our models *hot* or *stand-by* spares, which means that they replace immediately the failed sensor (there is no gap in time between the moment the sensor has failed and the moment the spare replaces it.) When a spare substitutes a module, then it has the same failure rate as the module. We study different models. We start with a model in which a spare can replace only one type of sensor, so there are different types of spares for different type of sensors, and we consider the case of two types. We continue with spares that can replace any type, and here we consider two-type and three-type pooled spares. To achieve a better reliability for the system, one solution is to improve the quality of the spares; another one is to increase the number of spares.

3.2.1 Modeling Single-Type Spares

Let A and B be two different types of sensors and two spares SA and SB that can replace only own type sensors (SA can replace only A, SB only B) (Figure 3.3).

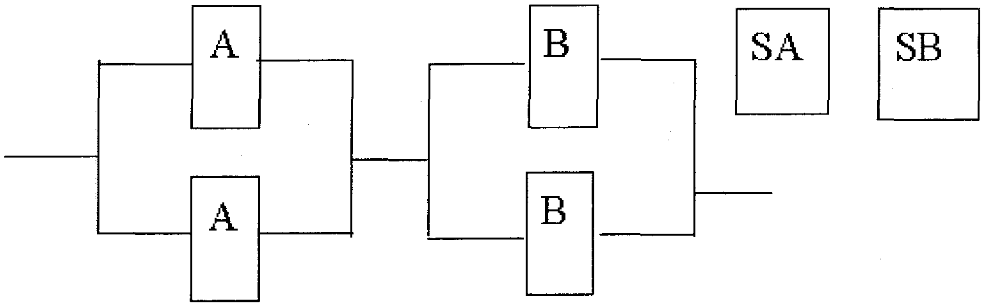


Figure 3.3 RBD diagram for two-single type spares

Given the failure rate for each component (λA,λB, λSA, and λSB), the Markov model for this example is drawn in Figure 3.4. If we consider only one spare or no spare, we obtain only portions of the Markov model drawn in Figure 3.4. If all components have the same failure rate λ (λA = λB = λSA = λSB = λ) then the reliability function is

$$R(t) = 9e^{-2\lambda t} - 18e^{-3\lambda t} + 15e^{-4\lambda t} - 6e^{-5\lambda t} + e^{-6\lambda t}$$ and

$$MTTF_{two-\sin gle-type} = \frac{73}{60\lambda} = 1.217$$

If we consider only one single-type spare, then the reliability function of the system becomes

$$R(t) = 6e^{-2\lambda t} - 9e^{-3\lambda t} + 5e^{-4\lambda t}$$ and

$$MTTF_{one-\sin gle-type} = \frac{21}{20\lambda} = 1.05$$

If we have no spare, then the reliability function becomes

$$R(t) = 4e^{-2\lambda t} - 4e^{-3\lambda t} + e^{-4\lambda t} \text{ and } MTTF_{no-spare} = \frac{11}{12\lambda} = 0.917 .$$

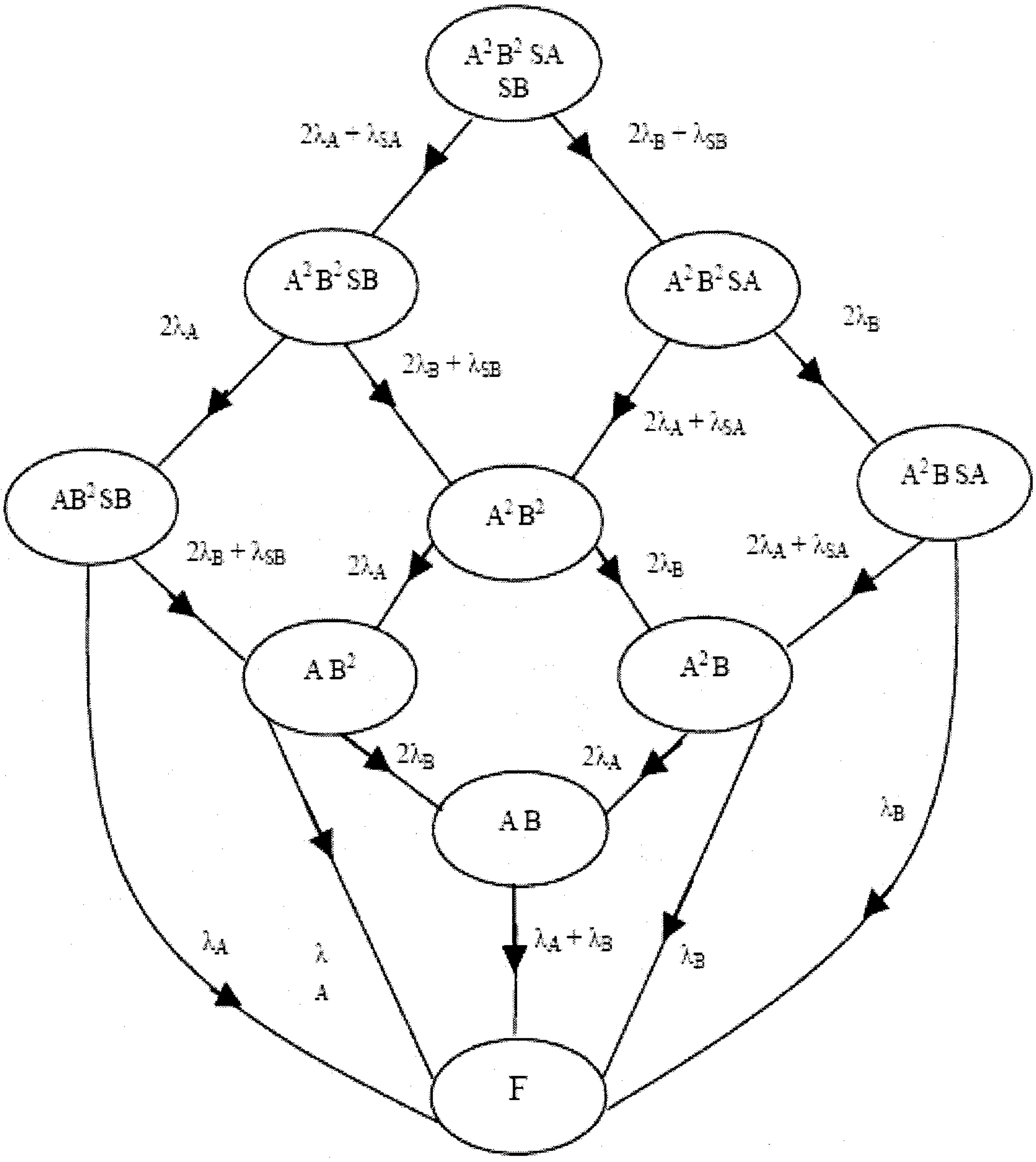


Figure 3.4 Markov model for two-single type spares

3.2.2 Modeling Pooled Spares

For modeling pooled spares, consider the case in which we have pooled spares that can replace any type of sensors.

3.2.2.1 Two-Type

Let A and B be two different types of sensors, and two spares of type AB that can replace any of the failed sensors, including themselves (see Figure 3.3 for the same RBD). Given the failure rate for each component (λ_A, λ_B, λ_{AB}) the Markov model is drawn in Figure 3.5, where S means AB.

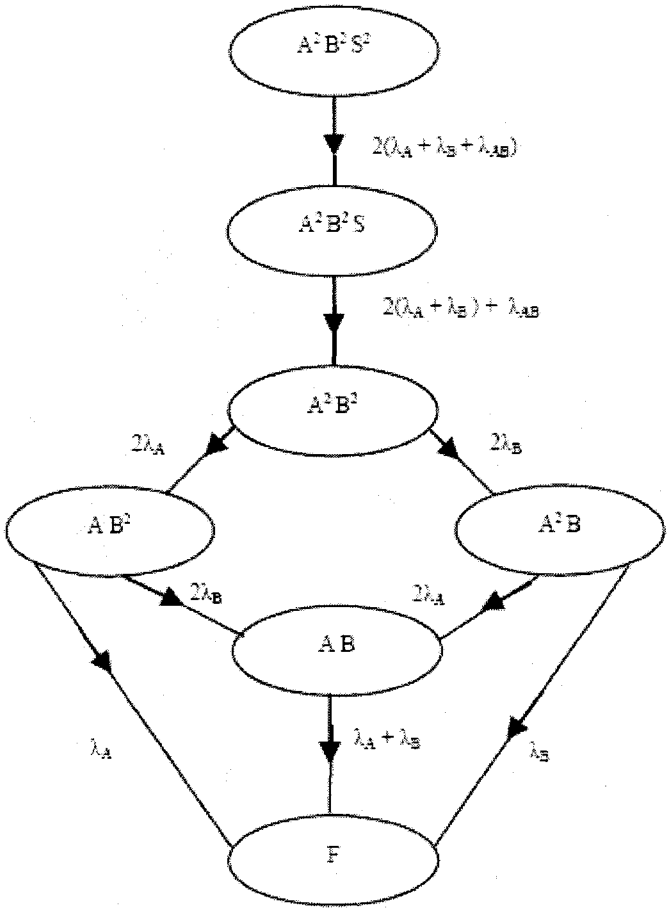


Figure 3.5 Markov model for two-pooled type spares

Assuming identical failure rates, $\lambda_A = \lambda_B = \lambda_{AB} = \lambda$, the reliability function of the system is $R(t) = 10e^{-2\lambda t} - 20e^{-3\lambda t} + 15e^{-4\lambda t} - 4e^{-5\lambda t}$ and

$$MTTF_{two-pooled-type} = \frac{77}{60\lambda} = 1.283 .$$

If we consider only one pooled-type spare, then the reliability function becomes

$$R(t) = \frac{20}{3}e^{-2\lambda t} - 10e^{-3\lambda t} + 5e^{-4\lambda t} - \frac{2}{3}e^{-5\lambda t} \text{ and}$$

$$MTTF_{two-pooled-type} = \frac{67}{60\lambda} = 1.117 .$$

If we consider no spare, then

$$R(t) = 4e^{-2\lambda t} - 4e^{-3\lambda t} + e^{-4\lambda t} \text{ and}$$

$$MTTF_{no-spare} = \frac{11}{12\lambda} = 0.917 .$$

3.2.2.2 Three-Type

Let A, B, and C be three different types of sensors, with the following RBD, and the spare of type ABC can replace any of the failed sensors, including themselves (Figure 3.6).

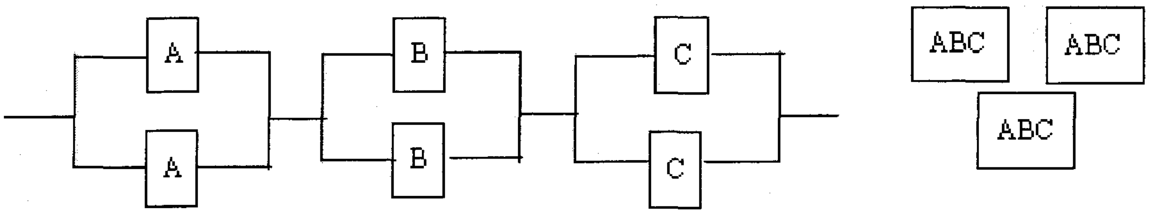


Figure 3.6 RBD diagram for three-pooled type spares

Given the failure rate for each component, λ_A, λ_B, λ_C, and λ_{ABC}, the Markov model for this example is drawn in Figure 3.7, where S means ABC.

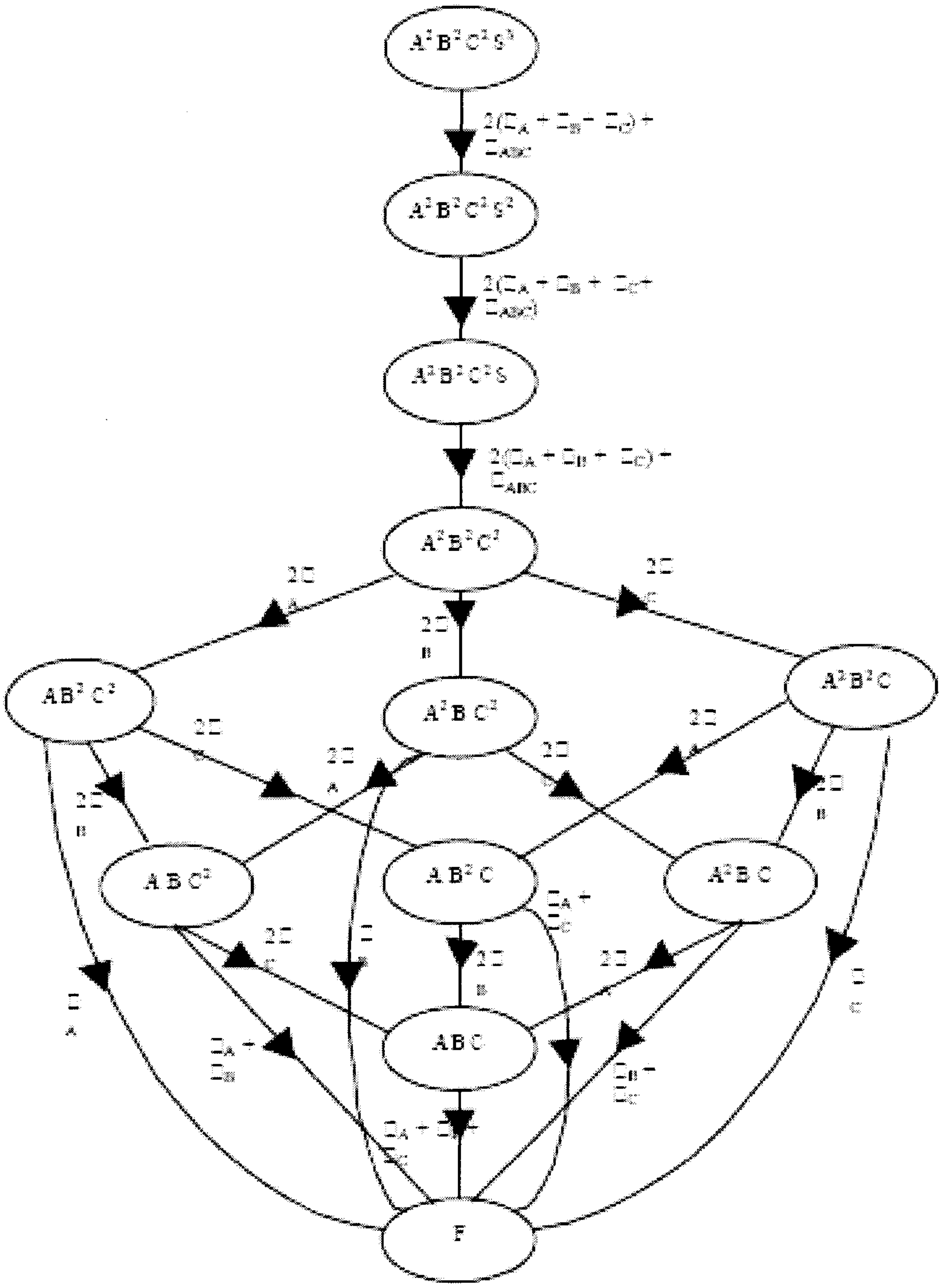


Figure 3.7 Markov model for three-pooled type spares

If we consider that all units including the spares have the same failure rate λ ($\lambda_A = \lambda_B = \lambda_C = \lambda_{ABC} = \lambda$) then

$$R(t) = \frac{168}{5}e^{-3\lambda t} - \frac{504}{5}e^{-4\lambda t} + 126e^{-5\lambda t} - 84e^{-6\lambda t} + 36e^{-7\lambda t} - \frac{63}{5}e^{-8\lambda t} + \frac{14}{5}e^{-9\lambda t}$$

and $MTTF_{three-pooled-type} = \dfrac{2719}{2520\lambda}$

If we consider only two-pooled spares, then the reliability function becomes

$$R(t) = \frac{112}{5}e^{-3\lambda t} - 56e^{-4\lambda t} + 56e^{-5\lambda t} - 28e^{-6\lambda t} + 8e^{-7\lambda t} - \frac{7}{5}e^{-8\lambda t}$$

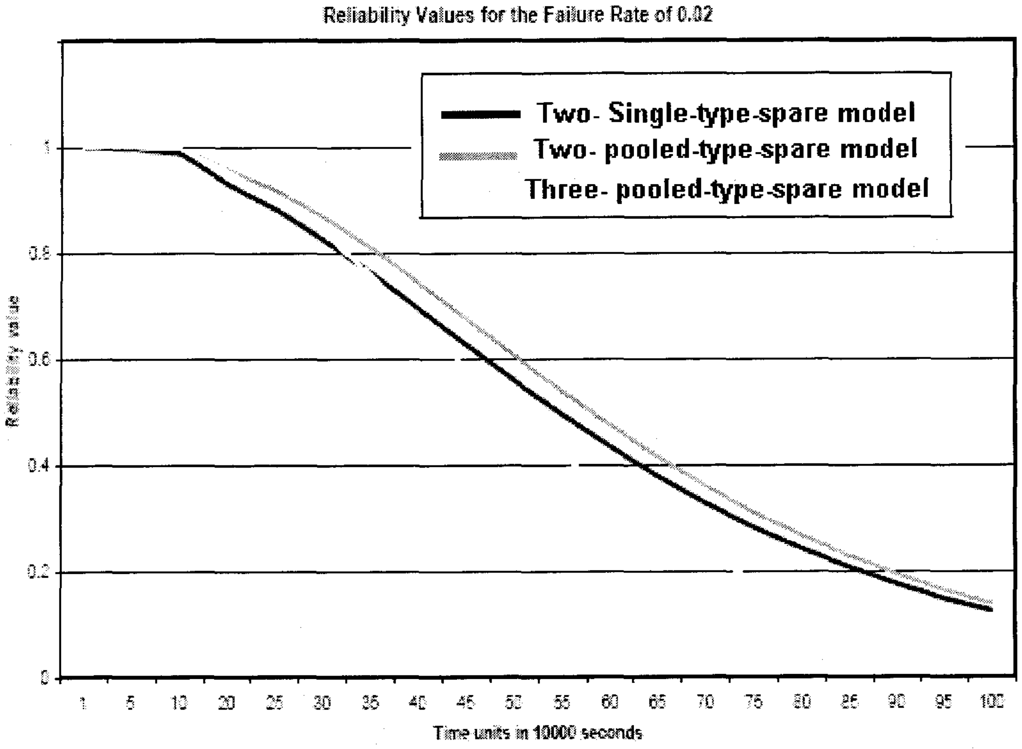
and $MTTF_{three-pooled-type} = \dfrac{271}{280\lambda}$.

3.2.3 Reliability versus Cost

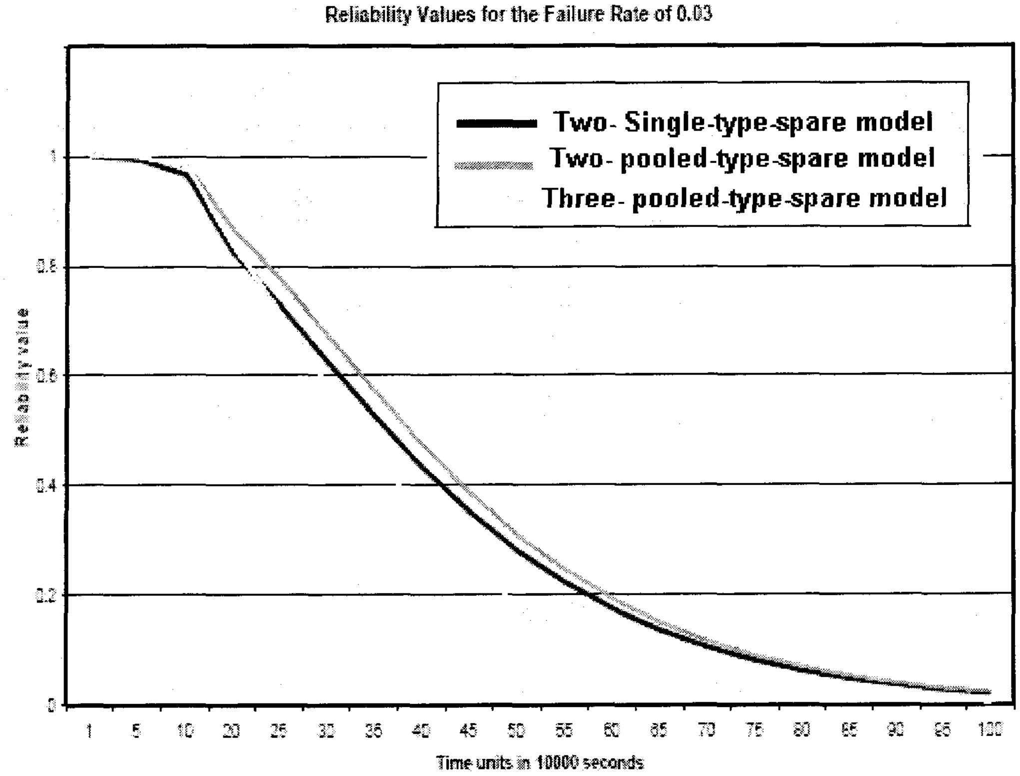
Consider the three models: two-single-type spares, two-pooled-type spares and three-pooled-type spares. In Figure 3.8 (a-d) are presented different reliability values, taking particular values for λ: 0.02, 0.03, 0.05 and 0.10 as the number of failures per 10000 seconds. Comparing these models in terms of MTTF, the third model has the lowest value, followed by the first model and the second model is the best, independent of the value of λ:

$$\mathrm{MTTF}_{\text{three-pooled-type}} = \frac{2719}{2520\lambda} < \mathrm{MTTF}_{\text{two-single-type}} = \frac{3066}{2520\lambda} < \mathrm{MTTF}_{\text{two-pooled-type}} = \frac{3234}{2520\lambda}$$

The cost of a non-redundant system is C; the added cost of a simple spare is c_1 and the added cost of a pooled spare is c_2. If a spare physically replaces a failed sensor then the cost of the system increases from C to C+c_1. If a spare virtually replaces a failed sensor, then the cost of the system increases from C to C+c_2.



Reliability Values for the Failure Rate of 0.02



Reliability Values for the Failure Rate of 0.03

Figure 3.8 (a) (b) Reliability values for λ = 0.02, 0.03 failures per 10000 seconds

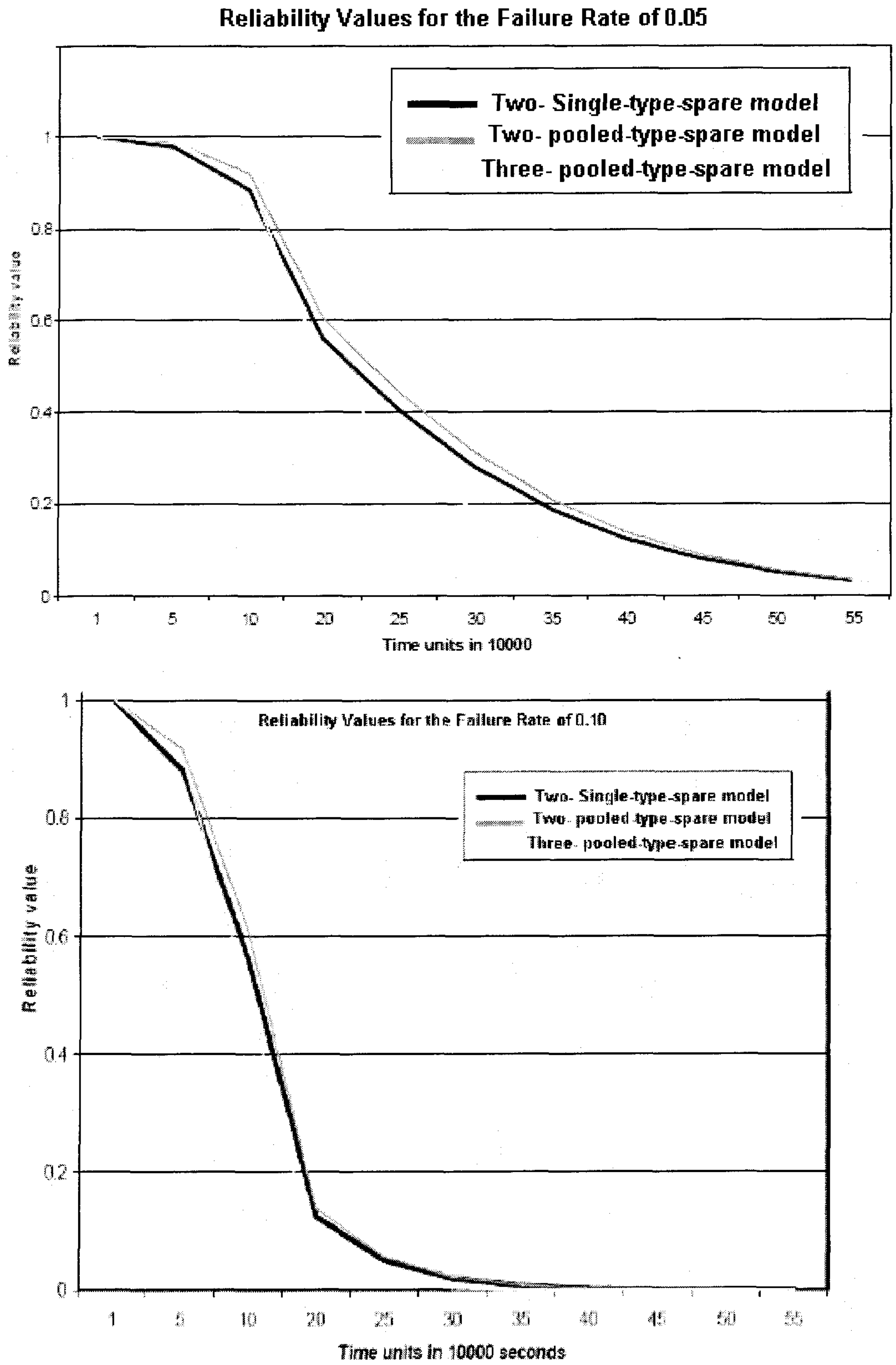


Figure 3.8 (c), (d) Reliability values for λ = 0.05, 0.10 failures per 10000 seconds

In Figure 3.9 an example is shown, where *C=4* for a two-type sensor system with no redundancy (no spares) with the cost of each component of 0.5. The single-type spare costs the same as one component, $c_1 = 0.5$. The pooled-type spare costs more than one component but less than two components, $c_2 = 0.75$.

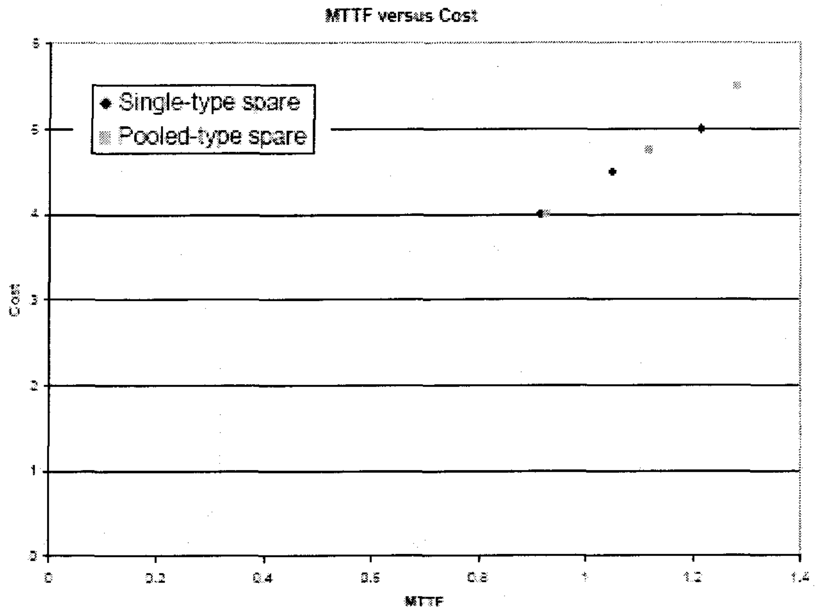


Figure 3.9 MTTF versus Cost for C=4, c_1=0.5 and c_2=0.75

3.2.4 Multifusion Sensor Networks

Consider a set *N* of *n* objects ($|N| = n$), a set *M* of *m* sensors ($|M| = m$) of *k* types: m_i sensors of type *i*, $\sum_{i=1}^{k} m_i = m$. The output of each sensor is binary $output_i(j) \in \{0,1\}, \forall i \in M, j \in N$. Shortly, consider *i(j)* to be *output* $_i$*(j)*.

Definition: Given two objects *a* and *b*, if $i(a) \neq i(b)$ then we can say that from the point of view of sensor *i*, the objects *a* and *b* are *distinguishable.* The assumption of the problem is that no two objects have the same properties, which means that for any two objects, there will be always a sensor which will differentiate them (the outputs of the sensor for those objects will be binary bit-wise).

Definition: If $M = \{ s_1, s_2, .. s_m\}$ is a set of sensors, then the *binary coding* of an object a is the ordered set of bits representing the output of each sensor s_i with regard to a :

$\text{coding}_M(a) = s_1(a)\, s_2(a) \ldots s_m(a)$.

Observation 1: By definition, two objects a and b is individualized by the set of sensors M is their binary encodings are different: $coding_M(a) \neq coding_M(b)$.

Observation 2: The maximum no. of sensors required to individualize n objects is $n-1$.

Observation 3: The minimum number of sensors required to individualize n objects is $\lceil \log n \rceil$

Proof: Given $L = \lceil \log n \rceil$, we can have $2^{\lceil \log n \rceil}$ different binary coding of length L. Based on observation 1, this means that we can have $2^{\lceil \log n \rceil}$ individualized objects. Because $2^{\lceil \log n \rceil} \geq 2^{\log n} = n$, L is a correct value. We prove by contradiction that taking less than L sensors, we cannot individualize all the n objects. Take $L_1 = \lceil \log n \rceil - 1$ and M_1 the set of L_1 sensors. The set of all binary coding of length L_1 has 2^{L_1} elements and $\log n + 1 > \lceil \log n \rceil \geq \log n \Rightarrow \log n > \lceil \log n \rceil - 1 \geq \log n - 1 \Rightarrow 2^{\log n} > 2^{\lceil \log n \rceil - 1} \Rightarrow n > 2^{L_1}$.

So there is only 2^{L_1} different binary coding but n objects, which mean that at least two binary coding of the objects in N are the same. This implies that with L_1 sensors we cannot individualize each object. As we see from the above observations, there is a parallel between *dimension redundancy* and *error correcting codes*. Given n bits of data, k bits of information and $(n-k)$ redundant bits, out of 2^n total number of binary strings, 2^k code words can be generated. Without redundant information, you are closing the room for error detection and error correction. So more redundant bits you have, better the chances of getting error correction.

3.3. Conclusions

This Chapter explored the reliability issues in multimodal fusion sensor networks. We presented the system reliability for the case of two types of sensors and three types of sensors. The system reliability was calculated and suggestive values for different λ are given in both cases. We compared these models in terms of reliability, cost and MTTF (Mean-Time-To-Failure). Finally we emphasize the similarity between dimension redundancy and error correcting codes. Chapter 4, Energy Efficient Information Dissemination in Wireless Sensor Networks, offer a detailed discussion of the proposed information dissemination scheme RM-IDLF.

CHAPTER 4

ENERGY EFFICIENT INFORMATION DISSEMINATION IN WIRELESS SENSOR NETWORKS

In this chapter, we begin with an explanation of the Information dissemination in WSNs and the basic assumptions that steer the design of our protocol scheme. We then define the network topology and actual working of IDLF- Information Dissemination by Label Forwarding. We also focus on the available alternate paths based on the quality of response time each path can provide. This is followed by an illustration of the steps involved in the construction of multiple paths as part of the proposed algorithm, RM IDLF- Reliable Multi-path Information Dissemination by Label Forwarding. In the end, we discuss salient features of the proposed multiple path protocol.

4.1. Motivation of Current Research

Wireless Sensor nodes are arbitrarily dispersed over the area of interest and are capable of RF communication to administer the communication protocols and Information processing tasks. Energy efficient routing protocols help optimize the number of transmissions required to set up routing paths and economize the cost of transmitting data packets. One of the challenges in designing a routing protocol for wireless sensor networks is to the find the most reliable path from the source to destination node, i.e. this path should deliver the data packets without retransmitting or discovering a new path. Secondly, a routing protocol for wireless sensor network should be well aware of sensor limitations. It should also take into consideration, the unique aspects of various applications running over wireless sensor networks, such as monitoring applications or acquisition of the sensitive data etc. Thirdly, the routing protocol design should support minimum-hop, hierarchical network topology with relatively high data throughput, and a deterministic latency. Finally, the routing in sensor networks must not involve creation of large routing-tables. Protocols must avoid network congestion.

In this section, an informal description of the shortcomings of the existing algorithms is illustrated. The criteria throughput, delay, complexity of routing algorithm, ease of implementation and number of request accepted are being used to evaluate the routing algorithms. In WSNs, Information dissemination protocols characterize methods for sensor nodes to transmit and receive queries and sense data in wireless sensor networks efficiently. There has also been interest in minimizing the transmission of redundant data in the network. In baseline protocols, such as flooding, the sensor nodes retransmits the data it receives to all its neighbors and broadcast within the entire sensor cloud. However, it results in data implosion with the destination getting multiple data packets from multiple paths. Due to indiscriminate transmission of data, sensor nodes in this scheme expend limited transmission energy and bandwidth. Another problem associated with flooding is overlap. This situation occurs when multiple nodes observe the same sensor region, and generate overlapping data. The sensor node within its neighborhood receives multiple copies of the same packet containing the same information. Overlap, like implosion expends transmission power and bandwidth. Similarly SPIN [41] (Sensor Protocols for Information via Negotiation) is based on the idea that a sensor node handshakes with its neighbors and the decision to forward the data packet is made after the handshake. Nodes in SPIN use high-level data descriptors called meta-data. SPIN uses meta-data negotiation to determine if a node needs the data and thus eliminates redundant transmissions. However the SPIN suffers from the weakness [63] of transmitting all the data packets at the same Energy level and not using the distance to a neighbor to adjust the energy level. Besides a large overhead in broadcasting the data is a concern in SPIN, which cannot be overruled.

On the other hand, directed diffusion is more suitable for applications with point-to-point data transmission. In directed diffusion, a sink node broadcasts an interest to initiate data collection. The interest is disseminated throughout the network in a hop-by-hop manner. The query is initiated by the *sink* node and it broadcasts its interest message periodically to all of its neighbors. An interest cache is maintained by each node. When a node receives an interest, it stores the interest and also sets up a gradient toward the node, from which it received the interest. This process continues until gradients are setup from the sources back to the sink. If a respective node has the requested data, which matches the received interest, the node sends back the data packet to the sink in multiple paths according to the gradients. On receiving the data packet at the sink, the reinforcement of the optimal path is initiated by the sink. The criterion for the selecting the optimal path highly depends on the application. It may be the shortest path or minimum energy consuming path, whichever suits the application. The best path is reinforced by the sink sending a new interest to the path.

Due to absence of a centralized control, wireless sensor networks are considered to be unreliable systems, where failures should be expected occasionally. Some of the common factors which make the sensor network communication unreliable are constrained power consumption requirements, high channel bit error ratio, external interference, asymmetric channel, data jamming and the hacking of the sensor information. Additionally, Sensor networks are highly dynamic. Within the network the node topologies frequently change due to a high rate of node failure, changes of power modes, and node mobility. It is a daunting research challenge to provide a robust data delivery under such a situation. Acknowledging that flooding based solutions fall short to handle the highly dynamic sensor networks, we propose a *"label forwarding dissemination"* solution for robust and reliable data delivery. In this solution, we aim at providing not only a reliable communication scheme, but also a fast response and recovery from the failures with a much less control overhead.

The two major contributions of the performed work, which are explained in this chapter are (1) Designing a scheme for a collaborative flow of information packet/s from source to sink. This scheme is further studied in different scenarios: (a) Fault-free single path (b) Fault-free multi-path (c) Single path with faulty nodes and (d) Multi-path with faulty nodes. For the ease of understanding we named this framework as IDLF-Information Dissemination by Label forwarding. IDLF is designed for point-to-point data transmission, and routing scheme is initiated by source nodes. IDLF is a reactive and on-demand routing scheme, which seeks a routing path only when it is needed. Every time a sensor node detects an event, a new data path is constructed. An extension to this scheme RM-IDLF- Reliable Information Dissemination by Label forwarding is proposed. RM-

IDLF also incorporates point to point data transmission where the source initiates the routing scheme and disseminates the information toward the sink (destination) node. Prior to transmission of actual data packet/s, a label path is formed, which is established by the source node issuing small label information to its neighbors locally. These labels are in turn disseminated in the network. By using small size labels, RM-IDLF avoids generation of unnecessary network traffic and transmission of duplicate packets to nodes. Another point of interest in this framework is the study of trade-offs between the achieved routing reliability using multiple disjoint path routing and extra energy consumption due to the use of additional path/s. Also, the effect of the failed nodes on the network performance is evaluated within the sensor system. It should be noted at this point that for RM-IDLF we used an alternate disjoint path. This alternate path scheme (RM-IDLF) may have a higher path cost in terms of energy consumption, but is more reliable in terms of data packet delivery to sink than the single path scheme (IDLF). In the latter scheme, the protocol establishes multiple (alternate) disjoint path/s from source to destination with negligible control overhead to balance load due to heavy data traffic among intermediate nodes from source to destination.

The second contribution of this work is the design and implementation of energy efficient schemes for uniform energy dissipation and service differentiation in a wireless sensor network. Maximizing the overall sensor network lifetime is considered as one of the vital objectives while designing a sensor network. Hardware and software design should direct towards reducing the energy consumption in sensor nodes. We propose a discrete energy efficient scheme, which is incorporated in the system in conjunction with IDLF and RM–IDLF. Setting up a battery threshold ensures that data packets will not be dropped after the sensor node's battery level falls below the threshold value. Minimum transmission around the sink prevents fast energy dissipation of the neighboring nodes to the sink. Finally, directional forwarding is applied to RM-IDLF. In directional forwarding, the sensor nodes narrow the range of broadcasting data packets by restricting communication only to the nodes lying in the direction towards the sink node.

4.2 Sensor Network Topology

In a multi-hop sensor network, a large number of potential paths exist between a source and a sink. The sensor nodes are initialized and arranged in a grid of the physical coordinates of the source-sink pair to construct a square boundary, with the sink constituting a fixed location within the grid. The location of the sink may or may not be fixed in actual scenario, but in this work we will assume the location of the sink to be always fixed. Practically, wireless sensor nodes are arbitrarily dispersed over the area of interest to administer the information processing and gathering task. These nodes are not arranged in a physical grid. Nonetheless, this assumption is necessary to understand the proposed routing algorithm and evaluate the overhead associated with it. The size of the physical grid is 10 x 10 unit sensor nodes. Figure 4.1, indicates the sensor network topology where the sink node, marked black is at (0, 0). The sink node only collects sensed data from other sensor nodes, but does not sense the event. Also, the sink is not resource constrained. It is equipped with enough memory space, battery power, and processing speed that the power consumed by the sink can be excluded from the total power consumed by an entire sensor network during simulations. Each sensor node can directly communicate with other nodes (neighbor nodes), which is located within one unit distance from the node. Every node in the network knows its coordinates in the physical field. The label dissemination protocols IDLF and RM IDLF intend to disseminate data towards the sink using negotiations.

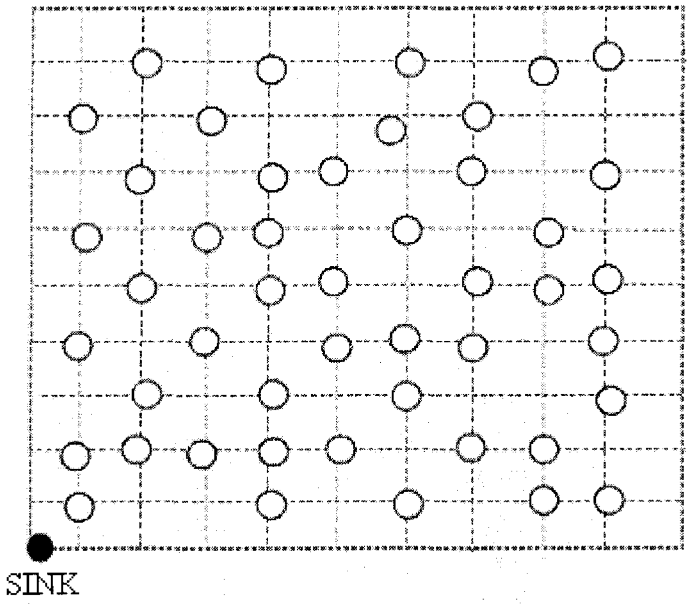


Figure 4.1 Topology of Sensor Network

4.3 Preliminaries

1) Sensor Node –A node in a sensor network is defined as a basic unit used to sense, process and direct the data packets to other sensor nodes.

2) Source node- a source node is the node which detects and records the event. It may have the capability to process the raw data. The primary objective of this listener node is to commune the data packets from a source node to a sink efficiently both in terms of energy and time. In our work, we assumed the presence of only one source node in the sensor field at a given time. After a data packet reaches the sink, a new source will be

selected randomly. However, depending on the application, the network could have multiple sources at the same time. In RM-IDLF, after the data packet reaches the sink, the simulation ends and the new source is not selected randomly. In the latter case, we are more interested in tracking the rate of successful data packets reaching sink. The evaluation of RM-IDLF is mainly dependant on the tradeoff of energy consumption and the reliability of the network. We would like to examine the effect of multiple sources on energy consumption and data dissemination.

3) Transmission Range- Authors in [111] describe that for "n" nodes randomly distributed in a disk, the network is asymptotically connected with probability one if the transmission range "r" of all nodes is selected. Range "r" is given by

$$r \geq \sqrt{\frac{\log n + \gamma(n)}{\pi n}}$$

Where $\gamma(n)$ a function that goes to infinity as "n" is becomes large. It is shown in [112] that the system-wide transport ability of the wireless network is optimized when every "hop" covers a very short distance. It is highly suggestive that the sensor nodes should therefore relay packets over very short distances to neighboring nodes, allowing them to transmit at low power. Practically, due to variations in implementation of physical device and in wireless propagation environment, the transmission ranges of different nodes are not exactly identical. In our work we assumed two transmission ranges "R1" and "R2"respectively. In range "R1", each sensor node can directly communicate with other nodes (neighbor nodes), which is located within one unit distance from the node. As shown in figure 4.2, the source node can directly communicate with (grid unit) one hop away node shown in gray region. In our simulation, based on the node configuration and communication range of nodes, each node can have a maximum of eight neighbors. On the other hand nodes with range "R2" can communicate with other nodes positioned two unit distances from it respectively. Figure 4.2 shows the range "R2" with black region. Any transmission using range "R2" is on the expense of energy, we assumed to be twice as much as using range "R2". The amount of energy consumed for exchanging information during a neighbor discovering stage is the same for any routing protocol for the same network topology. Therefore, we do not consider energy consumed during neighbor discovery in our energy analysis.

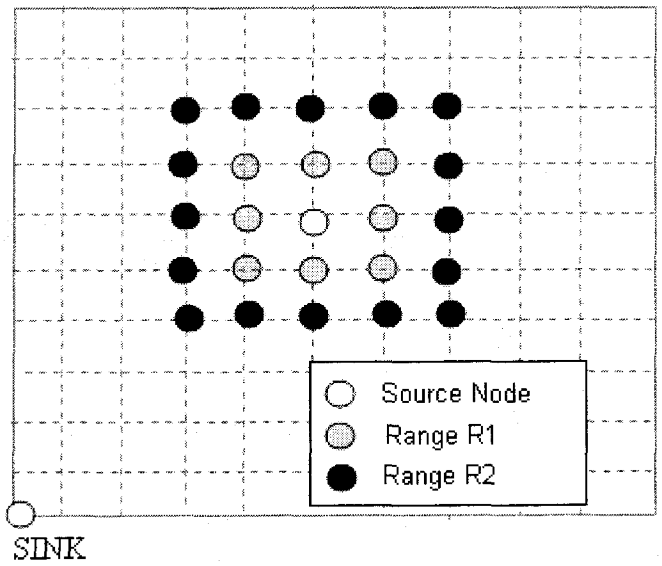


Figure 4.2 Transmission range R1 and R2

4) Node Initialization

(a) Grid (10 x 10) map

(b) Each node's parameter

- Location over X axis [x_loc]

- Location over Y axis [y_loc]
- Allocated Energy
- Neighbor (Range R1)
- Label Cache

5) Node Allocation

(a) Allocate Sink at (0, 0)

(b) Allocate all nodes by

- getting a random number
- check if the location is not assigned yet
- check if location is within the transmission range of the existing node

(c) If yes- node is allocated

(d) If not- node is not allocated, find another location

(Note: Default transmission range is R1 and a node can travel to 8 grinds around itself)

6) Neighbor Discovery

(a) Find nodes within the R1 range

(b) Check if the x- axis is in the simulation area (within the grid)

(c) Check if the y- axis is in the simulation area (within the grid)

(d) Find location to store the node information

7) Sensor Link Bandwidth- Bandwidth of a link is defined as the maximum traffic that the sensor node can accommodate at any given time. Wireless sensor networks are often used in continuous monitoring and control applications. Two resources—Link bandwidth and node energy—are scarce in many sensor networks, and need a careful administration .As sensor nodes get even smaller, and sensor networks grow larger in size,

the bandwidth consideration becomes increasingly more imperative. Sensor networks require a bandwidth allocation method, by which the nodes can decide how to assign network bandwidth to sensor streams. When a group of nodes in close proximity all detect an event of interest, this bandwidth assignment method has to handle traffic that demonstrates a high degree of spatial correlation. Depending upon the observed phenomenon, the bandwidth allocations should be varied [110]. For example, the goal of the temperature monitor sensor is to monitor changes in the temperature. If the sensor detects an unusual increase in temperature, it may imply a disastrous event like fire. In this case, it would be sensible to allocate almost all of the bandwidth to the main event stream.

8) Node Identification- Due to the relatively small sensor cloud size, the unique global Identification, such as IP addressing is not used in the sensor network. Node IDs have to be assigned before or after deployment [109].

Source ID	Event Location	Time Stamp	Sender ID

Figure 4.3: Example of Label Information

9) Label/Data Descriptor- In the current work we defined label as a short and fixed-length field containing the minimum information about the event so that it can be distinguished from other events. A label may be used as key in determining how to forward data packets. Generally labels are locally significant identifiers that are used to describe other data. For instance, if a wireless sensor network collects the data for a homogeneous event, the label should include the information about (a) the location where the event originated

(b) the time when the event generated (c) the identification of the source /listener node and (d) the sender node identification (Figure 4.3). Furthermore, the size of generated label should be significantly smaller than the size of data packets, because these labels are data descriptors and not the data packets themselves.

10) Neighboring Node- The neighboring nodes are those sensor nodes, which reside within the node's radio transmission range. In other words, neighboring nodes are the nodes located one transmission hop away from a node.

4.4 Information Dissemination by Label Forwarding

In the previous chapters, we have described related work in traditional wireless networks and sensor networks that has influenced the design of the proposed scheme. In this section, we introduce the label forwarding algorithm for Information dissemination. We then explain our motivation for applying this algorithm in the context to the wireless sensor networks and the associated challenges.

4.4.1 Assumptions

1. Only one sink node in the simulation field.
2. The location of sink node is fixed at (0, 0).
3. There is at least one sensor node within the normal transmission range of another sensor node
4. A source node will be assigned randomly.
5. There is only one source node in the sensor field at a time. After the data reaches to the sink, a new source will be assigned randomly.

6. When a source is neighbor to the sink, the source sends only the data to the sink without exchanging a label or request.
7. When transmitting a label, if the sending node of the label is neighbor of the sink, the sending node will transmit the label only to the sink.
8. However, based on various sensor simulation characteristic models shown in [41], [113], and [114], we assumed that the size of data packet is 31 times greater than the size of the label and request packet. Then if we assume that transmitting a label or request packet between two neighboring nodes takes one unit time, transmitting a data packet will take 31 unit times. Also, we assumed that transmitting information consumes 3 times more energy per unit time than receiving, so transmission and receiving takes up 3-unit energy per unit time and 1-unit energy per unit time respectively. Table 4.1 summarizes the network characteristics.

Table 4.1 Sensor Network Characteristics

Simulation Area	10 x 10 unit area
Number of Nodes	2-100
Number of Sinks	1
Radio Range	3 x 3 unit area
Data Size	31 unit
Request Size	1 unit
Label Size	1 unit

9. Propagation time between two nodes:
    a) Data Propagation Time – 31 unit time
    b) Request Propagation Time – 1 unit time

c) Label Propagation Time – 1 unit time

10. Energy consumption by a node:

    a) Transmission – 3 unit energy/unit time

    b) Receiving – 1 unit energy/unit time

4.4.2. Description of IDLF Algorithm

In the proposed algorithm the sensor nodes are deployed within the sensor cloud. The source node, which listens to an event, transmits information packets from sources to sink in low latency while conserving the power of each sensor node. The collaborative effort of the neighboring nodes reduces imbalance in the network utilization and increases the performance of the network. IDLF algorithm is divided into three stages:

- Label transmission stage,
- Request for transmission stage, and
- Data transmission stage

In each stage, a different type of information is exchanged among sensor nodes. To conserve the overall network energy consumption, wireless sensor networks are decentralized and distributed. There is no central hub or a server which controls the routing information in a routing table. Since, the sensor nodes are resource constrained, each node does not have enough memory space to store a routing table. Sensor nodes make use of the partial information within the sensor network. Each sensor node stores the information, such as relative location and ID, of neighboring nodes.

4.4.2.1 Label Transmission Stage

In this stage labels are flooded from the source node to sink node. The size of generated label is smaller than the size of the actual data packets, because these labels are data descriptors and not the data packets in themselves. The source node detects the physical phenomenon and listens to the event. On listening to the event, the source node forms a small information data descriptor, called a "label". A label as explained in the previous section is a short and fixed-length field containing the minimum information about the event so that it can be distinguished from other events. Since each sensor node has the local information about the node topology, the source node is unaware of the optimal path to route the query. The source broadcasts the label to all its neighboring nodes. On receiving the label, a receiving node examines its label cache, where all received labels are stored. If the node receives an entirely new label, the receiving node stores the label in the cache and retransmits the label to its neighbors. If the received label already exists in the label cache, the node disregards the received label.

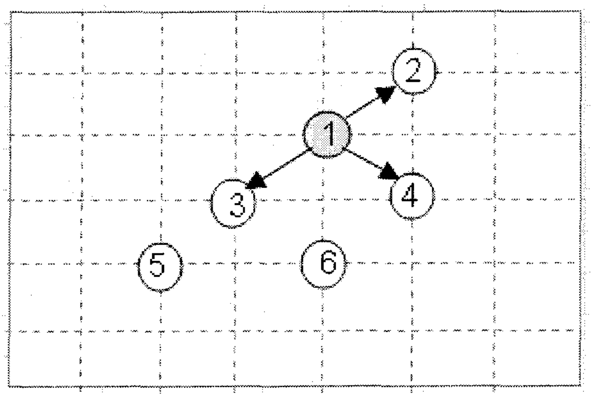


Figure 4.4: Label Propagation to immediate neighbors

In addition, if the label contains the information about a sending node, the receiving node can avoid retransmitting the same label to the sending node. This mechanism reduces unnecessary communication between nodes. The labels are flooded in the network until they reach the sink node or there are no more neighboring nodes left in the network.. A snapshot of label propagation is depicted in figure 4.4 below, where the sensor node "1" is assumed to be the source node, listening to the event. The label is initiated by node "1"and is transmitted to its neighboring nodes, node "2", node "3" and node "4" respectively. The receiving nodes store the label in their respective cache and at this point, the copy of the label information is stored in nodes "2", "3" and "4". These nodes retransmit the copy of the label to the appropriate neighbors. In label propagation, the implosion problem occurs, the same way it occurs in data flooding. When a label receiving node transmits the copy of label to its neighboring nodes, irrespective of whether the neighboring node already has the copy of the same.

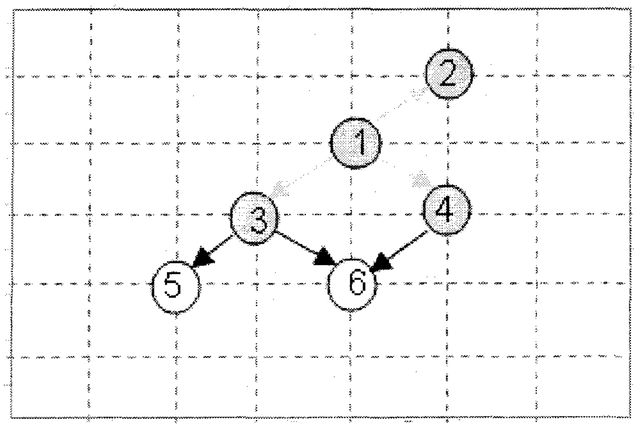


Figure 4.5: Label Propagation –Implosion

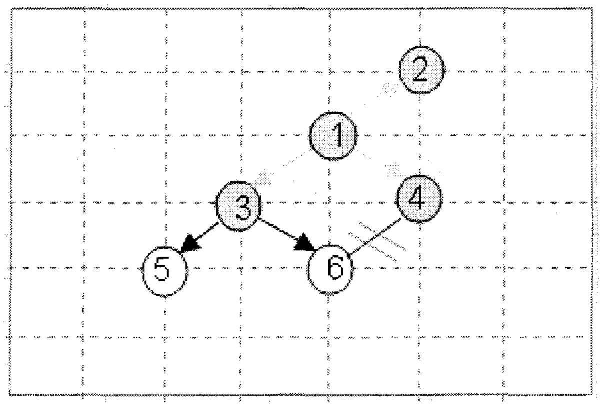


Figure 4.6: Label Propagation – Path Rejection to avoid implosion

Figure 4.5 shows the retransmission of labels, where node "3" and node "4" takes turn to transmit the label. Since node"3" receives the label from node "1", node "3" transmit the label to node "5" and node "6" respectively. Similarly, node"4" receives the label from node "1", node "4" transmit the label to node "6". Even though node "6" has already received the label from node "3", node "4" does not know the fact that node "6" already has the copy of the label. Node "6" being the neighbor to both node "3" and node"4", gets the same copy of the label information. Due to indiscriminate transmission of labels, sensor nodes in this scheme expend limited transmission energy and bandwidth. The transmission involves labels and not the actual data, therefore the energy and bandwidth expenditure are negligible. Nevertheless, to avoid the implosion, if the received label already exists in the label cache, the node disregards the received label. In this case, node "6" checks for the copy of the label cache to look for the label and disregard the path from node"4" to node"6",(figure 4.6). Also the label path from node"3" to node"1" or from node"4" to node"1" never exist because the receiving node avoids retransmitting the same label to the sending node. The labels are flooded in the network until they reach the sink node or there are no more neighboring nodes left in the network. Figure 4.7 & 4.8 shows the effective dissemination of the labels over the entire sensor network.

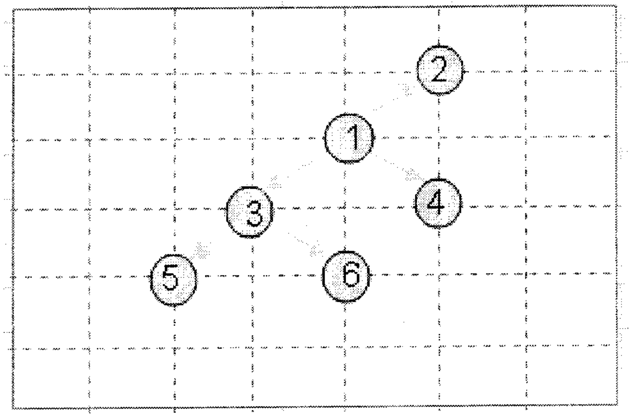


Figure 4.7: Label Propagation – Distribution over the network

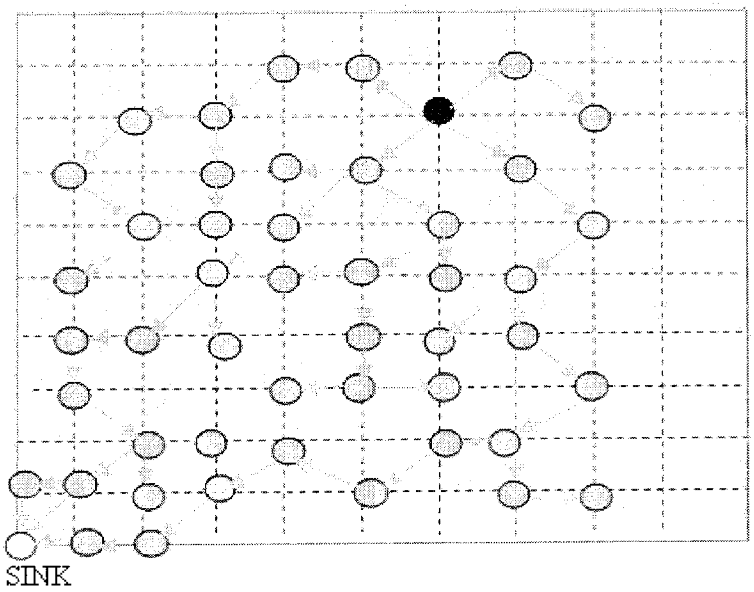


Figure 4.8: Promulgation of labels over the network

4.4.2.2 Request (Req) for Transmission Stage

At this point it should be noted that the labels are the data advertisement of the actual data. In view of the fact that the label contains only metadata, they are smaller, and inexpensive to send and receive than the actual information packet. The basic idea is to provide a route from source node to the sink using a three way handshake that permits the data packets to associate with Sink node securely. Once the labels are propagated in the network, the sink receives the label. The Sink wishes to receive the actual data from the source node, it responds back by transmitting a request packet (REQ) toward the source.

Similar to a label, this request packet is small in size compared to the actual data packet in order to minimize communication burden between sensor nodes. The request packet follows the trace, on which the label traversed from the source to the sink. An illustration of this scheme is presented below. In figure 4.9, the labels arrive at the sink by taking the path (S-1-2-3-4-5-6-7-8-SINK). It should be noted at this point that there may be many potential label paths from source to sink. However, we are interested in the first established path towards the sink, here (S-1-2-3-4-5-6-7-8-SINK). Figure 4.10 shows that the request packet (REQ) is transmitted back to the source node from the sink by taking the path (SINK-8-7-6-5-4-3-2-1-S).

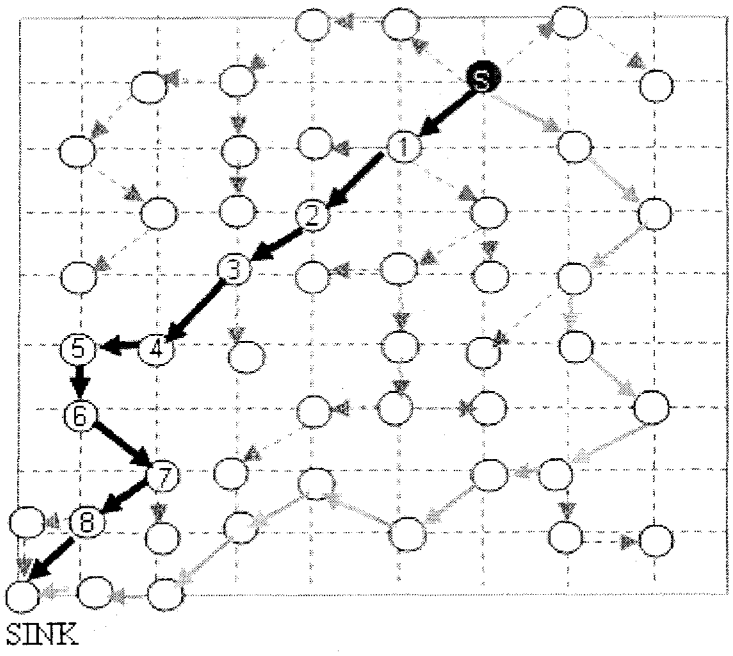


Figure 4.9.Label path/s from Source to Sink

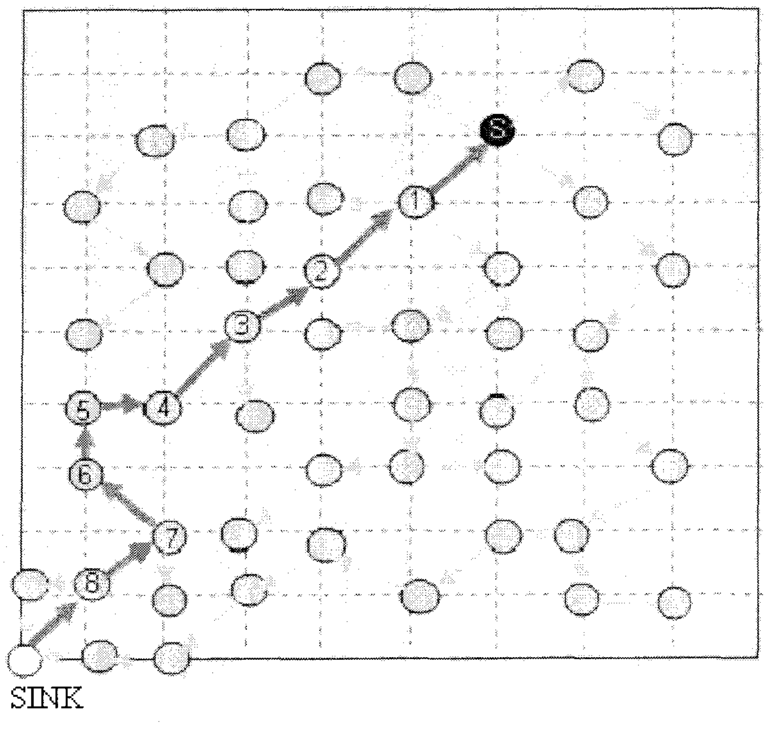


Figure 4.10 Request packet pursues the trace, on which the label traversed from the source to the sink.

4.4.2.3 Data Transmission Stage

On receiving the request packet (REQ) from the sink, the source node starts transmitting the actual data packet by following the same path one more time. Figure 4.11 below, shows that the data packets are delivered towards the sink through a label tunnel, which is an outcome of the three way handshake between the source and sink node respectively.

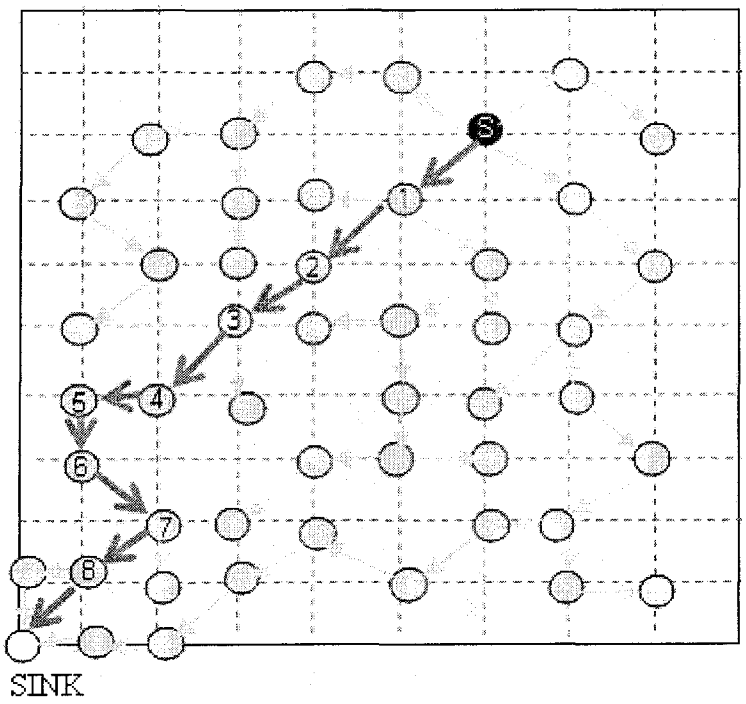


Figure 4.11 Data Transmission

If a node notices that its neighbor list has changed, it can spontaneously re- advertise the already established labels. The benefit of this routing scheme lies in its performance simplicity. Each node in the network has the knowledge of a small portion of the entire network topology. Each node stores minimum routing information to save the restricted bandwidth of sensor node, which as a result reduces the processing time for routing. Also, since an actual data packet is transmitted after a data path has been established; redundant data packet transmissions can be avoided. Thus IDLF algorithm can be executed over an entirely un-configured sensor network with a small, initialization cost to determine nearest neighbors.

4.5 IDLF- Directional Forwarding Model

In IDLF the nodes make local decisions based on label propagation. We use directional forwarding as a special case, in which the only prerequisite for the node is to know the direction of the fixed sink within the network relative to the source node. To disseminate a label to a sensor network, there are several possible methods. One choice is just broadcasting a received label to all the neighbor nodes without any restriction, which is merely applying a classic flooding. As explained in (section 4.4), this method is simple, robust, and effective if sensor nodes have no knowledge of sink's location (Figure 4.12).

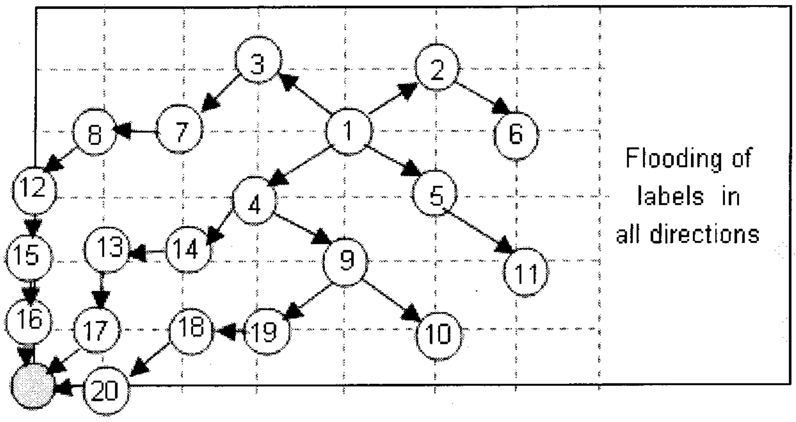


Figure 4.12 Label Propagation in all directions

Another possible choice is forwarding the labels by employing the sense of directionality within each sensor node. This method is based on the assumption that each node in the sensor network has the knowledge of the probable location of the sink node. This scheme can lessen the involvement of the neighboring nodes, to which a label has to be disseminated. The performance evaluation of this scheme, as explained in chapter 5 results in energy saving.

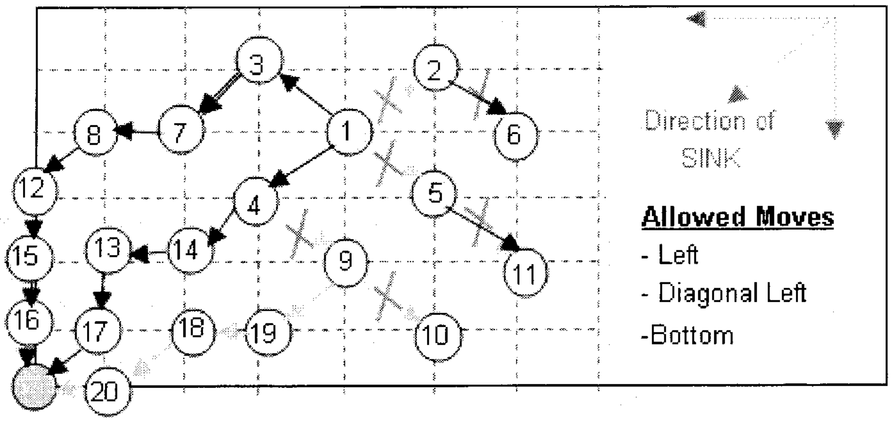


Figure 4.13 Restricting Propagation of Labels in all directions

Figure 4.13 illustrates an example of directional forwarding; here node “1” is a source node having its immediate neighbors as nodes “2”, “3”,”4“, and“5” respectively. On the physical grid, the location of sink is (0,0), which is south-west of node “1”. The label transmission of the source node and subsequent nodes can be restricted to only neighbors which has the physical location lying south (bottom), west (left), or south-west (diagonally left). In Figure 4.13, with respect to source node “1”, node “3” and “4”meet the criteria for receiving the label. By limiting the number of nodes information disseminated, energy consumed for exchanging information will be reduced.

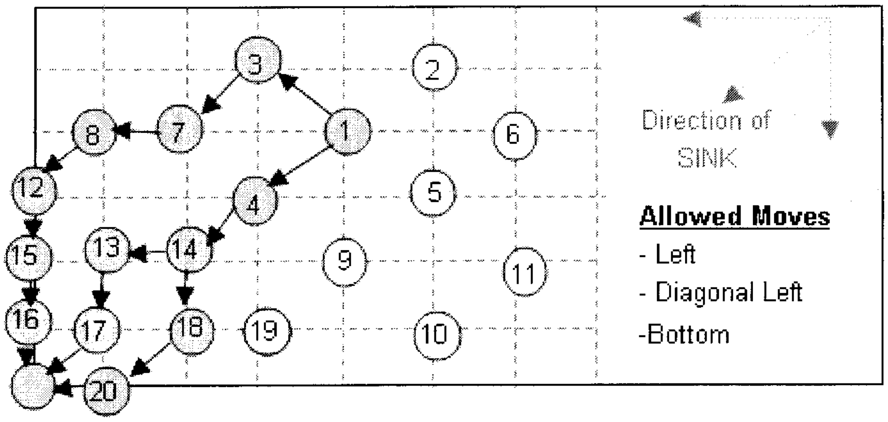
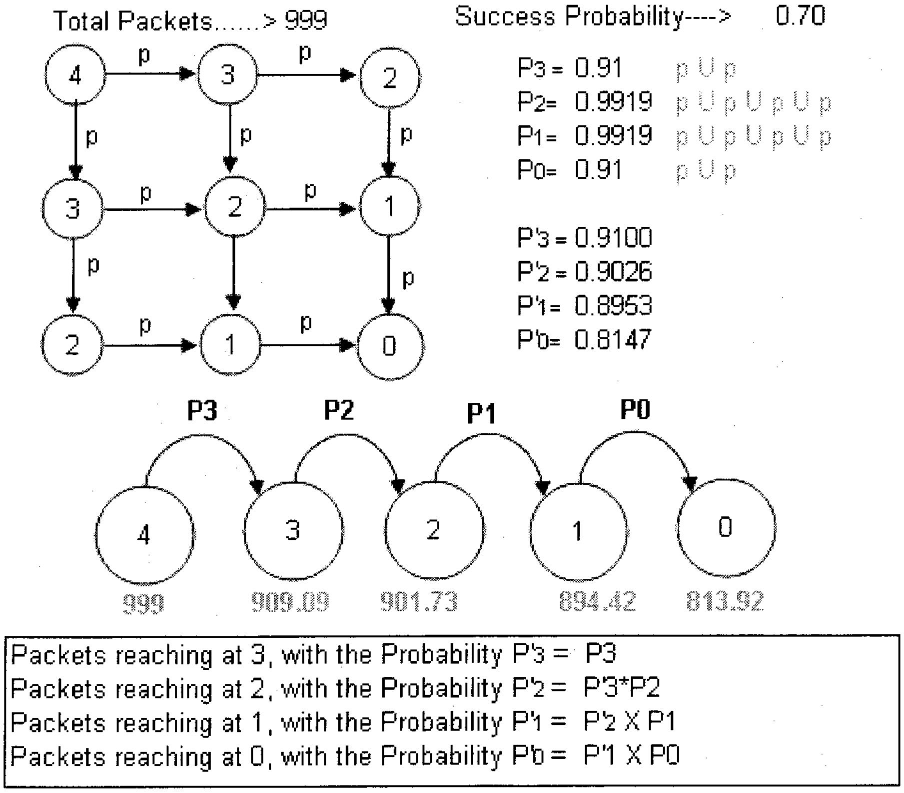


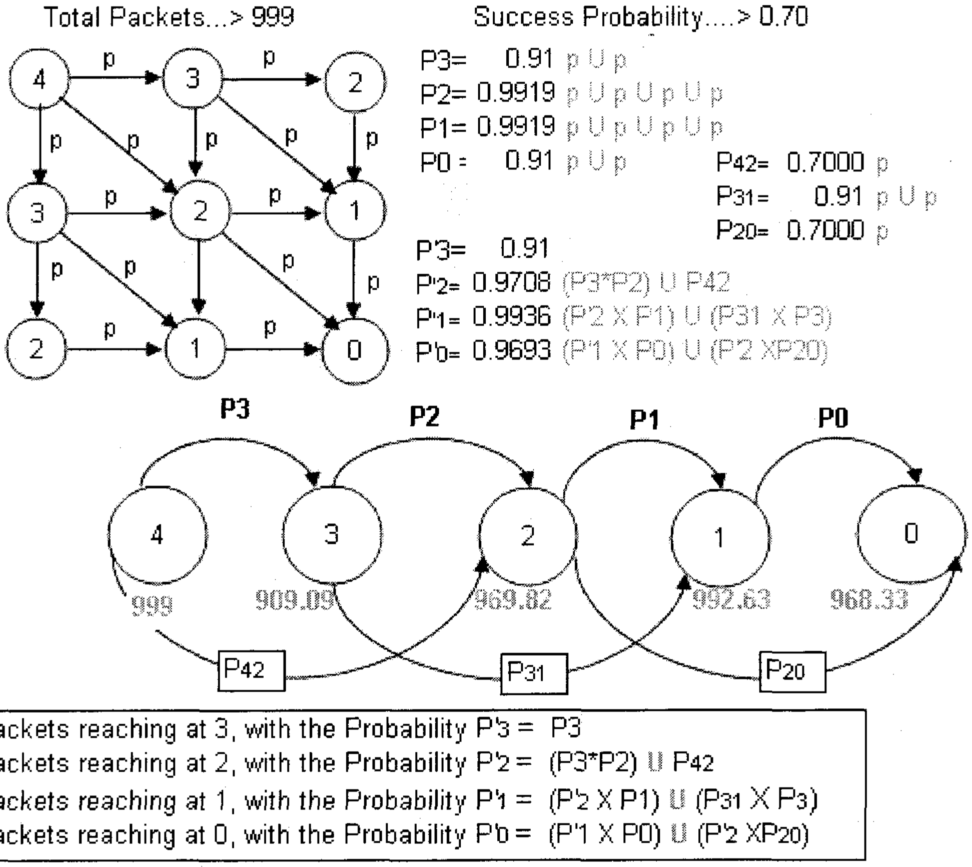
Figure 4.14 Directional Forwarding



Packets reaching at 3, with the Probability P'3 = P3
Packets reaching at 2, with the Probability P'2 = P'3*P2
Packets reaching at 1, with the Probability P'1 = P'2 X P1
Packets reaching at 0, with the Probability P'0 = P'1 X P0

Figure 4.15 Probabilistic analysis of simple label flooding

Choosing directional forwarding mechanism, helps propagate the labels from source in the direction towards the destination, thereby making the proposed scheme more energy efficient. Directional forwarding (figure 4.14) can focus the label tunnel in one direction and enhance the efficiency of the system. Figure 4.15 and Figure 4.16 shows the result of a probabilistic comparison between simple label flooding and directional forwarding. For the fixed time "t" and the success probability (to reach the sink) "p" as 70%, the simple label flooding scheme delivers 814 packets as compared to the directional forwarding where 969 packets have reached the sink. This is an increase of 19 % over the simple label flooding scheme. Performance evaluation of this scheme with more detailed comparison is described in chapter 5.



Total Packets...> 999

Success Probability....> 0.70

P3= 0.91 p U p

P2= 0.9919 p U p U p U p

P1= 0.9919 p U p U p U p

P0 = 0.91 p U p

P42= 0.7000 p

P31= 0.91 p U p

P20= 0.7000 p

P'3= 0.91

P'2= 0.9708 (P3*P2) U P42

P'1= 0.9936 (P'2 X P1) U (P31 X P3)

P'0= 0.9693 (P'1 X P0) U (P'2 XP20)

Packets reaching at 3, with the Probability P'3 = P3

Packets reaching at 2, with the Probability P'2 = (P3*P2) U P42

Packets reaching at 1, with the Probability P'1 = (P'2 X P1) U (P31 X P3)

Packets reaching at 0, with the Probability P'0 = (P'1 X P0) U (P'2 XP20)

Figure 4.16 Probabilistic analysis of Directional Forwarding of labels

Downside of this scheme is (a) the participating nodes falling in the directional path can run out of the battery quickly, (b) unavailability of neighboring node/s meeting the criteria for receiving the label. Moreover, employing this scheme encourages disparity in energy spending among the sensor nodes. Nevertheless this scheme is a suitable candidate for the low bandwidth consuming applications, where the node energy takes back seat as compared to the speed of retrieving the data.

4.6 Energy Management Proposal

Within a wireless sensor network cloud, the sensor nodes may be installed in remote areas where repairing and replenishing the nodes may be impossible. The life span of the sensor node deeply depends on its battery lifetime. To prolong the sensor network lifetime, minimum battery power of the sensor nodes should be consumed. A considerable change in the topology of the sensor network may take place with each failing node, which requires re-organizing and re-routing the information. The proposed energy management, which we describe in this section, is purely decentralized and consumes enough power for an efficient information transfer. We address a power management routing scheme appropriate for wireless sensor networks, which focuses on the dissemination of information from source to the sink. We concentrate on following two issues

- Restricting minimum transmission around the sink and
- Setting up a battery threshold value

we minimize transmission between the sink and its one-hop neighbors to reduce energy consumption, and apply a battery threshold value so that the probability of information packets being dropped significantly decreases. There are several benefits to do so. First, the one-hop neighboring node/s to the sink can transmit the information directly to the sink, instead of broadcasting. Second, it ensures the graceful degradation of the network in a low-energy network thereby, enhancing the fault tolerance of the system. A detailed explanation is followed in the subsections below.

4.6.1 Minimum Transmission around the Sink

Upon deployment of the sensor nodes in a network, all the nodes have an identical initial energy. Within a sensor cloud, variation in the rate of consuming power by each node depends on the various factors such as event sensing rate, distance from sink node, and location of each node relative to other nodes. This disparity in energy consumption in wireless sensor network causes an imbalance of node power status resulting in diminishing overall network lifetime. If the sink is at an unchanging location, data packets are collected from entire network to one fixed location. The data traffic at the sensor nodes located around the sink is denser than around the nodes located away from the sink. This indicates that those nodes located adjacent to the sink will expend more energy in node communication than those away from the sink. Nodes located in vicinity to the sink, when expended can isolate the sink from the entire sensor network, since no sensor node can reach the sink. To avoid the isolation of sink node from entire network, it is primarily important to adapt power management heuristic on nodes located around the sink. We propose the scheme to bind the communication between the sink and its one-hop neighbors. In our proposed IDLF algorithm, each sensor node has the knowledge of its neighboring nodes. The neighboring one-hop away nodes from the sink can directly transmit the label to the sink, instead of waiting to broadcast it to other neighbors. Consequently, the respective neighboring nodes to the sink expend less energy. A detailed evaluation of this scheme is presented in chapter 5.

4.6.2 Battery Threshold Value

As discussed in detail in previous chapters that in order to optimize the performance, wireless sensor networks lack the centralized controlling unit for monitoring. There is a high chance that the data packets can be dropped on the way to the destination. There could be several reasons for the data packets not delivered to the sink. One of the foremost reasons of data delivery failure is the limited battery power of the sensor node. In wireless sensor communication, continuous exchange of information consumes battery power and the bandwidth of a respective node. A communicating sensor node may not be aware of the battery status of the next hop node. It is highly likely that, while a node transmits the data packet to its one-hop neighbor, the neighbor node runs out of battery, or the data sending itself, runs out of battery. This results in losing the data packet. In [73], to evade the loss of data packet, a threshold energy value is established. This was referred to as Modified Conditional Max-Min Battery Capacity Routing (Modified-CMMBR). This scheme uses three possible battery threshold values. The source node picks a different routing scheme depending on the residual energy in the sensor nodes so that all nodes contribute to data propagation. When a node reaches a certain threshold value, the node sends a signal to the source node. After receiving the notification about the failure of node, the source then re-route the data using a different routing scheme depending on the available battery thresholds for the sensor network nodes. This scheme, however, generates transmission overhead by using control signals.

In our proposal, we employ a single battery threshold value. The threshold value is estimated using (a) total energy required to receive and broadcast a label, (b) receive and transmit a request, and (c) receive and transmit a data. This scheme does not need the control signal even if sensor node's battery level falls below threshold value. IDLF algorithm is used when the nodes operate above the threshold value. When the battery level of the sensor node falls below the threshold value, it does not stop participating. However, the node does not participate in the rest of routing stages. By utilizing the battery threshold value the probability of information packets being dropped significantly decreases. Threshold value for a sensor node is employed based on the IDLF algorithm. We assumed that a node has a maximum number of neighboring nodes, which is eight, so that as long as a node has a battery power over the threshold value, the node will never drop a data packet. Table 4.2 shows the assumptions for employing the threshold value. We set the total threshold value to be 160 energy units. The performance evaluation is shown in chapter 5

Table 4.2: Battery Threshold Value

Receiving Label (8 neighbors)	8 unit energy
Transmitting Label (8 neighbors)	24 unit energy
Receiving Request	1 unit energy
Transmitting Request	3 unit energy
Receiving Data	31 unit energy
Transmitting Data	94 unit energy
Total	160 unit energy

4.7 Reliable Multipath Information Dissemination

by Label Forwarding (RM-IDLF)

In this section we explain multipath routing by revealing its characteristic advantages and overhead. We then explain our motivation for applying multipath routing in the perspective of sensor networks and the associated challenges. This is followed by the explanation of the RM IDLF algorithm

4.7.1 Background

Multipath routing takes advantage of the connectivity of the underlying communication networks by providing multiple paths between source-destination pairs. For the robust exchange of information, sometimes it is desirable to allow packets with the identical source and destination to take more than one viable path. The reason for the multipath routing can either be to lessen the network congestion or to surmount node failures. The initial node therefore can have a preference of more than one potential path to a particular destination at any given time. There are two reasons to study the multipath routing [115].

(a) Load balancing- Data traffic between the source and destination is divided across multiple (partially or fully) disjoint paths to avoid congestion on any one path.

(b) Reliable Information retrieval- Employing multipath routing increases the probability of reliable data delivery due to use of independent paths [98]. Duplicate copies of the data may be sent along alternate routes, to guarantee the reliable data delivery.

In sensor networks, we study multipath routing to avoid inconsistency in power spending in the network. In comparison to the single path routing, multipath routing is advantageous for dense sensor node arrangement with heavy information traffic stream. With the scattered sensor nodes within the sensor field, there is absolutely no control over the topology of the network. Disseminating information load evenly among the nodes with discrete topological arrangement poses complexity. Single path routing is more desirable for the small set of information packet exchange. Multipath routing is economical in dense communication traffic. Implementing load balancing [115] is useful in sensor networks as the network lifetime depends more on the relative energy level than on the absolute energy level of the participating sensor nodes. In [116], assuming each sensor node having a fixed lifetime, the authors explained that the network lifetime can be enhanced, if the routing protocol minimizes the inconsistency in the residual energy of every node, rather than minimizing the total energy consumed in routing. For high density sensor networks, the connection throughput is improved through multipath routing. In [117], the information traffic is distributed proportionally over the nodes positioned at different paths between the source and the sink, with respect to their residual energy. This helps each node spend the same amount of energy for data transmission. The idea is to involve the under-utilized paths and relieve the over-utilized path during data communication.

Much energy in the network is consumed by the few nodes closer to sink, which is a bottleneck for network. In [124], the distributed sink has been proposed, where the information arrives at sink via multiple proxy nodes called "Prongs". These prongs are connected to the sink via high bandwidth links (i.e. when packet arrives at prongs, it is delivered intact to the sink). Each Information packet of "M" fragments is encoded to "M+K" fragments with an Erasure Code and sends it over the multiple disjoint paths to the prongs. The sink can reconstruct the packet, if it receives more than "M" fragments. The advantage is that the source can send most of the fragments on the path with the lowest energy and still achieve the desired reliability by increasing the number of parity fragments. The scheme suffers from the weakness that if more than N-M packets are lost, the system can not recover the entire data. Also erasure codes introduce a fixed redundancy overhead, wasting the bandwidth on all packets. [119] established random walks between a source and sink to prevent the overhead of caching multiple paths. The node failure is assumed to be temporary, as the nodes are to be powered by a renewable source of energy. The nodes are randomly failed to evaluate the performance of the scheme. In [120], Split multipath routing has been proposed, to improve the reliability of the network. It employs multipath concurrently by splitting the information among the promising paths. [121] used directed diffusion protocol [64] to execute multiple path routing. The routing load is spread on more than one path to avoid congestion on any one path. Alternate promising paths are discovered during the route discovery phase of the directed diffusion. Using probability one of the paths is chosen for routing.

In [98], a multipath scheme is proposed, the basic idea is to have a power efficient and yet resilient protocol. This protocol builds on the directed diffusion [64]. A primary path, which is considered the best from the application's point of view, is constructed (e.g. a low delay path). Small numbers of alternative paths are also constructed, which will be used in case of failure in the primary path. The source periodically floods low-rate data over all alternate paths, therefore permitting fast recovery from failures on the primary path. This approach eliminates the need to flood the entire network for a new path in case of failure along the current path (as done in directed diffusion).There are two different ways of constructing the multipaths

(a) Disjoint Multipaths

Disjoint routes are chosen so that a link failure in one route does not affect the others. A small number of alternate paths that are node-disjoint with the primary path, and with each other can be constructed .These alternate paths are thus unaffected by failures on the primary path, but can potentially be less desirable (e.g., have longer latency) than the primary path. But for the applications where the reliable information is to be transferred, formation of disjoint paths is very valuable.

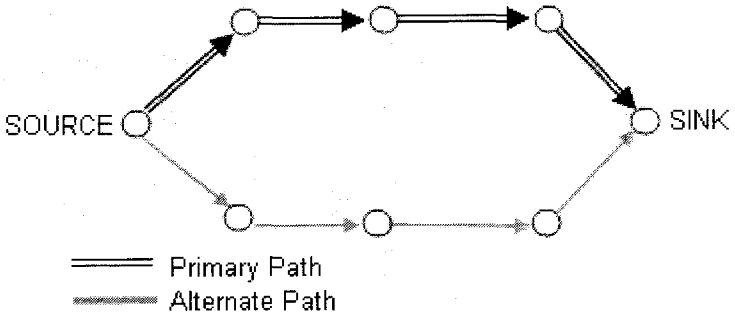


Figure 4.17 Disjoint Multipath Network

(b) Braided Multipaths

Braided multipath [98] waives the condition for the sensor node to be disjoint. Instead of not completely node-disjoint path, the alternate paths within a braid can partially disjoint from the primary path. For each sensor node on the primary path, find the best path from source to sink that does not contain that node. This alternate path may not necessarily be completely node disjoint with the primary path. This creates a braid-like path set consisting of a primary path and a series of alternate paths. Braided Multi-path increases the resilience of the path, but at a lower path maintenance cost. The links can be expected physically adjacent to the primary, and so it can be said, that the braid expends energy comparable to the primary path.

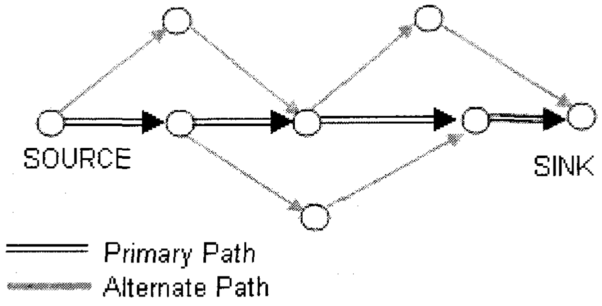


Figure 4.18 Braided Multipath Network

4.7.2 RM-IDLF Explanation

We have developed a deterministic model which is an extension to IDLF algorithm introduced in the previous sections. This scheme involves point to point data transmission where the source initiates the routing scheme and disseminates the information toward the sink. Similar to IDLF, a label path is formed, prior to the transmission of actual data packet/s. This label path is established by the source node and the labels are disseminated in the network. It should be noted at this point that for RM-IDLF we used an alternate disjoint path. This alternate path scheme (RM-IDLF) may have a higher path cost in terms of energy consumption, but is more reliable in terms of data packet delivery to sink than the single path scheme (IDLF). In the latter scheme, the protocol establishes multiple (alternate) disjoint path/s from source to destination with negligible control overhead to balance load due to heavy data traffic among intermediate nodes from source to the destination. Another point of interest in this framework is the study of trade-offs between the achieved routing reliability using multiple disjoint path routing and extra energy consumption due to the use of additional path/s. Also, the effect of the failed nodes on the network performance is evaluated within the sensor system.

Similar to IDLF, RM IDLF algorithm is divided into three stages:

- Label transmission stage,
- Request transmission stage, and
- Data transmission stage

At this point, it is to be noted that the elaborated explanation of each stage is described in section 4.4. In this section the brief overview of each stage will be explained. In addition, the features, in which the RM IDLF differs from the previous scheme is elaborated. Initially, an event is detected at the source; the source broadcasts the label to all the neighboring nodes. The label receiving node checks for the particular label in its label cache. If the received label already exists in the label cache, the node ignores the received label. If the node receives a fresh label, the receiving node stores the label in the cache and retransmits the label to its neighbors. Fig 4.19 shows that, the label is transmitted from node A to its neighbor nodes, B, C and D. Node B further transmits the label to E, F and C. Although C is the neighbor to B, but C already has the copy of label it received from A, so node C discards label from node B. Similarly C further transmits the label to G, H and so on. Node D stores the label in its label cache, because it has no immediate outlet to transmit the label

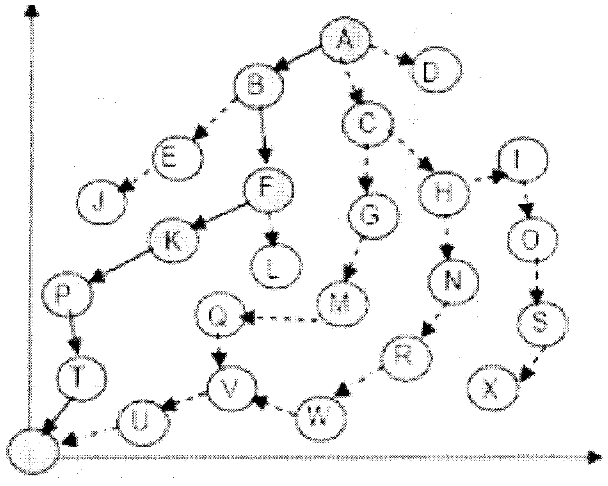


Figure 4.19 Label Transmission Path- RMIDLF

This label transmission process is repeated until the label reaches the sink or there is no more neighboring node, which does not have the label in its label cache. At this point all the nodes have the copy of the label. Figure 4.20 shows that the path A-B-F-K-P-T-SINK is the first label based path from source to sink. At this point the Sink replies back by sending a request packet toward the source. Similar to a label, this request packet is small in size compared to the actual data packet in order to minimize communication burden between sensor nodes. Request packet is similar to the label and much smaller than the actual data packet. As shown in Figure 4.21, the request packet follows the trace, on which the label moves across from the source to the sink, back to the source.

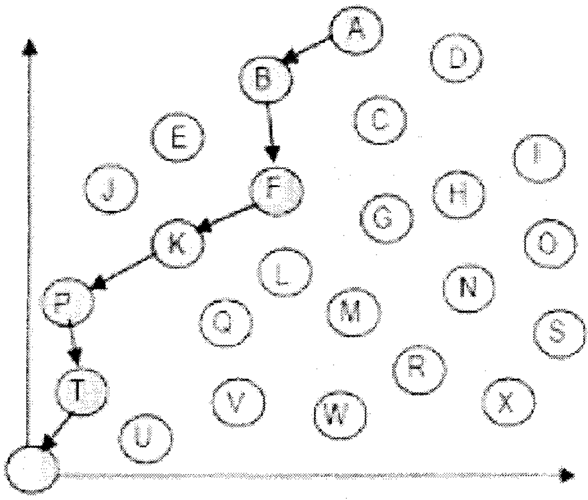


Figure 4.20 Label Propagation Path

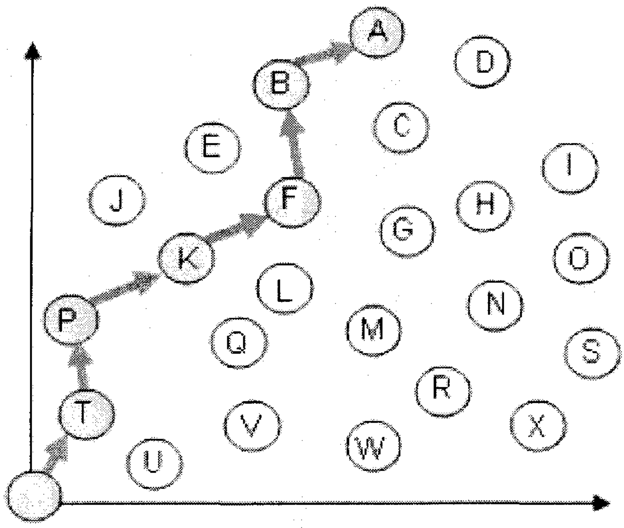


Figure 4.21 Request packet pursues the trace, on which the label traversed from the source to the sink- RM IDLF

On receiving the request packet source node performs two operations

1. Sends the Information packet along the Primary path, similar to IDLF

2. Initiate a new label path for the creation of Disjoint path

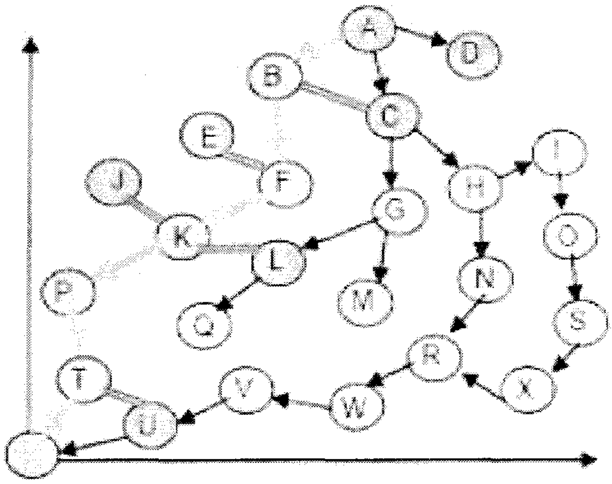


Figure 4.22 Creation of a new Label Path

Source node initiates a new label which is of the same size as the initial label. The source node A, (figure 4.22) has three neighboring nodes B, C and D respectively. Node A transmits the new label to nodes C and D. Node B rejects any label from Node A, because it is part of the initial or primary path. Similarly node C is the neighbor to node B, but node B rejects the copy of the new label from node C. Thus node C will only send the label to G and H. This will continue until

1. Either a disjoint label path to sink is created A-C-H-N-R-W-V-U-SINK (Figure 4.23), or
2. Disjoint path is not created – There is no node in the vicinity of the sender node (no outlet).Nodes Q and W has no immediate neighbors. In this case we are dependent on the primary path (Figure 4.24)

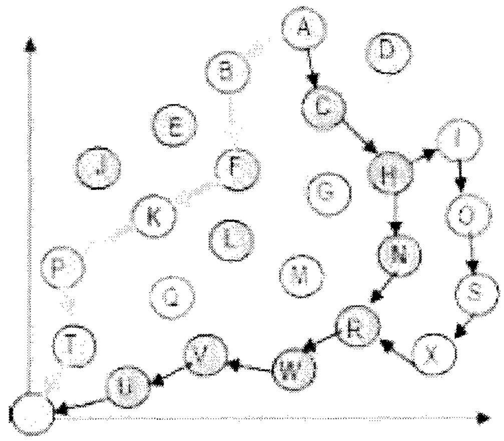


Figure 4.23 Creation of Disjoint Path

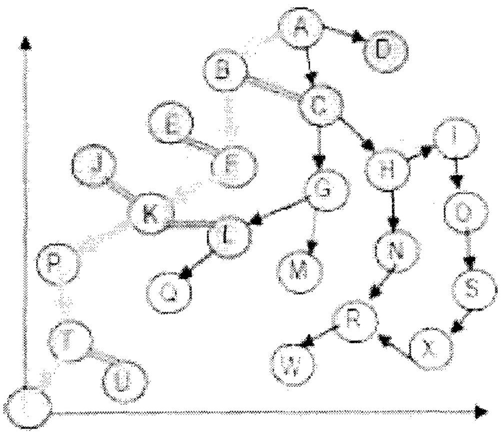


Figure 4.24 No outlet condition

It is obvious that maintaining multiple paths increase total power consumption. However, it increases the probability of information packets reaching destination node. The main purpose of using Reliable Multipath IDLF is to improve the fault tolerance of the sensor networks. Figure 4.25 and Figure 4.26 shows, In case of isolated node failures, existence of an alternate path helps diverting the information packets through the active nodes.

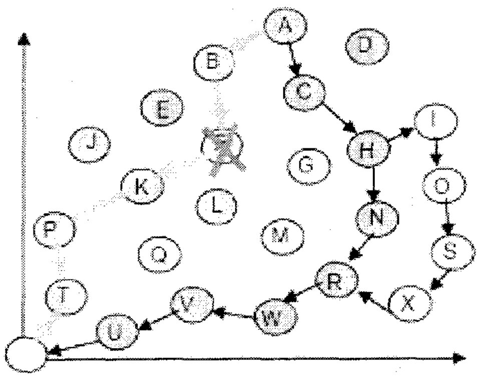


Figure 4.25 Node failure in path 1

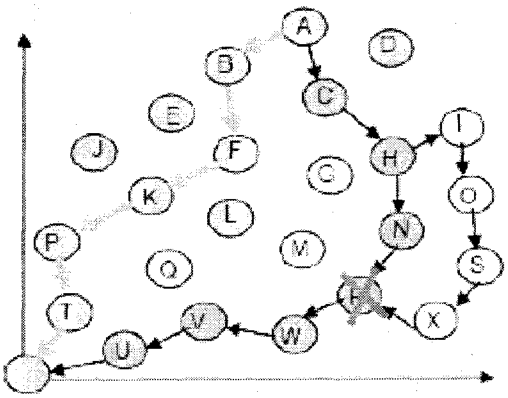


Figure 4.26 Node failure in path 2

RM-IDLF is resilient to node or link failure. If a route fails and the data delivery is not accomplished, the probability of information reaching the sink is high because the chance of failure of alternate path is small as compared to single path routing. For the experimental purpose, we took one alternate path. Depending on the application, the number of paths can be increased to make the system more robust. In single path routing, as the path fails, the sink stops receiving the data packets due to absence of a backup route. Also, in RM IDLF, there are no periodic updates to detect the availability of the alternate paths. This makes the sequential accuracy of the multiple paths independent of the frequency of the updates exchanged. Chapter 5 presents the performance evaluation of the proposed routing framework to establish its effectiveness in improving network lifetime, throughput, and quality of service

CHAPTER 5

PERFORMANCE EVALUATION OF THE DATA DISSEMINATION FRAMEWORK

In this chapter, we evaluate the performance of the label dissemination framework through extensive simulation results. We begin with a description of the simulation scenario, network topology and the simulator employed for conducting the experiments. We then demonstrate simulation results for IDLF and alternate path RM-IDLF and compare their performance with Flooding and SPIN protocol. The implementation of the energy management scheme is evaluated next. Faulty nodes scenario has been incorporated for each of the label dissemination schemes and finally the inferences are drawn from the outcome.

5.1 Simulation scheme

We designed a wireless sensor network simulator in C++. In our simulator, a specified number of sensor nodes, which is ranging from 2 to 100 nodes including one sink node, are randomly placed in a 10×10 unit sensor simulation grid. The sink node, marked black is at (0, 0) and is equipped with enough memory space, battery power, and processing speed that the power consumed by the sink can be excluded from the total power consumed by an entire sensor network during simulations. Every node in the network knows its coordinates in the physical field. The label dissemination protocols

IDLF and RM-IDLF intend to disseminate data towards the sink using negotiations.

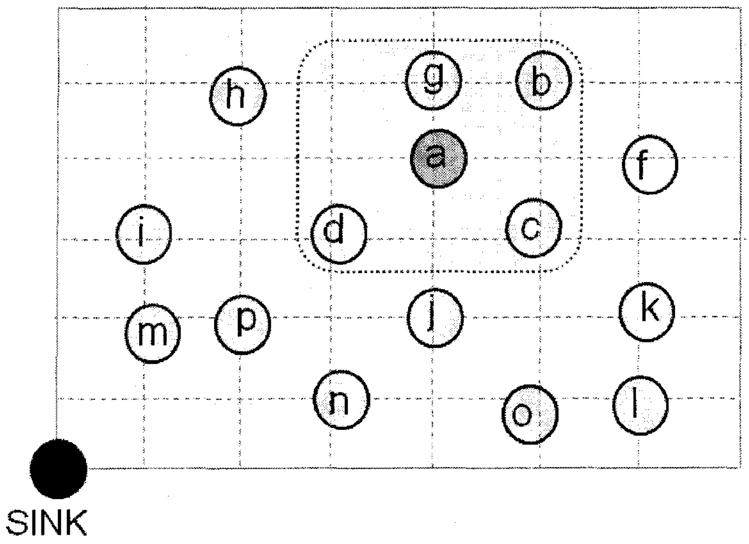


Figure 5.1 Sensor network design

An example of test sensor network configuration is shown in Figure 5.1. As explained in the last Chapter, each sensor node directly communicates with other nodes located within range “R1”distance, e.g. node “a” in figure 5.1 can directly communicate with nodes ‘b’, ’c’, ’d’” and “g” respectively. On the basis of node configuration of sensor nodes in our simulation, each node can have a maximum of eight neighbors’ .i.e. communication range ‘R1’ is the default metrics assumed in simulation results. When directional forwarding is applied, a node transmits a packet only to neighboring nodes, which are located closer to the sink than the sender. For instance node ‘j’ disseminates information packets only to nodes‘d’ and ‘n’ and not to ‘o’ and ‘c’. The amount of energy consumed for exchanging information during a neighbor discovering stage is the same for any routing protocol for the same network topology. Therefore, we do not consider energy consumed during neighbor discovery in our energy analysis. During a simulation, there will be only one source node in the sensor field at a time. In case of IDLF, after a data packet reaches the sink, a new source will be selected randomly. Then, the new source starts propagating a label. In RM-IDLF, after the data packet reaches the sink, the simulation ends and the new source is not selected randomly. In the latter case, we are more interested in tracking the rate of successful data packets reaching sink. The evaluation of RM-IDLF is mainly dependant on the tradeoff between energy consumption and the reliability of the network.

5.2 Performance Assessment –IDLF

IDLF, as explained in Chapter 4, will be compared initially with flooding and SPIN on the basis of (a) Energy consumption over time, (b) Data transmission over time, (c) Energy consumption by a data packet. Next, to measure the effectiveness of IDLF, we implement directional forwarding in IDLF, Flooding and SPIN. For the initial algorithm the energy supply of each sensor node is set to be unlimited. Later, the performance of proposed energy management scheme will evaluated; first, the effect of minimum transmission around a sink node on the energy consumption of sensor nodes will be studied and then, we will discuss the experimental results of employing a battery threshold value on sensor nodes. Finally, we will evaluate the RM –IDLF algorithm in the presence of fault-free and faulty nodes.

5.3 IDLF Evaluation

In IDLF evaluation, within the physical grid, we used 30 sensor nodes, each within the communication range of each other. A source node is assumed to be randomly selected. As explained in Chapter 4, we assumed that the size of data packet is 31 times greater than the size of the label and request packet. Then if we assume that transmitting a label or request packet between two neighboring nodes takes one unit time, transmitting a data packet will take 31 unit times. Also, we assumed that transmitting information consumes 3 times more energy per unit time than receiving.

5.3.1 Consumed Energy over Time

In this section we evaluate the performance of the IDLF algorithm on the basis of the total energy consumed over the simulation time. We then alter the number of nodes participating in the network. In addition to IDLF, we also simulate flooding (Flood), flooding with directional forwarding (Flood-D), IDLF with directional forwarding (IDLF-D), SPIN, and SPIN with directional forwarding (SPIN-D). Figures (5.2 -5.7) shows the consumed energy over the simulation time with different nodes for each of the above schemes. The experimental results illustrate that when the number of sensor nodes in a network is less, the cost of disseminating the data packets is not high. This holds true for any routing scheme.

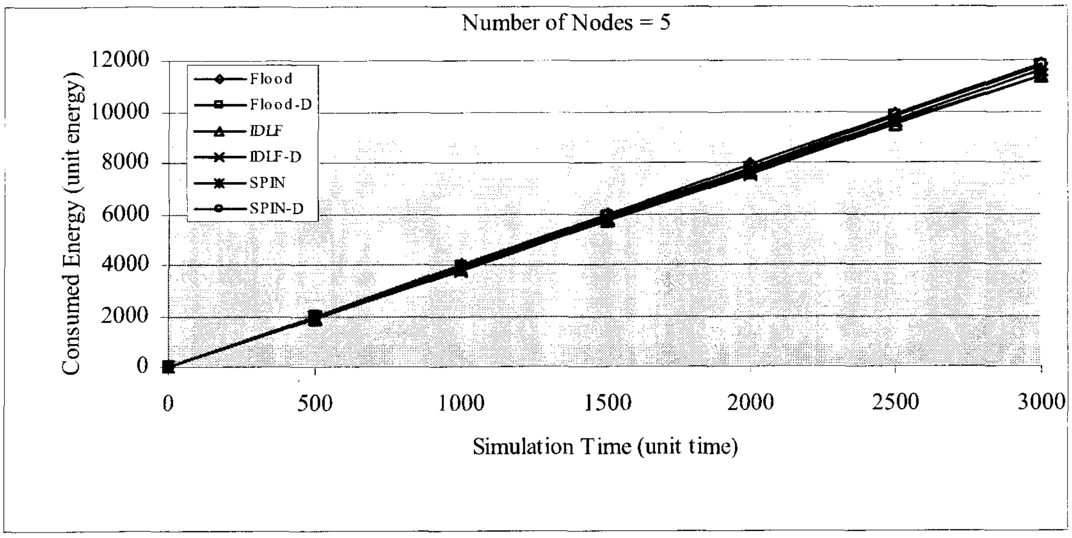


Figure 5.2 Energy Consumed by entire sensor network over time - 5 nodes

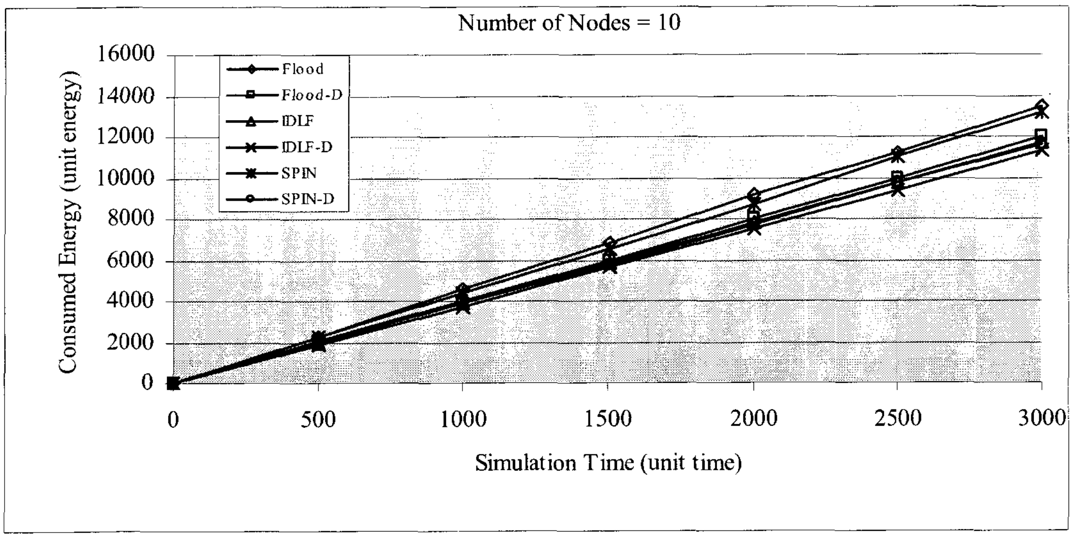


Figure 5.3 Energy Consumed by entire sensor network over time - 10 nodes

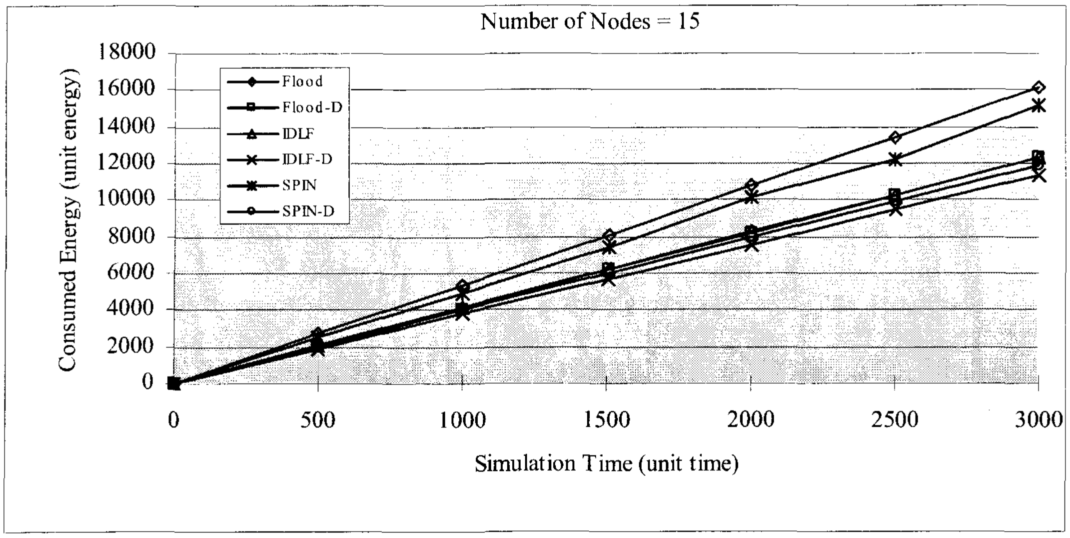


Figure 5.4 Energy Consumed by entire sensor network over time - 15 nodes

Figure 5.2 shows that for the total of 5 nodes within the sensor network, the difference in the amount of spent energy by each routing scheme is small. As expected, after the simulation time of 3000 units, the flooding consumes 4.3% more energy than IDLF with directionality (IDLF-D).The difference in energy consumption among routing schemes becomes quite obvious as the number of nodes in a network increases. Figure 5.3 shows that, for 10 nodes in the network, this difference in energy is 19% more than IDLF-D. Similarly for 15 nodes, the difference is 43% more than IDLF-D (Figure 5.4), for 20 nodes, the difference is 64% more than IDLF-D (Figure 5.5) and for 25 nodes, the difference is 89% more than IDLF-D (Figure 5.6). For 30 sensor nodes in a network after 3,000 simulation time, Flood, Flood-D, IDLF, SPIN, and SPIN-D consumed 112, 21, 29, 83, and 15% more energy than IDLF-D, respectively.

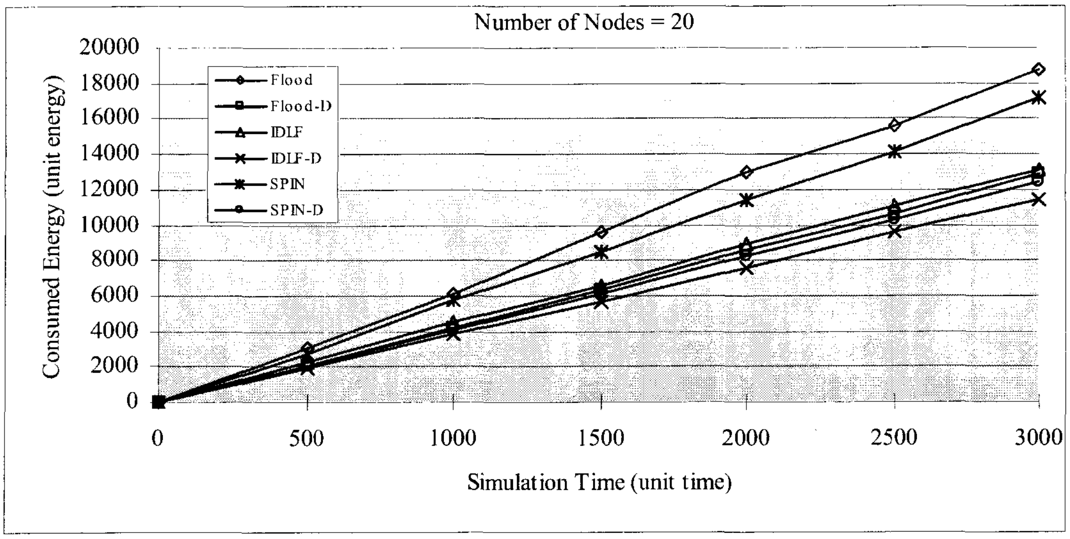


Figure 5.5 Energy Consumed by entire sensor network over time - 20 nodes

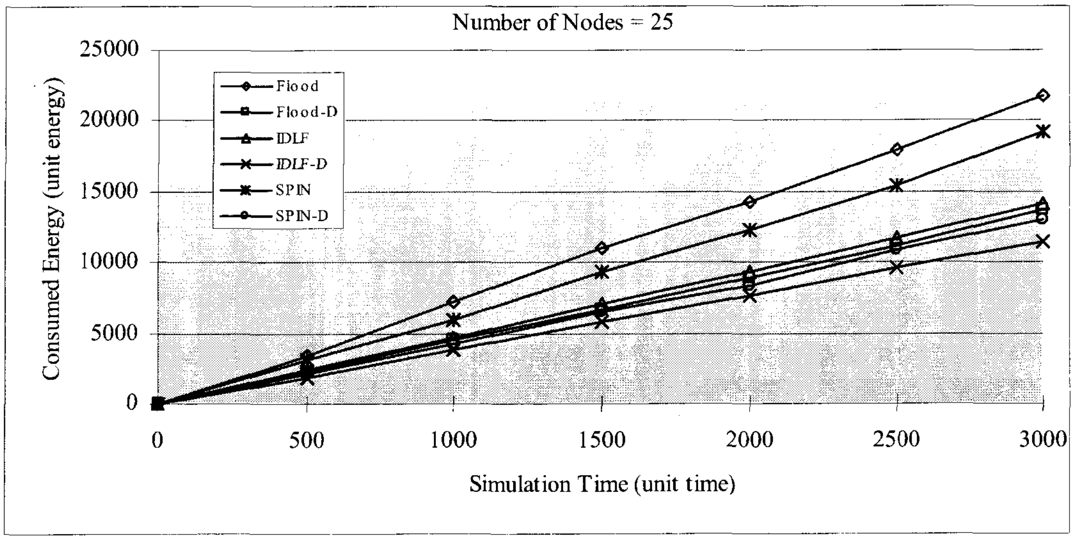


Figure 5.6 Energy Consumed by entire sensor network over time - 25 nodes

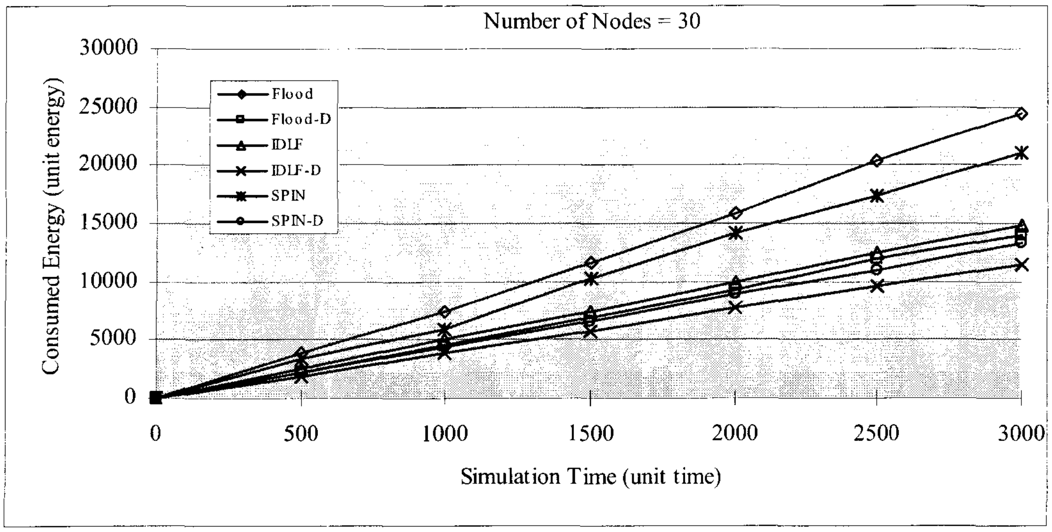


Figure 5.7 Energy Consumed by entire sensor network over time - 30 nodes

The experimental evaluation of IDLF on the basis of expended energy shows that by applying directional forwarding each routing scheme achieves significant energy savings. This is true irrespective of the total number of nodes within the sensor network. We also conclude that the flooding and SPIN consumes more energy than IDLF (with or without directional forwarding). This is because flooding and SPIN are intended for disseminating data packets through the entire sensor network, whereas, IDLF is a point-to-point data transmission. Also, SPIN performs better than flooding in terms of energy consumption, which is true, because it is designed to prevent implosion and overlap in flooding.

5.3.2 Data Transmission Over Time

This section explains the simulation results, where for a given time; we accumulate the number of data packets arriving at the sink. We then vary the number of participating nodes to study the network behavior. To explain data dissemination efficiency for various routing schemes, we incorporated the comparison of IDLF with flooding (Flood), flooding with directional forwarding (Flood-D), IDLF with directional forwarding (IDLF-D), SPIN, and SPIN with directional forwarding (SPIN-D).

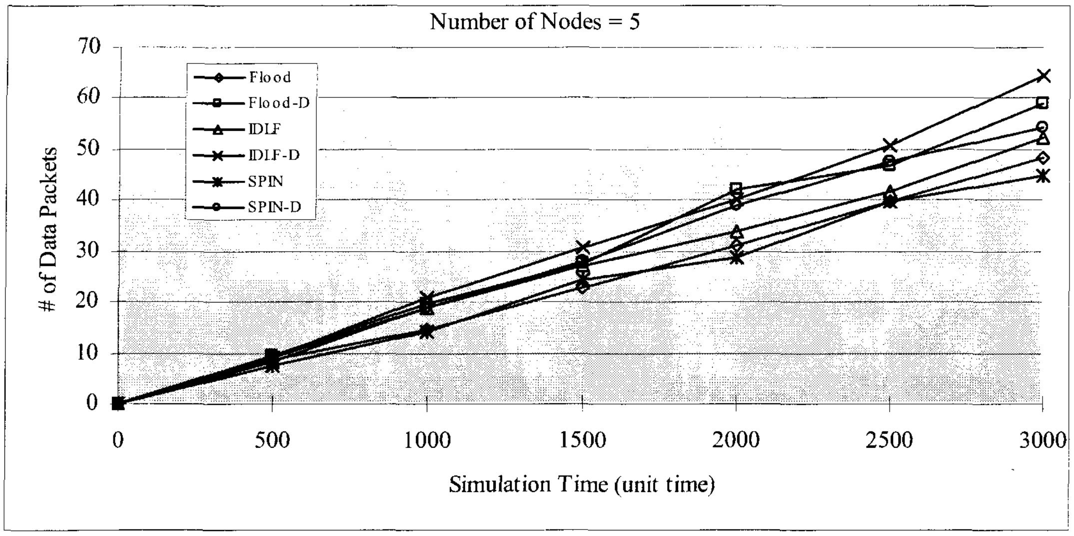


Figure 5.8 Number of Data Packets Delivered at Sink over Time- 5 nodes

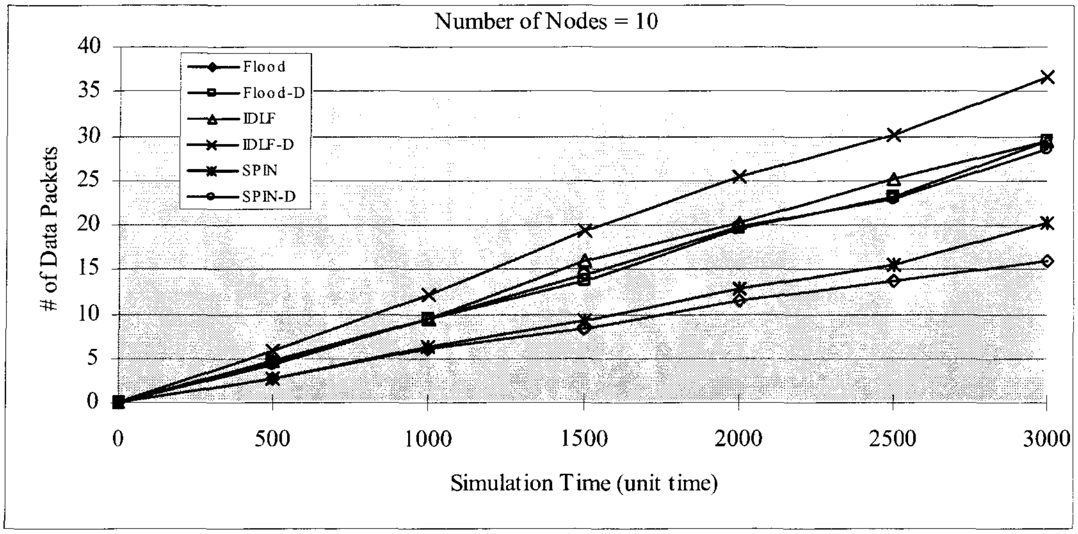


Figure 5.9 Number of Data Packets Delivered at Sink over Time- 10 nodes

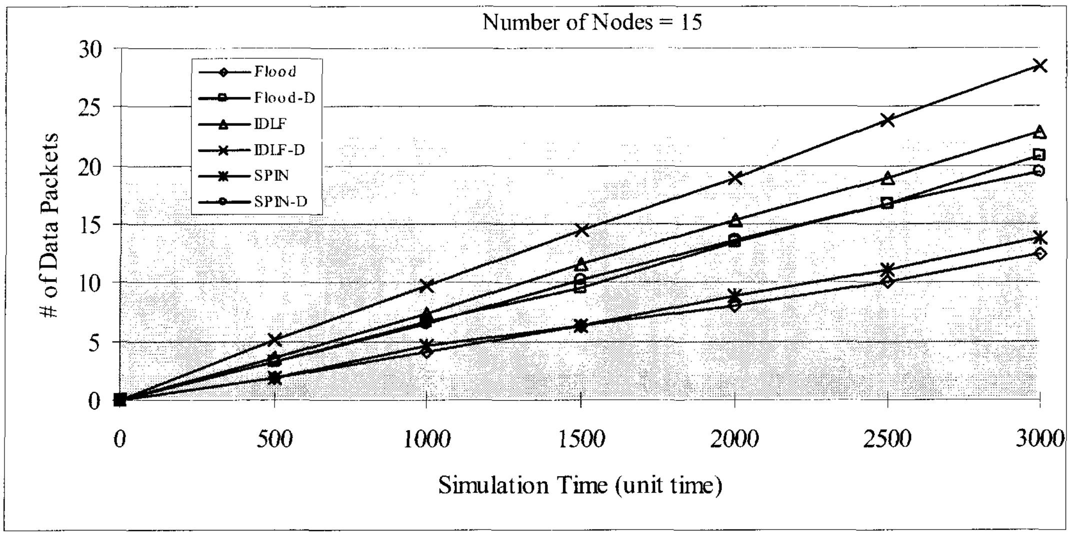


Figure 5.10 Number of Data Packets Delivered at Sink over Time- 15 nodes

Figure (5.8 – 5.13) shows the number of data packets arrived at a sink for a given simulation interval. We then vary the number of nodes and analyze the performance of each. For a given time interval, the number of data packets delivered to sink diminishes when the number of nodes in a network increases. This is because the average distance from a source to a sink increases, so it takes more time to deliver a data packet. IDLF with directional forwarding always outshines the routing without directional forwarding. In terms of number of data packets delivered. Flooding with directional forwarding conveys 22 to 50% more data packets than normal flooding. SPIN improved its performance between 20 to 34% by adopting directional forwarding. Likewise, IDLF improved between 23 to 40%. Applying directional forwarding highly improves the data transfer efficiency in flooding. This shows that flooding exchanges more information among the nodes than other two protocols.

In case of IDLF, it performs better than SPIN and flooding. IDLF delivers 101% more data packets than flooding and 72% more data packets than SPIN respectively for

30 sensor nodes in a network. Similarly, in a given simulation time interval, IDLF with directionality delivered more data packets to sink than other two routing schemes with directionality. IDLF-D delivered 80% more data packets than Flood-D and 68% more data packets than Flood-D and SPIN-D at 30 sensor nodes respectively. The number of data packets delivered by Flooding with directionality is very close to that delivered by SPIN with directionality. By applying directional forwarding, inefficiency caused by implosion and overlap in flooding is minimized. Thus, flooding can perform as better as SPIN in this scenario. Furthermore, SPIN has to exchange ADV and REQ packets before transmitting actual data. In some cases, these two packets become overhead compared to just transmitting data only. Therefore, in this simulation Flood-D could deliver as many packets as SPIN-D.

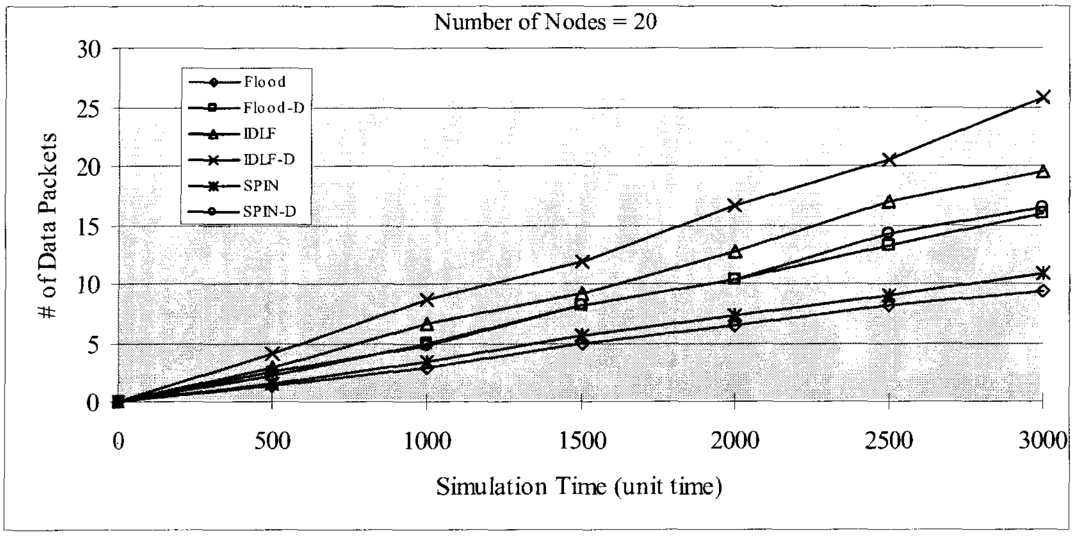


Figure 5.11 Number of Data Packets Delivered at Sink over Time- 20 nodes

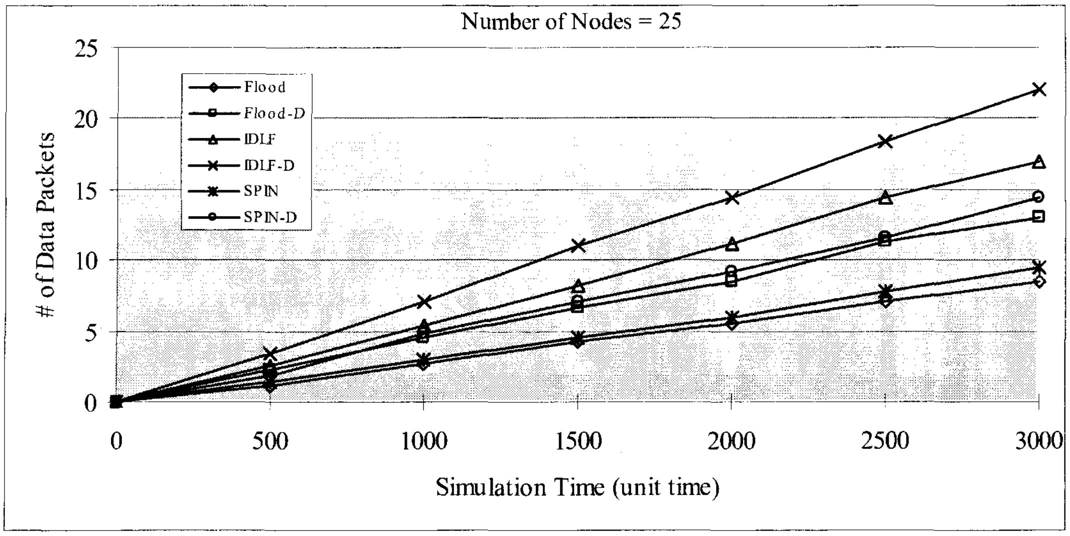


Figure 5.12 Number of Data Packets Delivered at Sink over Time- 25 nodes

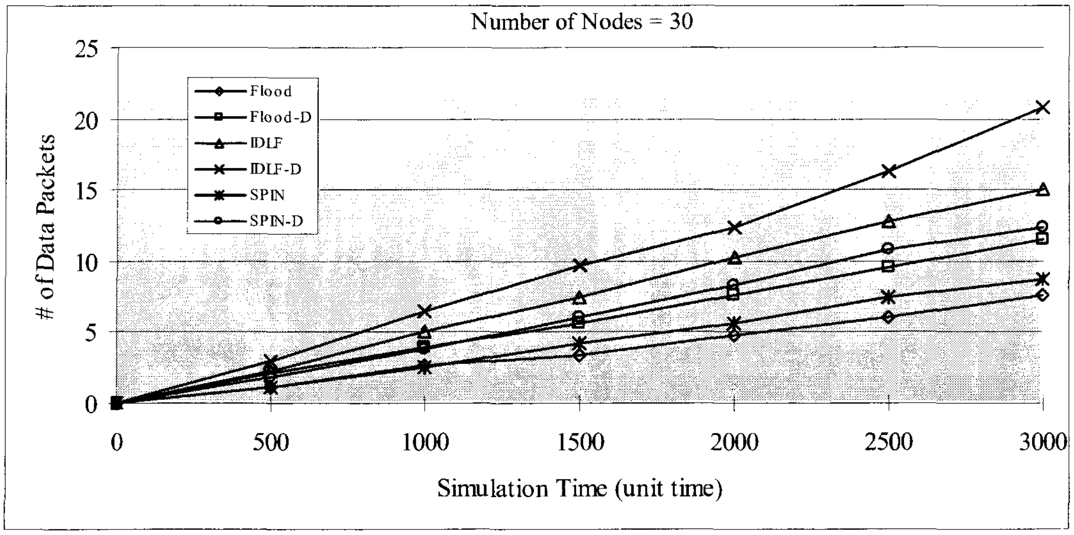


Figure 5.13 Number of Data Packets Delivered at Sink over Time- 30 nodes

5.3.3 Energy Consumption per Data Packet

This section compares for each scheme, the average energy consumed by delivering each data packet from a source to a sink. Varying the number of nodes in the sensor network, IDLF-D outperformed other five routing schemes. In directional forwarding, the sensor nodes narrow the range of broadcasting data packets by restricting communication only to the nodes lying in the direction towards the sink node. With IDLF-D, the energy consumed per data packet remains less as compared to IDLF, SPIN-D, Flood-D, SPIN and flooding (in the same order).For 5 sensor nodes, IDLF-D consumed only 34, 8, 17, 37, and 8% less energy for each data packet than flooding, Flood-D, IDLF, SPIN, and SPIN-D respectively, (Table 5.1). When the number of sensor nodes is increased to 30, IDLF consumes 447, 99, 67, 325, and 82% less energy for each data packet than flooding, Flood-D, IDLF, SPIN, and SPIN-D respectively (Figure 5.14).

Table 5.1 Energy consumed by transmitting each data packet

Number of Nodes	Flood *Energy (units)*	Flood-D *Energy (units)*	IDLF *Energy (units)*	IDLF-D *Energy (units)*	SPIN *Energy (units)*	SPIN-D *Energy (units)*
0	0	0	0	0	0	0
5	251	203	220	188	257	204
10	811	424	394	307	712	413
15	1345	618	548	391	1160	601
20	2050	833	699	461	1608	782
25	2676	1007	863	533	2058	951
30	3294	1198	1006	602	2557	1094

This pattern signifies that IDLF-D is appropriate for wireless sensor networks since scalability is one of the main concerns in wireless sensor networks. As explained in Chapter 4, the limitation of IDLF-D is that (1) the participating nodes located in the directional path can run out of the battery quickly, and (2) unavailability of neighboring node/s meeting the criteria for receiving the label. Moreover, employing directional forwarding encourages disparity in energy spending among the sensor nodes. Nevertheless this scheme is a suitable candidate for the low bandwidth consuming applications, where the node energy takes back seat as compared to the speed of retrieving the data.

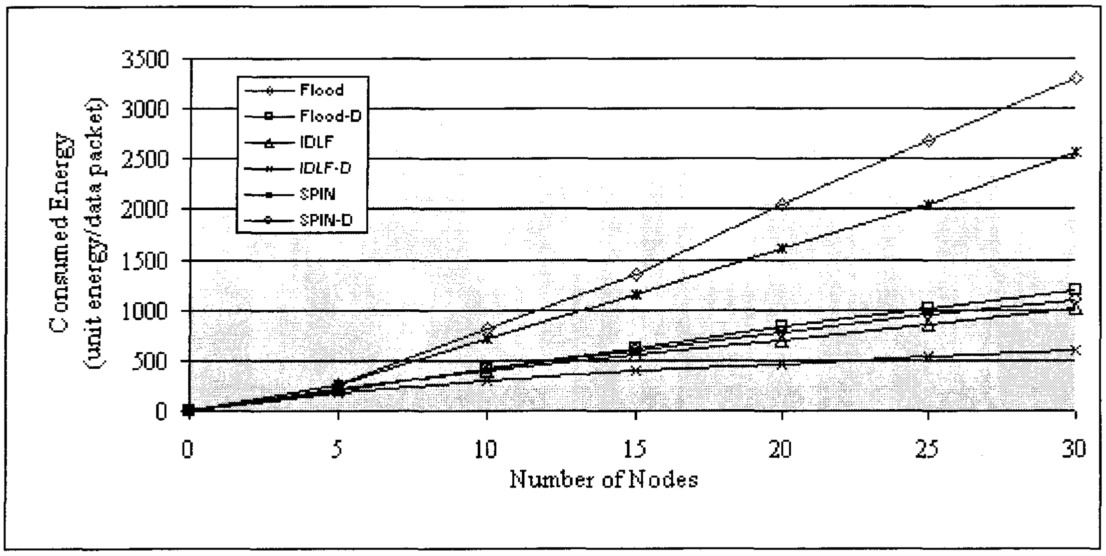


Figure 5.14 Energy consumed per Data Packet

Furthermore, IDLF requires less energy to deliver a data packet than flooding, SPIN, Flood-D and SPIN-D. This explains that IDLF, performs better in terms of energy consumed and data packets delivered. Even incorporating directional forwarding in flooding and SPIN cannot impact IDLF in its performance. Therefore, IDLF is more appropriate for disseminating information point-to-point in wireless sensor networks than flooding and SPIN.

5.4 Energy Management

In this section, we analyze our power saving and management schemes. We study the data collected to measure the importance of our scheme. In this power saving scheme, we minimize transmission between the sink and its one-hop neighbors to reduce energy consumption, and apply a battery threshold value so that the probability of information packets being dropped significantly decreases. The primary purpose is to ensure the uniform degradation of the network and enhance the fault tolerance of the system. First, the simulation results for measuring the effect of minimum transmission around a sink node on nodes' energy consumption are studied. Then, we discuss the experimental results of employing a battery threshold value on sensor nodes.

5.4.1 Minimum Transmission around the Sink

Data traffic at the sensor nodes located around the sink node is intense than around the nodes located away from the sink. The nodes situated adjacent to the sink will expend more energy in node communication than those away from the sink. These nodes, when expended can isolate the sink from the entire sensor network, since no sensor node can reach the sink. To avoid the isolation of sink node from entire network, it is primarily important to adapt power management heuristic on nodes located around the sink. In the minimum transmission scheme, each sensor node has the knowledge of its neighboring nodes. The neighboring one-hop away nodes from the sink can directly transmit the label to the sink, instead of waiting to broadcast it to other neighbors. Consequently, the respective neighboring nodes to the sink expend less energy. We simulated two scenarios – with and without transmission control around a sink node – and collected energy consumption data of nodes. Simulations for each scenario were performed using IDLF algorithm using 30 sensor nodes for 3,000 unit time. The average energy consumed by sets of sensor nodes, which are located in the same number of hops from the source node, is calculated and graphed. The result is shown in Figure 5.15.

Table 5.2 Average Energy consumed per node at different hop location

from the source node

Number of Hops from the source node	Unrestricted Broadcast *Energy (units)*	Restricted Broadcast *Energy (units)*
1	1046	791
2	764	763
3	591	613
4	462	473
5	351	370
6	296	272
7	260	228
8	196	192
9	141	99

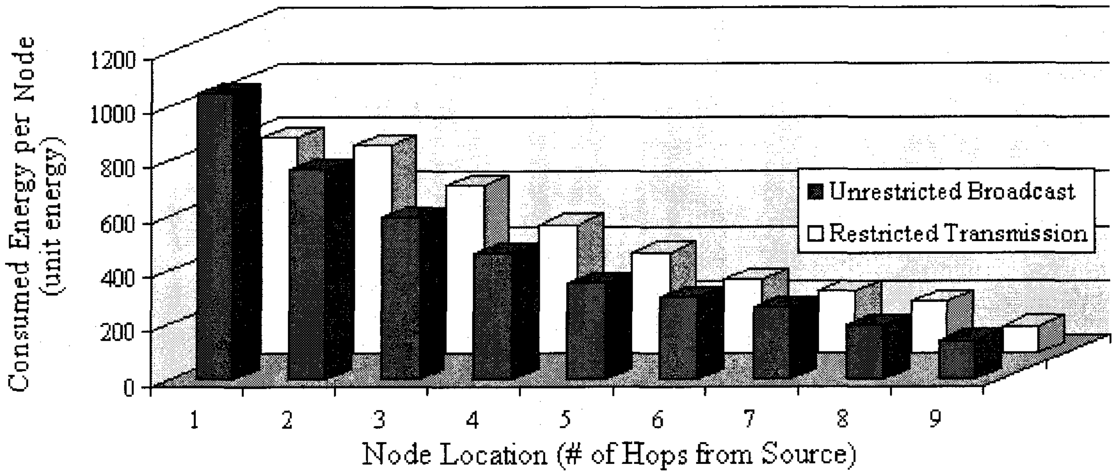


Figure 5.15: Average Sensor Node Energy Consumption by Hop Count

The simulation results in Table 5.2, shows that nodes located one hop away from the sink consumed an average of 1046 unit energy in unrestricted broadcast scenario. On employing the restricted transmission scheme, the average energy consumed by nodes is decreased to 791 unit energy. This shows 24.37 % improvement. Additionally, the energy consumption of other nodes is not affected. Hence by applying the restricted transmission scheme, total energy consumption of a sensor network can be reduced. Nodes located one hop from the sink in unrestricted broadcast scenario consumed 7.5 times more energy than those locates nine hops from the sink

5.4.2 Battery Threshold Value

We employ a single battery threshold value using the IDLF algorithm. The threshold value is estimated using (1) total energy required to receive and broadcast a label, (2) receive and transmit a request, and (3) receive and transmit a data. Utilizing the battery threshold value does not need the control signal even if sensor node's battery level falls below threshold value. The idea is that the sensor node does not stop participating even if the battery level of the sensor node falls below the threshold value. However, the node does not participate in the rest of routing stages. By incorporating the battery threshold value the probability of Data packets being dropped significantly decreases. Threshold value for a sensor node is employed based on the IDLF algorithm. We assumed that a node has a maximum number of neighboring nodes, which is eight, so that as long as a node has a battery power over the threshold value, the node will never drop a data packet. We set the total threshold value to be 160 energy units (as explained in Table 4.2, chapter 4). We execute the simulations for different number of nodes in a sensor network – 25, 30, 35, 40, and 45 nodes – with four different initial nodes energy – 600, 700, 800, and 900 respectively. Simulation time was set to 3,000 unit time.

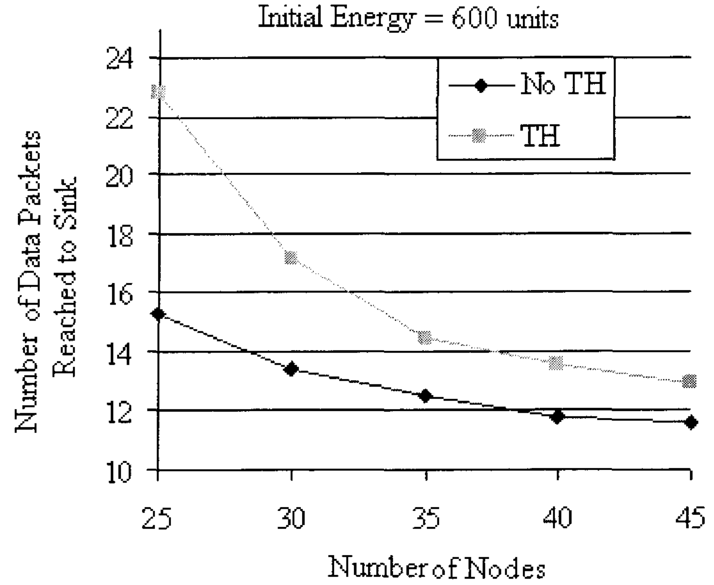


Figure 5.16 (a) Number of Data Packets Delivered vs. Number of Nodes

600 unit energy

Figures 5.16(a, b) and 5.17(a, b) shows the number of data packets reaching to the sink for a given simulation time. To explain this scheme, we took two cases- Initial battery energy with and without threshold –TH and No TH. We observe that, as the number of sensor nodes increases in the network, the number of data packets delivered to sink decreases. This holds true for any initial node energy. The reason for this behavior is, when the number of nodes increases in the network, the average distance from a source to the sink also increases. Thus, it takes more time for each data packet to reach to sink and consequently, for a given time, the number of data packets delivered is reduced. Secondly, as the number of sensor nodes increases, the difference in number of data packets delivered between the two cases (a) with threshold and (b) without threshold value is minimized.

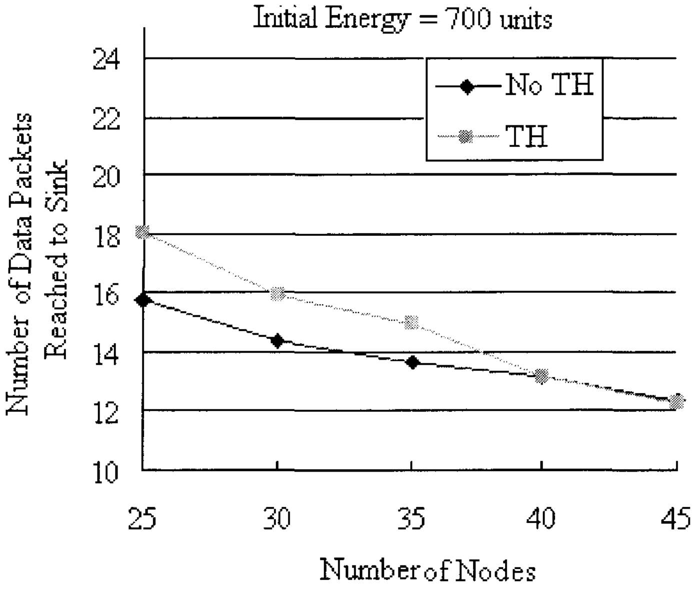


Figure 5.16 (b) Number of Data Packets Delivered vs. Number of Nodes

700 unit energy.

Figure 5.16, 5.17 shows that a network with a large number of nodes propagates less number of data packets. In IDLF algorithm, source node is randomly selected and the data path is selected each time. If the number of sensor nodes in the network is large, the probability of each sensor node selected as a part of data path is small. It reduces energy consumption of each sensor node. Due to this reason, power failure of sensor nodes in a network with large number of sensor nodes is less likely to happen than that with smaller number of sensor nodes [118]. Thus we conclude that for the larger number of sensor nodes, the data packets delivered by a routing with threshold value becomes close to that delivered by a routing without threshold value, i.e. there is no difference between a routing with and without battery threshold value. Applying a threshold value is more appropriate at low initial battery energy with less number of participating nodes. At low initial battery energy, sensor nodes fail more quickly than those at high initial battery energy.

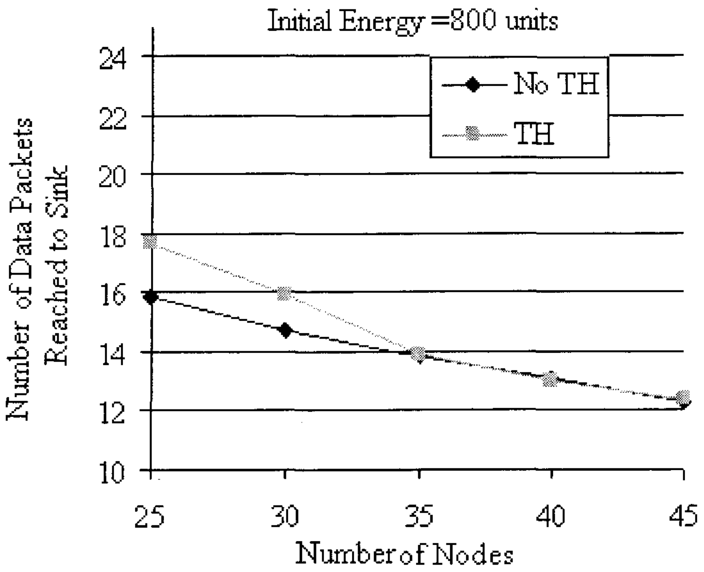


Figure 5.17 (a) Number of Data Packets Delivered vs. Number of Nodes

800 unit energy

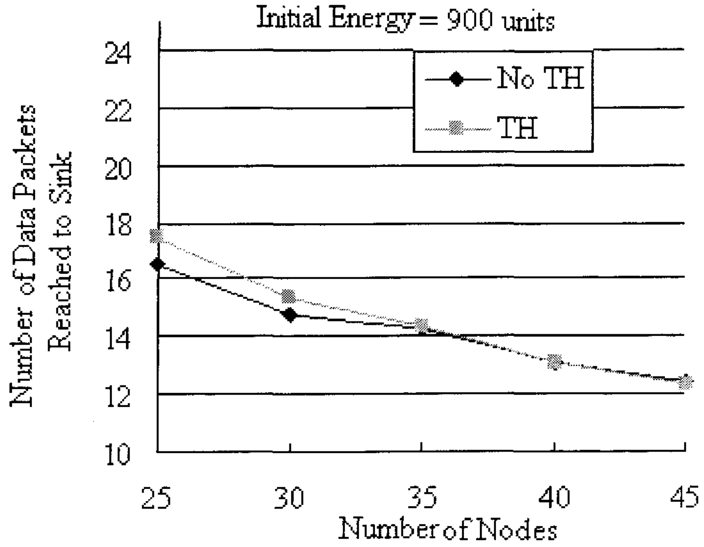


Figure 5.17 (b) % of Data Packets Delivered vs. Number of Nodes -900 unit energy

Figures (5.18 – 5.21) show the energy consumed by an entire network vs. number of nodes for different values of initial battery. We observe that, for a smaller value of initial battery, the difference between (a) with threshold and (b) without threshold is more significant as compared to the larger value of initial energy in sensor nodes respectively. At initial energy of 600 units, a network with threshold value consumed more energy than that without a threshold value. The reason for consuming more energy is that the number of data packets delivered is more in the threshold case. Figures 5.16 and 5.17 indicate that a network with a threshold value delivers more data packets than that without a threshold. As the initial battery energy increases, the difference in energy consumption between with and without a threshold value is reduced.

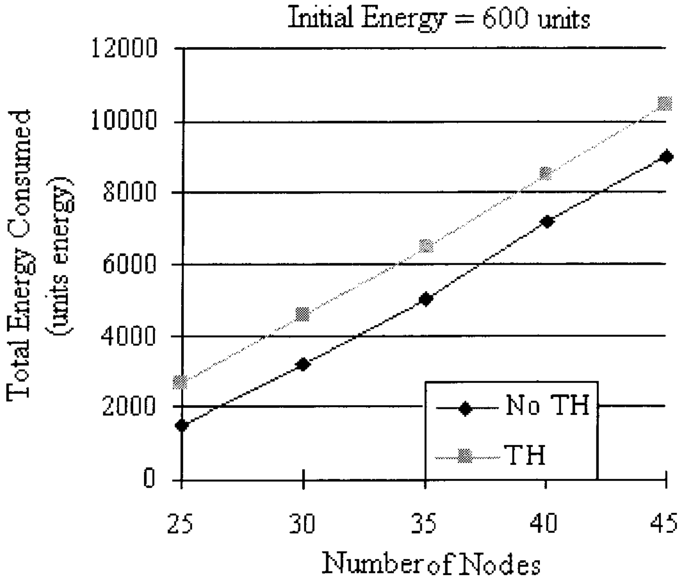


Figure 5.18 Energy consumption in sensor network - 600 unit energy

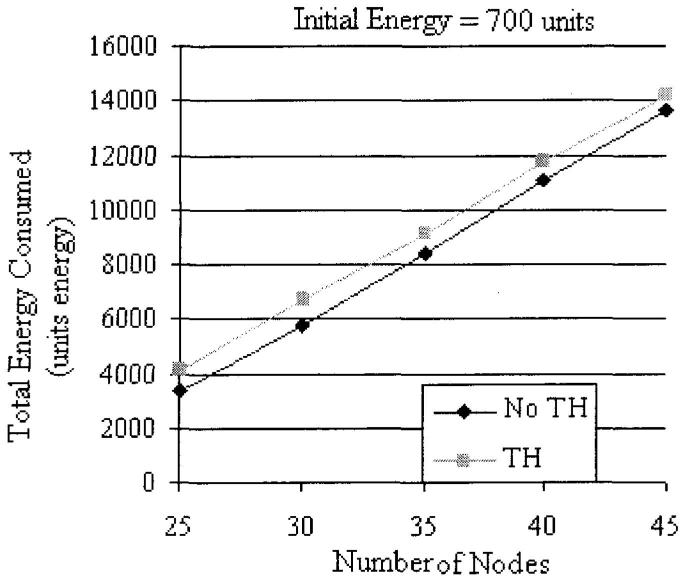


Figure 5.19 Energy consumption in sensor network - 700 unit energy

There is a change in power consumption for given number of nodes with respect to initial battery energy. Figure 5.16 shows that the number of data packets delivered to the sink does not change abruptly for a given number of sensor nodes (except at 25 nodes with initial energy of 600). Total energy consumption by a network for a given number of sensor nodes steadily increases as initial battery energy increases. However (a) the number of nodes in a network, (b) the number of data packets delivered, and (c) the simulation time are the same, there should be another factor, which affects energy consumption. In IDLF algorithm, this factor can be attributed to the phenomenon of the label exchange. For the large initial battery energy, the number of nodes running out battery in a given simulation time will be reduced. Hence, more nodes participate in exchanging labels. Therefore, at higher initial battery energy, more total energy is consumed by a network.

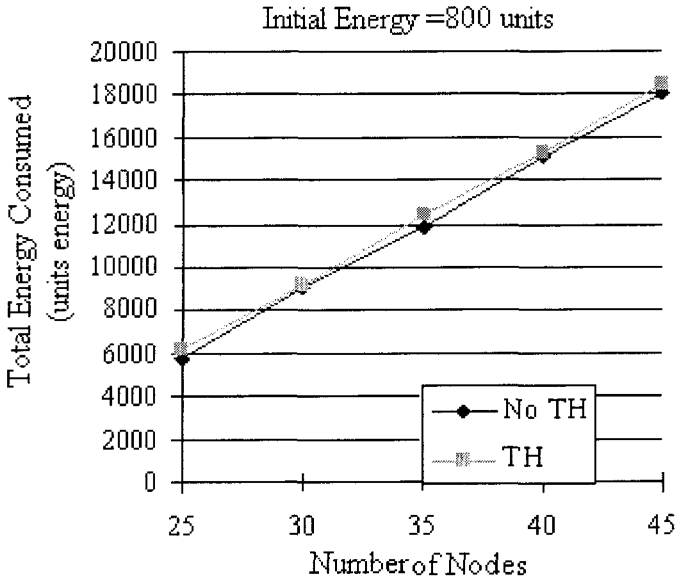


Figure 5.20 Energy consumption in sensor network - 800 unit energy

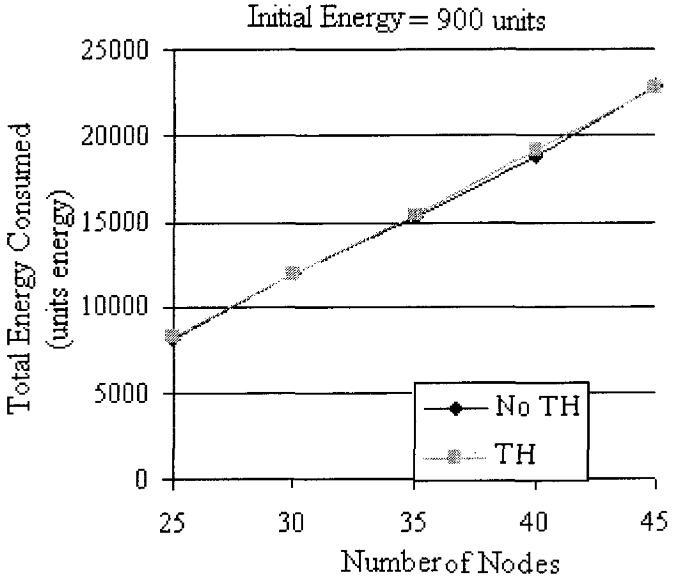


Figure 5.21 Energy consumption in sensor network - 900 unit energy.

The percentage of information packets dropped vs. number of sensor nodes are shown in figures (5.22 - 5.25). In IDLF, three types of packet drop occurs a) label drop,

b) request drop and 3) data drop. A data packet may never reach sink if either request drop or data drop happens. However, the label drop can occur because it is broadcasted in the network. Many copies of the label packets may travel the sensor network at the same time. In a case, when label never reached sink node, it is considered to be a label drop. In case of IDLF with threshold value, only label drop occurs. Due to threshold value, request drop and the data drop may never happen, because a node does not participate in information transmission at all.

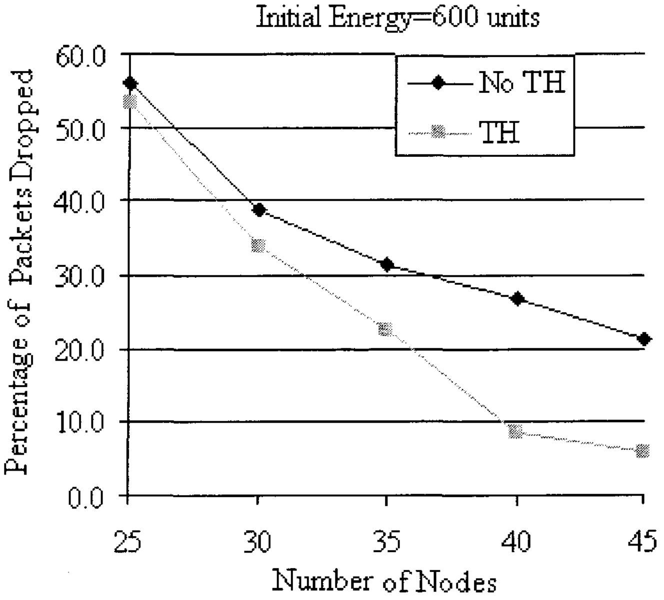


Figure 5.22 Percentage of Data Packets Dropped- 600 unit energy .

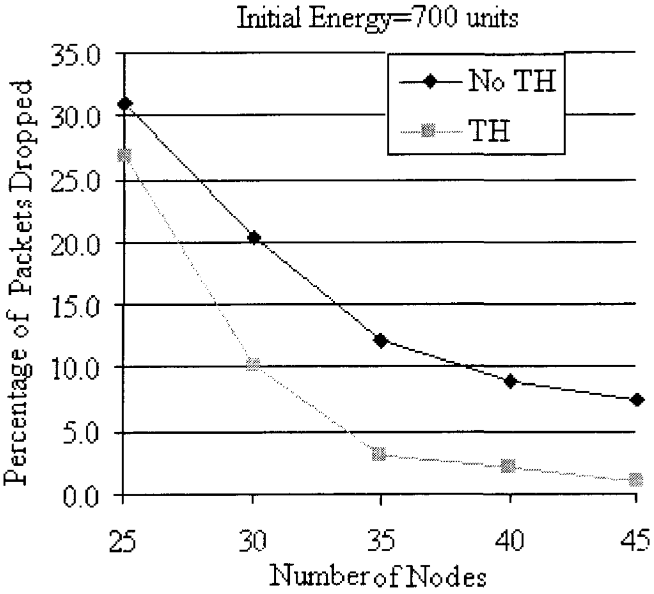


Figure 5.23 Percentage of Data Packets Dropped -700 unit energy

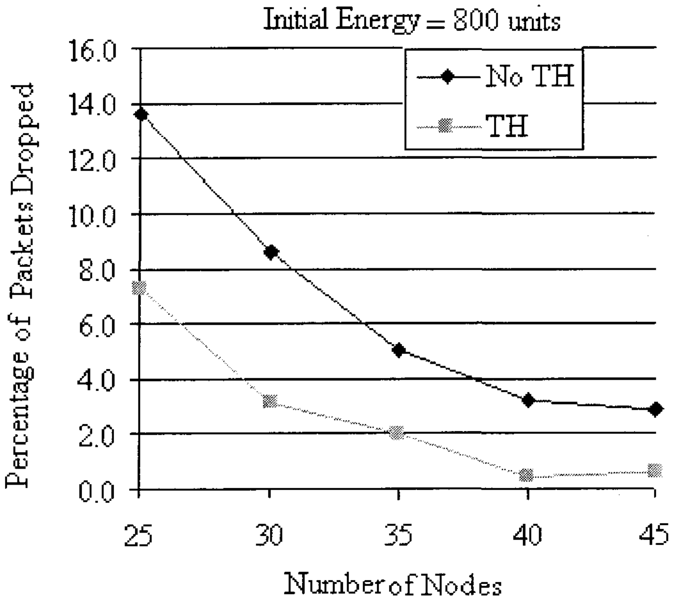


Figure 5.24 Percentage of Data Packets Dropped - 800 unit energy

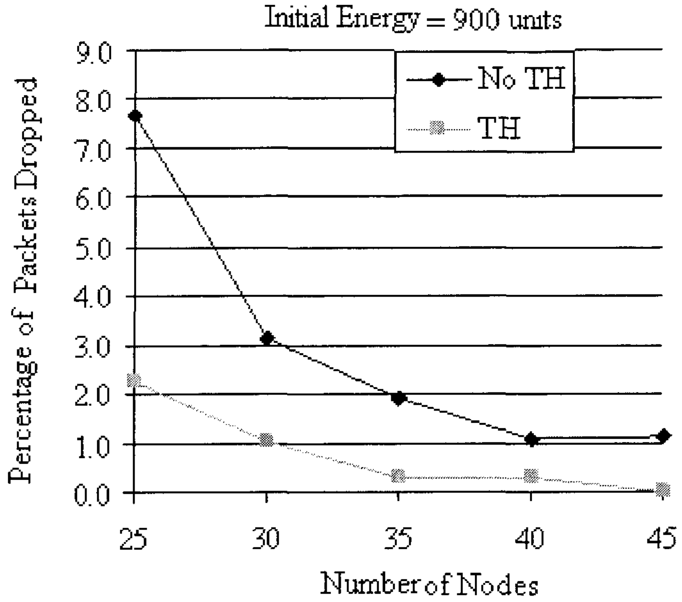


Figure 5.25 Percentage of Data Packets Dropped - 900 unit energy

For smaller number of nodes and small initial battery energy, the percentage of information packets dropped is more (Figures 5.22- 5.23). This is due to the scarcity of sensor nodes participating. In absence of alternate paths, it is more likely that labels will not reach a sink. Therefore, the phenomenon of dropping labels happens more frequently than request and data dropping. However, as the number of nodes increases, the difference between threshold and without threshold becomes significant due to the availability of alternate paths.

5.5 Faults in Sensor Nodes

When a sensor node crashes due to battery exhaustion or other physical event, the primary path breaks down and the re-routing in the network takes place. Most of the protocols discussed in the previous chapters, including IDLF, do not offer specific knowledge regarding the state (faulty or fault-free) of the sensors in the network. We study the impact of faulty nodes on the performance of label dissemination framework. We employ an alternate disjoint path. This alternate path scheme (RM-IDLF) may have a higher path cost in terms of energy consumption. However, under the faulty nodes scenario, it proves to be more reliable in terms of data packet delivery to sink than the single path scheme (IDLF). Additionally, in single path routing, as the path fails, the sink stops receiving the data packets due to absence of a backup route. In multipath routing, we took one alternate path. However, depending on the application and importance of the data delivery, the number of paths can be increased to make the system more robust. We set the faulty nodes of the order of 10%, 20%, 30%, 40% and 50% of the total nodes participating in the simulation. Thus, if 50 sensor nodes are participating within the sensor network and the network has 30% fault, it means that only 35 nodes are actually participating in the network. To study the worst case scenario, a faulty node in the sensor network is assumed to be functional until the label and request communication. It becomes non-functional (fails) before the onset of data transmission.

5.6 Performance Assessment - RM-IDLF

RM-IDLF is resilient to sensor node failure. Thus, if a primary route fails, there is still a likelihood of the information reaching sink because the risk of collapse of an alternate path is lesser as compared to the single path routing. In RM-IDLF, we are more interested in tracking the rate of successful data packets reaching sink. Within the physical grid, we used 70 sensor nodes, each within the communication range of each other. A source node is assumed to be randomly selected and during the simulation, there is only one source node in the sensor field at a time. It is assumed that every node in the network knows its coordinates in the physical field. The label dissemination protocol RM-IDLF intends to disseminate data towards the sink using negotiations. As explained in Chapter 4, we assumed that the size of data packet is 31 times greater than the size of the label and request packet. Then if we assume that transmitting a label or request packet between two neighboring nodes takes one unit time, transmitting a data packet will take 31 unit times. Also, we assumed that transmitting information consumes 3 times more energy per unit time than receiving. An average of 1000 simulation is taken for different set of nodes. We took 20 nodes in a sensor field and increased to 70 nodes with an addition of 10 nodes for various cases. Contrary to 5 nodes in IDLF, In RM-IDLF, we started with 20 nodes because for alternate path, it is appropriate to start with comparatively larger number of nodes in the sensor network. Also, in RM-IDLF, there are no periodic updates to detect the availability of the alternate paths. This makes the sequential accuracy of the multiple paths independent of the frequency of the updates exchanged.

5.6.1 RM-IDLF Evaluation

We will investigate different scenarios: (a) Fault-free single path (b) Fault-free multi-path (c) Single path with faulty nodes and (d) Multi-path with faulty nodes. In this section we will compare the performance of RM-IDLF with IDLF on the basis of (a) Energy consumption over time, (b) Percentage of data packets reaching the sink over time (c) Average time to reach to the sink. A brief comparison of RM-IDLF with flooding and SPIN will be performed at the end. To justify the fair comparison of RM-IDLF with IDLF, we evaluate the latter under the same assumptions as the former.

5.6.1.1. Energy consumed vs. Data packets delivered by entire sensor network over time

In this section, we assess the performance of the RM-IDLF algorithm on the basis of (a) total energy consumed over the simulation time and (b) the percentage of data packets delivered by entire sensor network over time. We study each result under both fault-less and faulty situation. Figure 5.26 shows the performance comparison of both IDLF and RM-IDLF algorithms involving fixed 70 nodes. The simulation time is varied from 50 units to 500 time units. We assumed all the sensor nodes are functioning properly and there is no node failure during the simulation. Under this fault-free scenario, a plot of total consumed energy vs. simulation time shows that 82.85% extra energy (than IDLF) is consumed for the functioning of RM-IDLF. Conversely, figure 5.27 shows the analysis of percentage of data packets delivered at sink over time. Under the parallel simulation settings both IDLF and RM-IDLF has been compared. This fault-free sensor node analysis shows that using RM-IDLF 27.65% of extra data packets is delivered at sink. The reason for using an alternate path is not very strong under the fault free scenario. This can be explained by the fact that the extra energy is consumed for creating a disjoint path. If the possibility of occurrence of fault does not exist, then a single path routing is appropriate for disseminating the information, e.g. low power simple applications. In RM-IDLF, an extra amount of energy is consumed to route a data packet through the alternate path. In addition, a small amount of energy is also consumed to find an alternate path. Furthermore the creation of alternate path is not always guaranteed in RM-IDLF. The reason for the delivery of additional data packets reaching the sink in the latter case can be attributed to the fact that using an alternate path ensures less control delay. The possibility of delivering the data packets increases in alternate path scenario.

Similarly, the performance of RM-IDLF is evaluated by increasing the faulty nodes in the system. Figures (5.28 -5.29) shows that when 10% nodes are faulty RM-IDLF consumes 83.73 % more energy than IDLF, but the percentage of data packets delivered at the sink node increases considerably by 84.55 percent. Figures (5.30 and 5.31) show that for the total of 20% faulty nodes, the percentage of extra energy consumed in RM-IDLF is 73.36 %, whereas the increase in percentage of data packets delivered is 134.29 %. Likewise, for 30% faulty nodes the energy consumed is 68.06% as compared to 240% of increase in data packets delivered, figures (5.32 and 5.33). For 40% faulty nodes, the extra energy consumed is 42.91% as compared to177.33 percent increase in fetching the data packets at sink, figures (5.32 and 5.33). For the occurrence of 50% fault in the system, 26.18% of the extra energy is consumed. The percentage of extra data packets reaching to sink is increased by 93.42%.

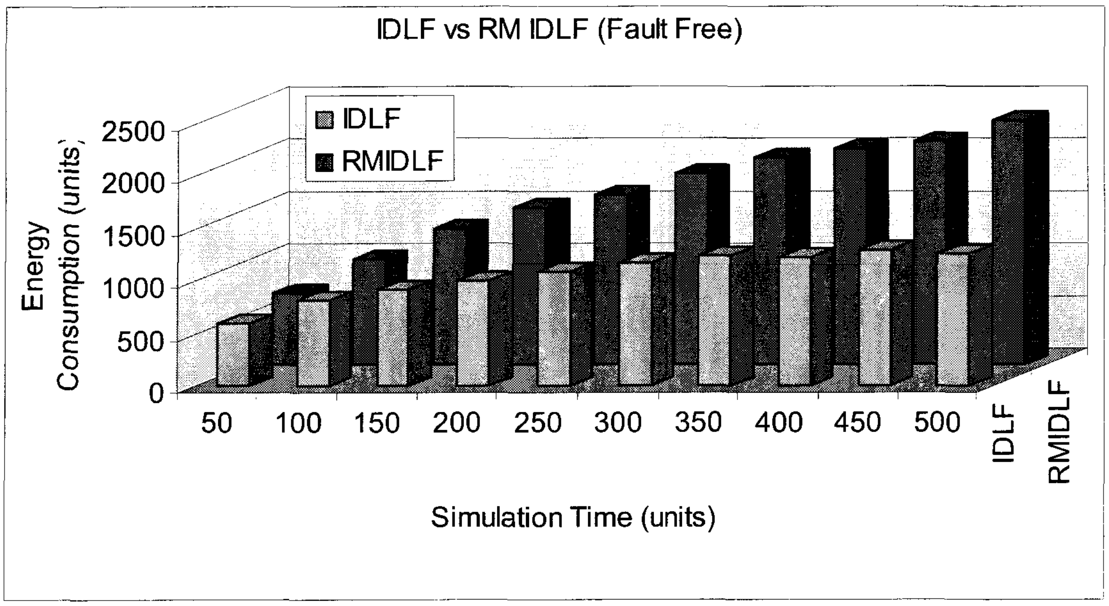


Figure 5.26 Energy Consumed by entire sensor network over time – Fault Free

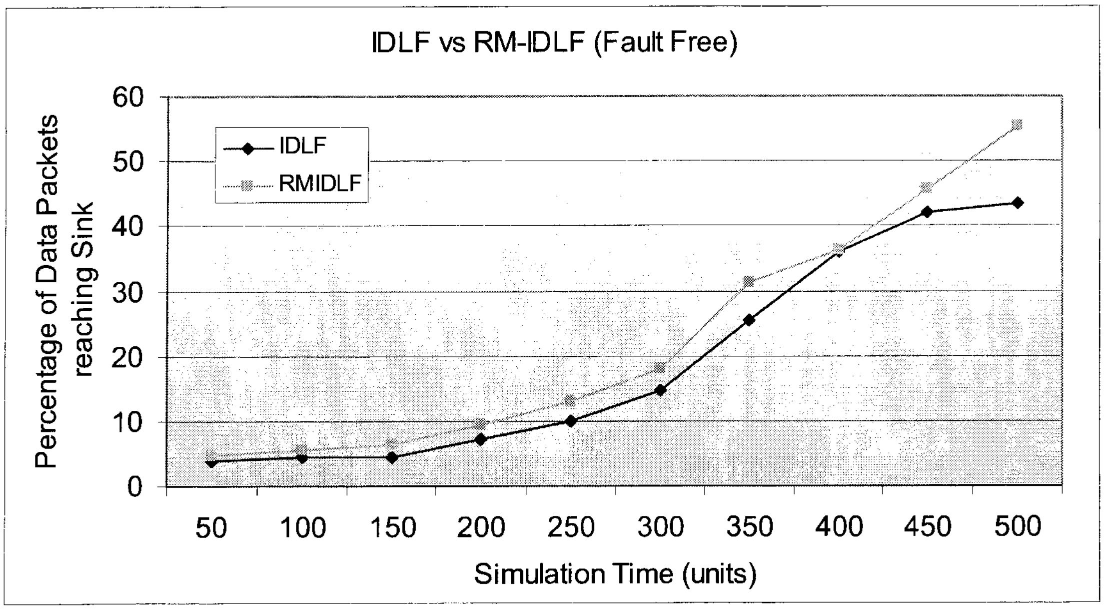


Figure 5.27 Percentage of Data Packets Delivered at Sink over Time- Fault Free

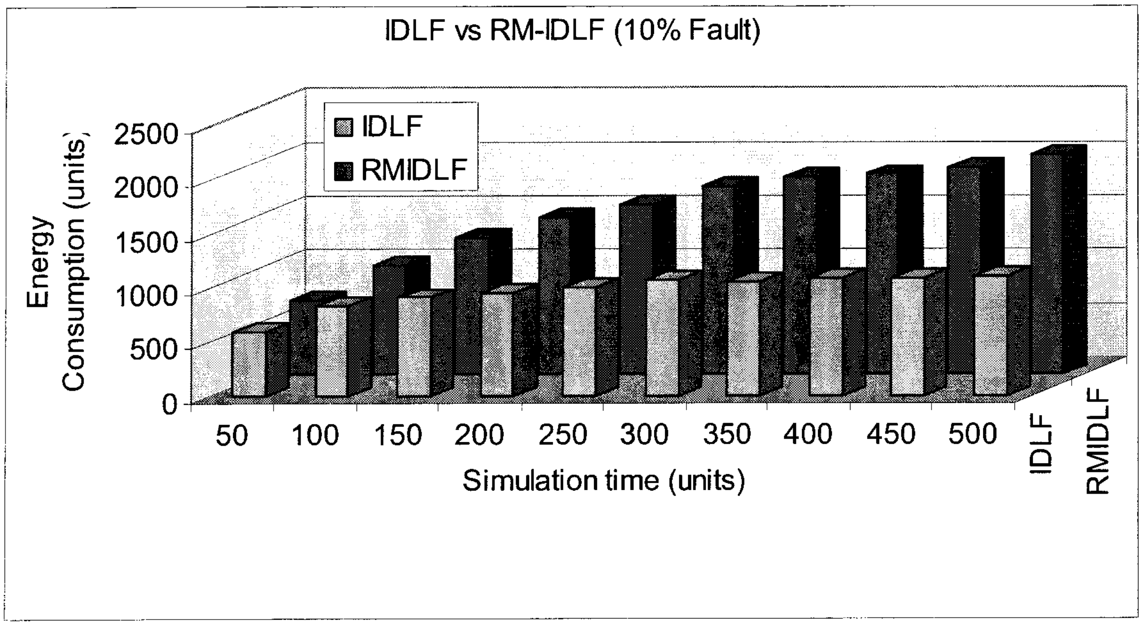


Figure 5.28 Energy Consumed by entire sensor network over time – (10 % Fault)

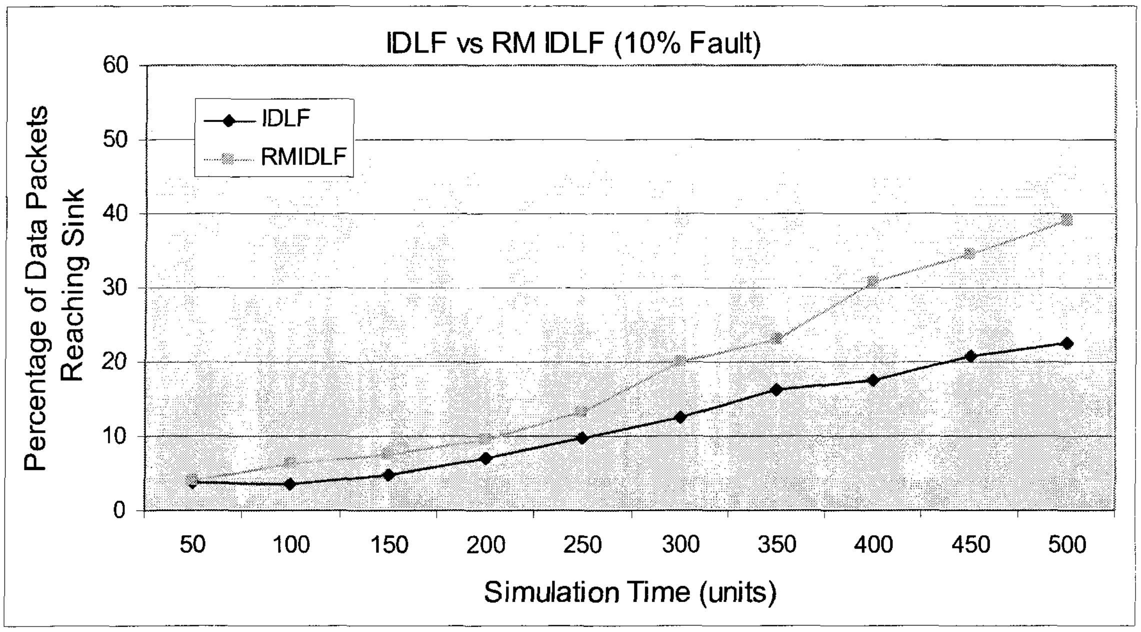


Figure 5.29 Percentage of Data Packets Delivered at Sink over Time– (10 % Fault)

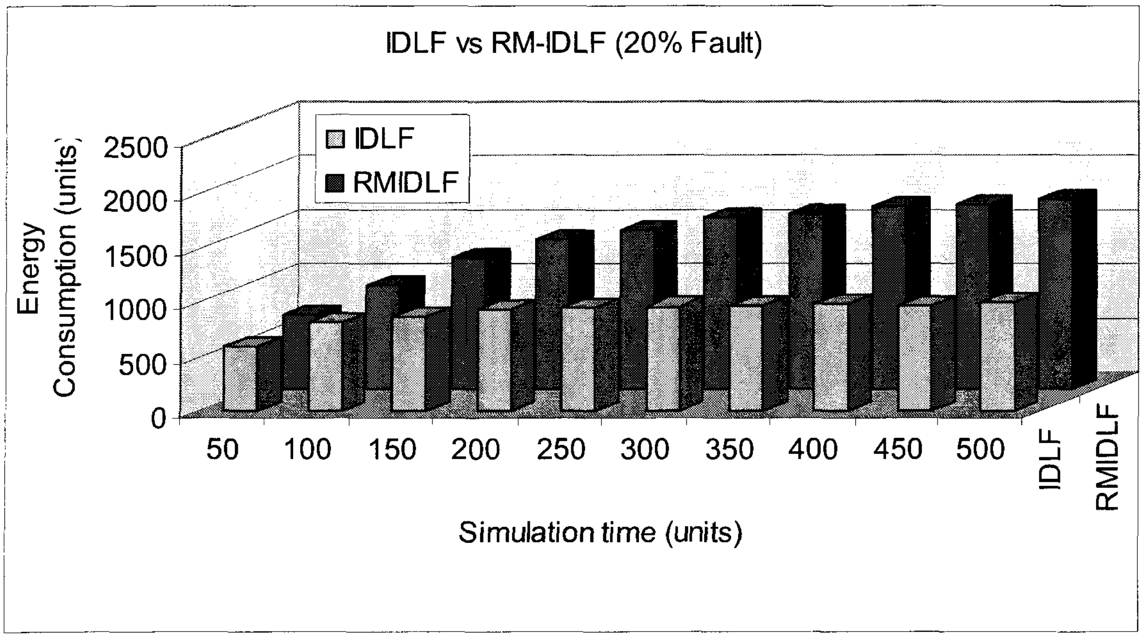


Figure 5.30 Energy Consumed by entire sensor network over time – (20% Fault)

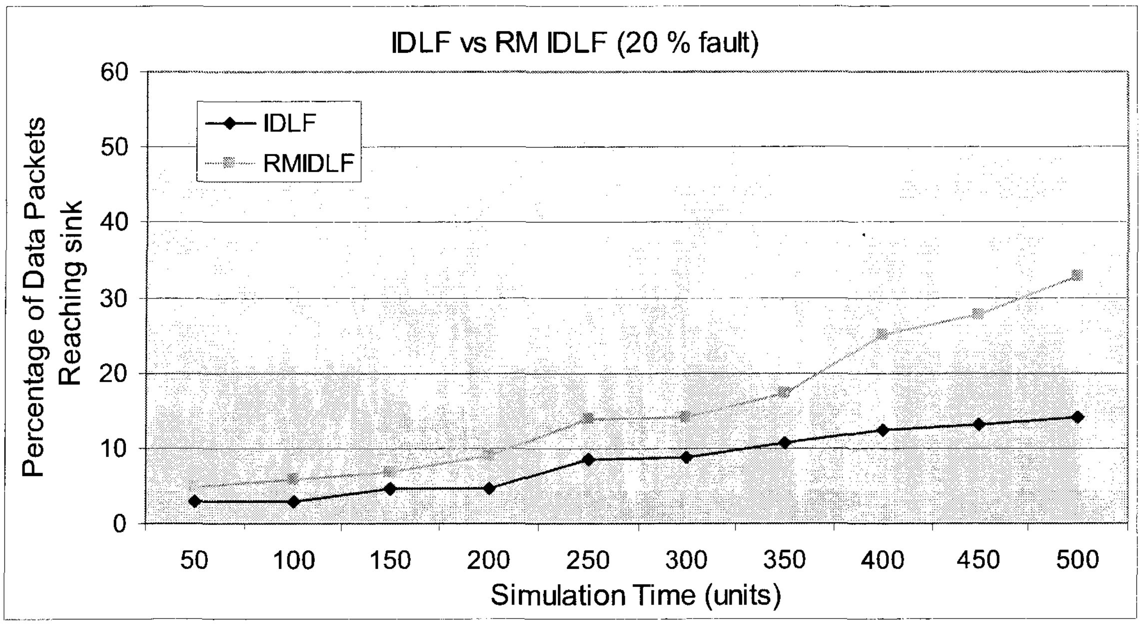


Figure 5.31 Percentage of Data Packets Delivered at Sink over Time– (20% Fault)

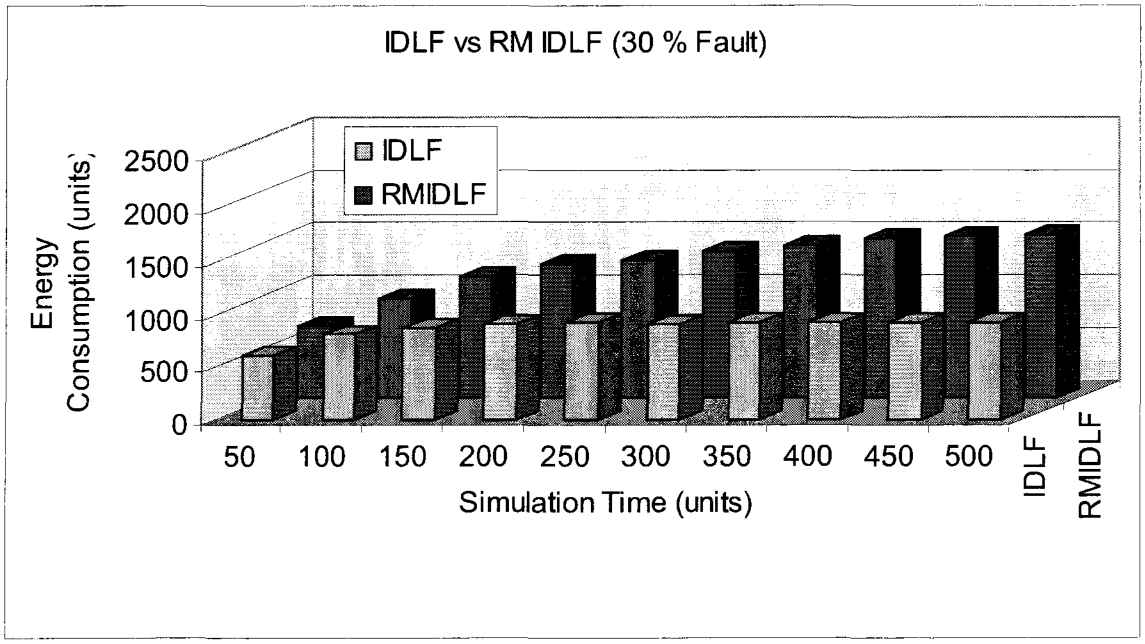


Figure 5.32 Energy Consumed by entire sensor network over time – (30% Fault)

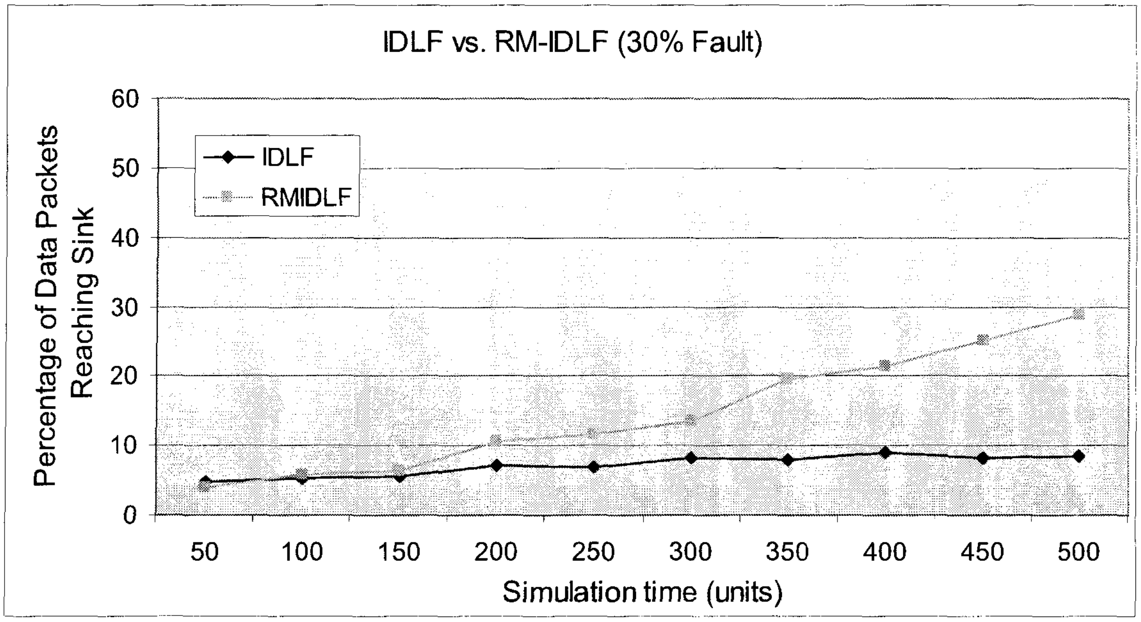


Figure 5.33 Percentage of Data Packets Delivered at Sink over Time – (30% Fault)

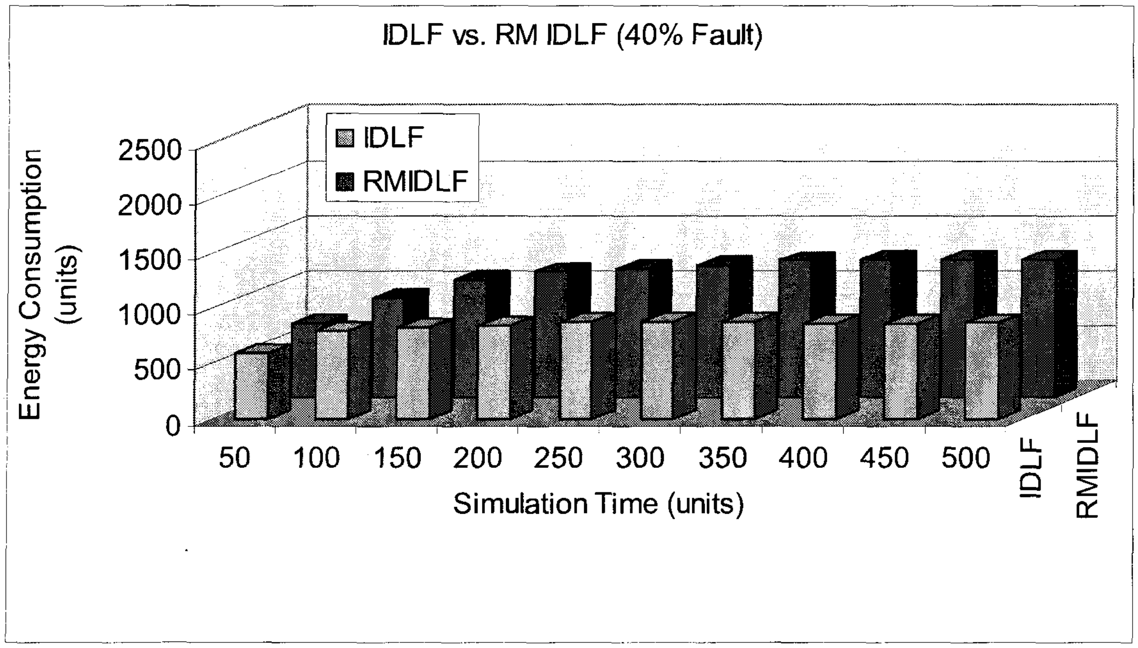


Figure 5.34 Energy Consumed by entire sensor network over time – (40% Fault)

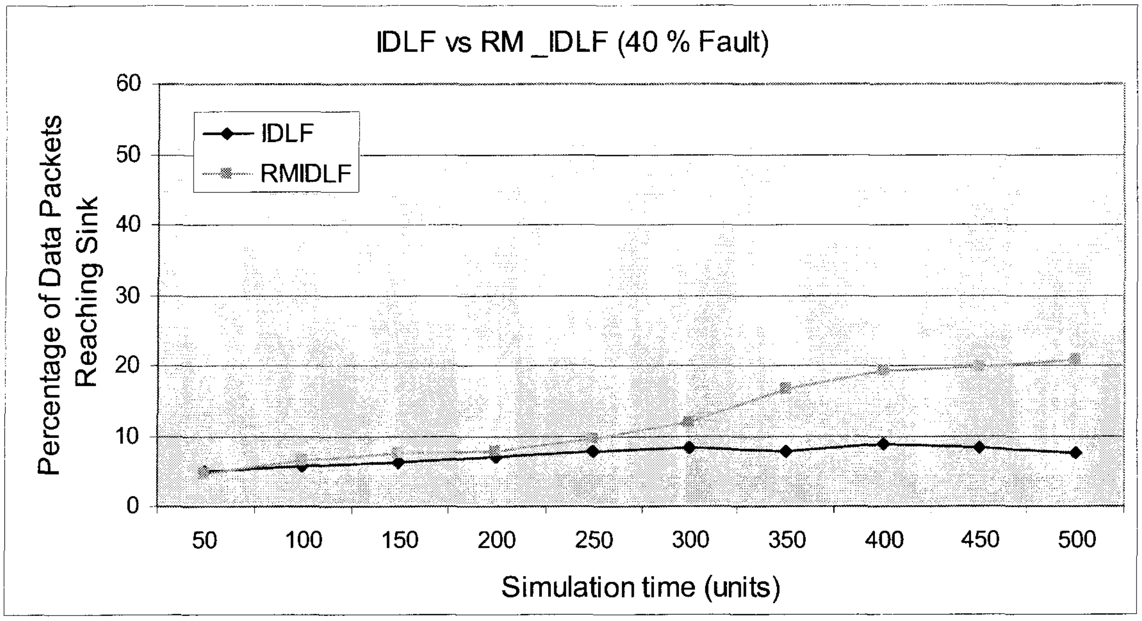


Figure 5.35 Percentage of Data Packets Delivered at Sink over Time – (40% Fault)

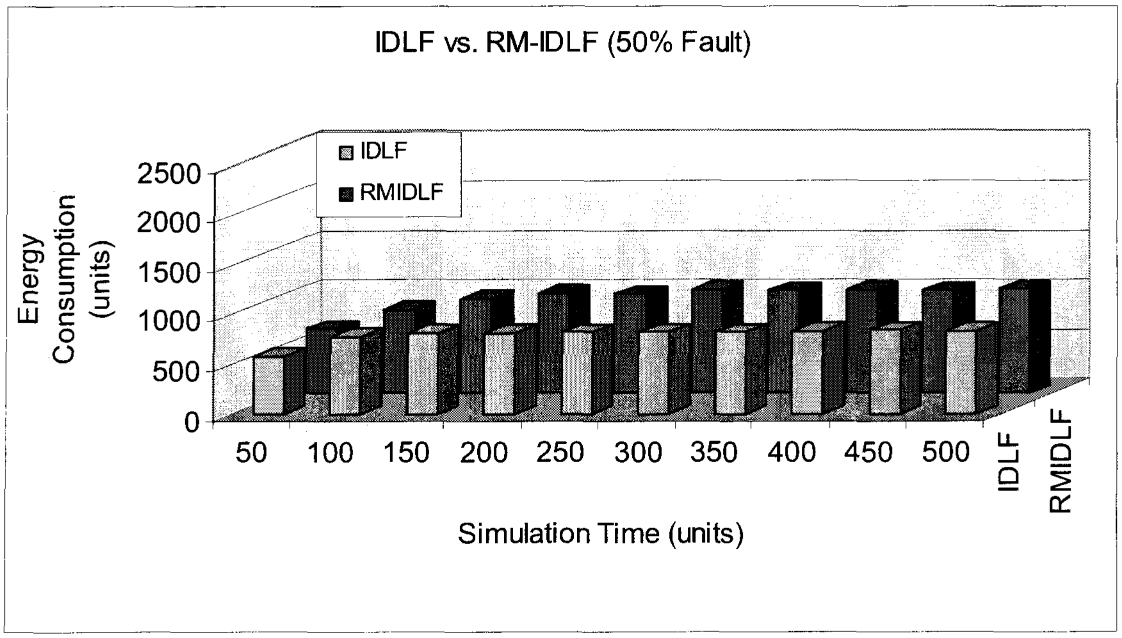


Figure 5.36 Energy Consumed by entire sensor network over time – (50% Fault)

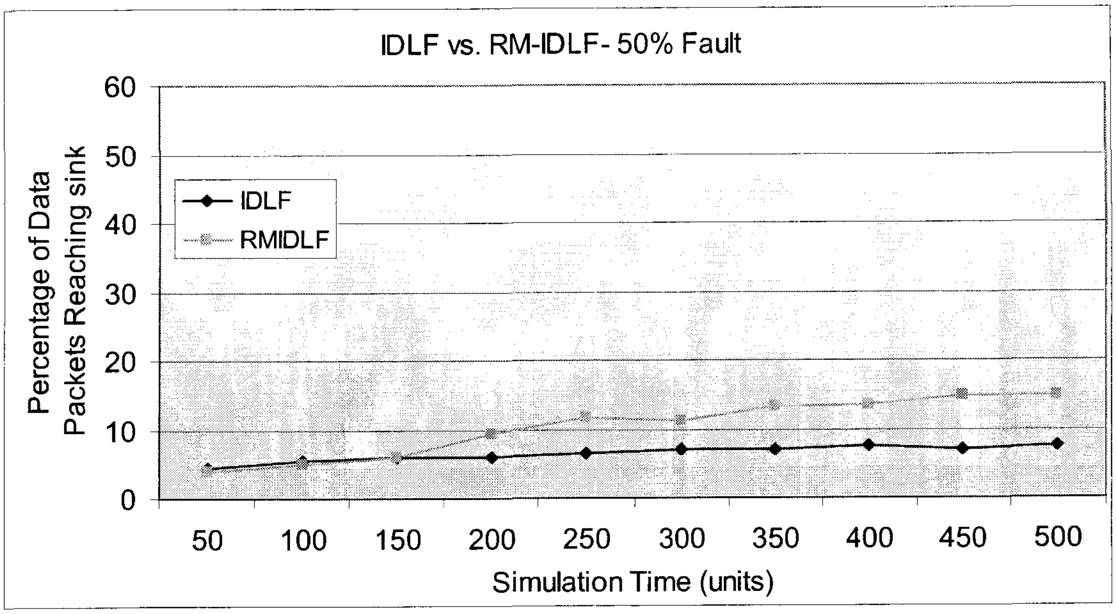


Figure 5.37 Percentage of Data Packets Delivered at Sink over Time– (50% Fault)

Table 5.3 Percentage of Extra Energy Consumed in RM-IDLF

in comparison with IDLF

TIME (units)	Fault Free	10% Fault	20% Fault	30% Fault	40% Fault	50% Fault
50	11.84	12.15	11.58	11.71	11.04	11.69
100	21.49	20.54	16.09	15.95	10.82	8.74
150	41.08	37.83	36.45	31.56	26.77	15.06
200	49.38	51.81	46.40	40.56	32.91	21.84
250	48.04	55.88	51.68	40.93	31.98	19.81
300	55.08	62.16	63.78	52.75	36.40	25.24
350	57.58	71.58	63.66	57.67	41.06	24.23
400	68.31	69.97	68.05	62.97	43.33	25.13
450	63.92	75.30	73.59	65.64	44.01	22.17
500	82.85	83.73	73.36	68.06	42.91	26.18

Table 5.4 Percentage of Extra Data Packets delivered in RM-IDLF

in comparison with IDLF

TIME (units)	Fault Free	10% Fault	20% Fault	30% Fault	40% Fault	50% Fault
50	17.95	5.41	63.33	14.89	2.04	13.64
100	27.91	80.00	93.33	13.46	15.79	9.09
150	46.51	57.45	53.33	14.55	20.63	1.64
200	30.14	36.23	93.62	47.89	11.27	57.63
250	30.30	34.69	63.53	68.12	23.38	81.25
300	23.97	60.48	63.22	62.65	46.34	62.32
350	22.83	42.59	63.21	143.75	115.58	88.57
400	1.11	76.44	99.20	134.07	115.73	81.33
450	8.81	66.67	112.31	204.88	138.55	114.49
500	27.65	84.55	134.29	240.00	177.33	93.42

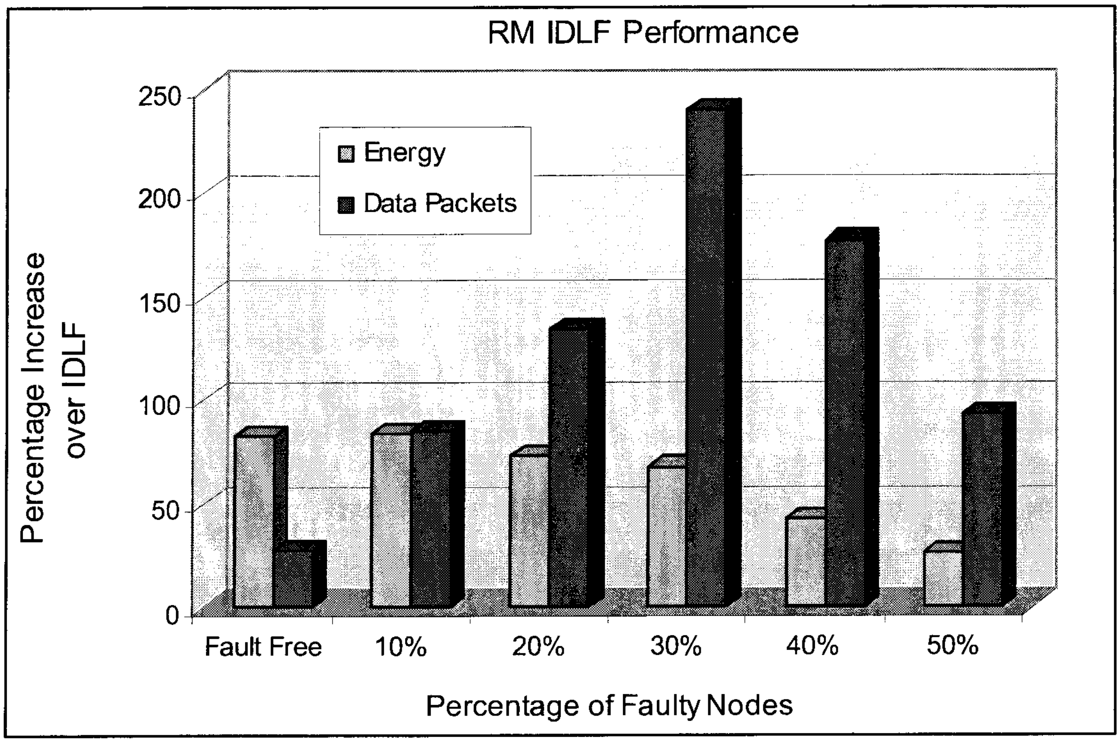


Figure 5.38 Cost Performance Graph –RM-IDLF

Table 5.3 below illustrate the percentage of extra energy spend in RM-IDLF. Simulation result shows that, as the faulty nodes in the network increases the energy consumption in the network decreases. The experimental results illustrate that when the number of sensor nodes in a network is less, the cost of disseminating the data packets is not high. In Table 5.4, the percentage of additional data packets delivered is studied. As the faulty nodes in the network are increased, the performance of RM-IDLF gets better in terms of percentage of data packets delivered. This behavior is expected from RM-IDLF in a way that, in the advent of failure of the primary path, an alternate path takes over. As the faulty nodes increases, single paths tear down and the network has to re-initiate the primary path. Figure 5.38 shows a cost performance graph showing the performance of RM-IDLF over IDLF in terms of energy consumed and the data packets reaching the sink.

5.6.1.2. Energy consumed vs. Number of Participating Sensor Nodes

In Section 5.6.1.1, we analyzed the results for energy consumed for the fixed number of sensor nodes. In this section, we alter the number of sensor nodes participating in the network and study each result under both fault-less and faulty situation. The simulation time is kept constant at 500 time units. Table 5.5 shows the average of 1000 simulation for both RM-IDLF and IDLF. The experimental results explain that it takes more energy to disseminate the data packets when the number of nodes increases in the sensor network. For fewer nodes, due to unavailability alternate paths, it is more likely that labels will not reach a sink. Therefore, the phenomenon of dropping labels happens more frequently than request and data dropping. For 20 nodes, under the fault free condition (figure 5.39), it takes 935.46 units of energy for RM-IDLF as compared to 2319.00 units of energy for 70 nodes. In case of fewer nodes in the network, the chances of data packets not reaching sink is high due to unavailability of the neighboring nodes. This causes less consumption of energy due to dropping of data packets. Secondly, as the number of nodes increase, the average hop distance from source to sink also increase and the possibility of creation of an alternate path increases as well. Thus, the network expends more energy. In presence of faulty nodes, a descending pattern for energy consumption is observed, figures 5.40- 5.44. Occurrence of failure creates a void among the participating nodes, thereby abandoning the data packets and eventually consuming less energy. More the number of faulty nodes exists in the network, the less is the energy consumed.

Table 5.5 Energy Consumed in RM-IDLF in comparison with IDLF by varying the number of sensor nodes

Number of Nodes	Fault Free		10%		20%	
	IDLF	RM-IDLF	IDLF	RM-IDLF	IDLF	RM-IDLF
20	659.73	935.46	561.29	783.72	469.20	634.42
30	929.93	1463.38	775.96	1188.33	657.43	1003.23
40	1105.06	1840.37	909.48	1563.09	793.75	1269.54
50	1201.95	2026.02	1007.43	1744.57	891.62	1480.85
60	1237.00	2139.52	1040.98	1958.73	945.55	1623.04
70	1268.23	2319.00	1104.22	2028.77	1008.43	1748.22
Number of Nodes	30%		40%		50%	
	IDLF	RM-IDLF	IDLF	RM-IDLF	IDLF	RM-IDLF
20	407.48	552.32	368.17	465.79	328.06	389.78
30	580.61	819.72	527.54	671.95	471.14	576.20
40	713.97	1045.18	643.64	862.75	592.67	722.45
50	807.43	1190.57	729.45	981.62	692.23	813.70
60	860.37	1387.23	791.80	1146.89	762.78	952.12
70	914.92	1537.63	870.10	1243.44	833.09	1051.19

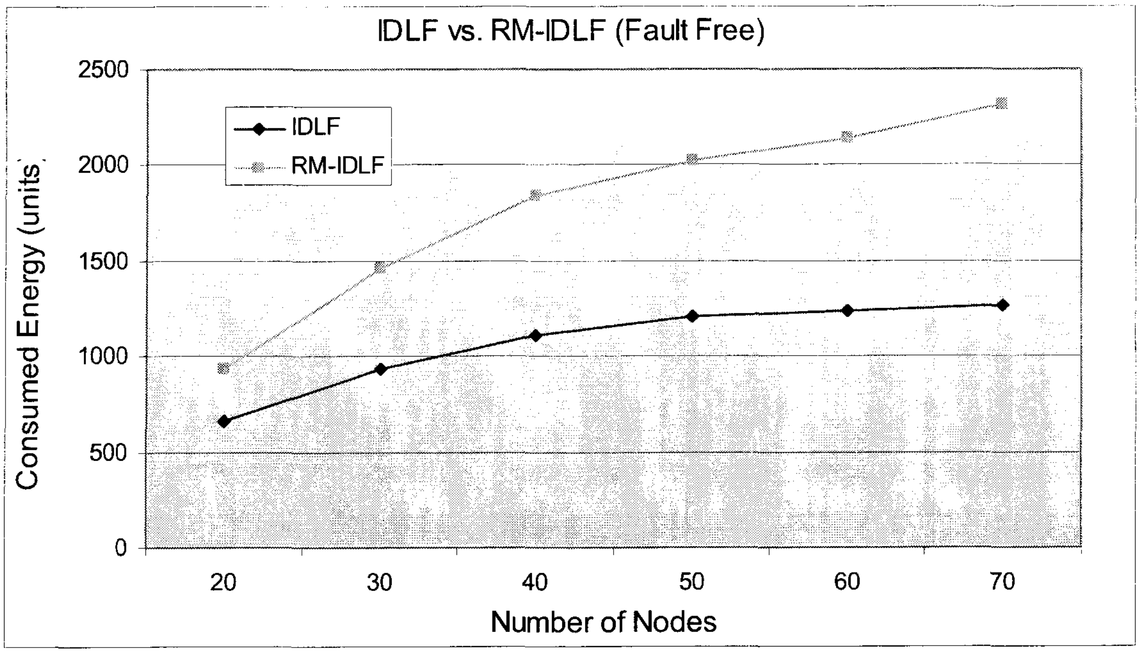


Figure 5.39 Consumed Energy vs. Number of nodes – (Fault Free)

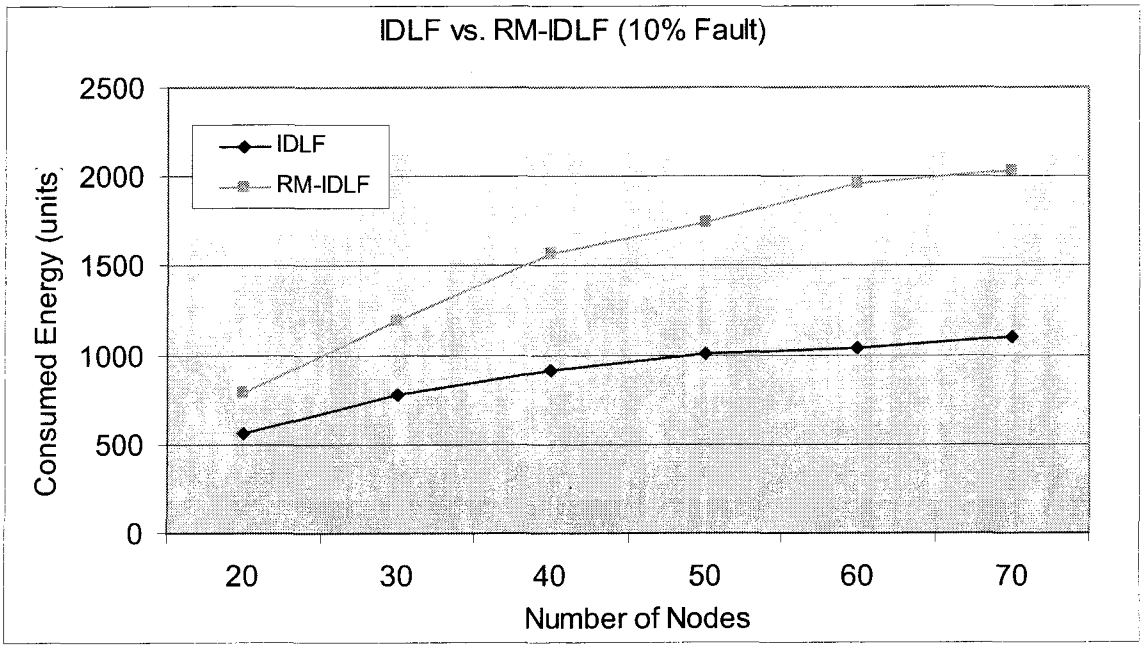


Figure 5.40 Consumed Energy vs. Number of nodes – (10% Fault)

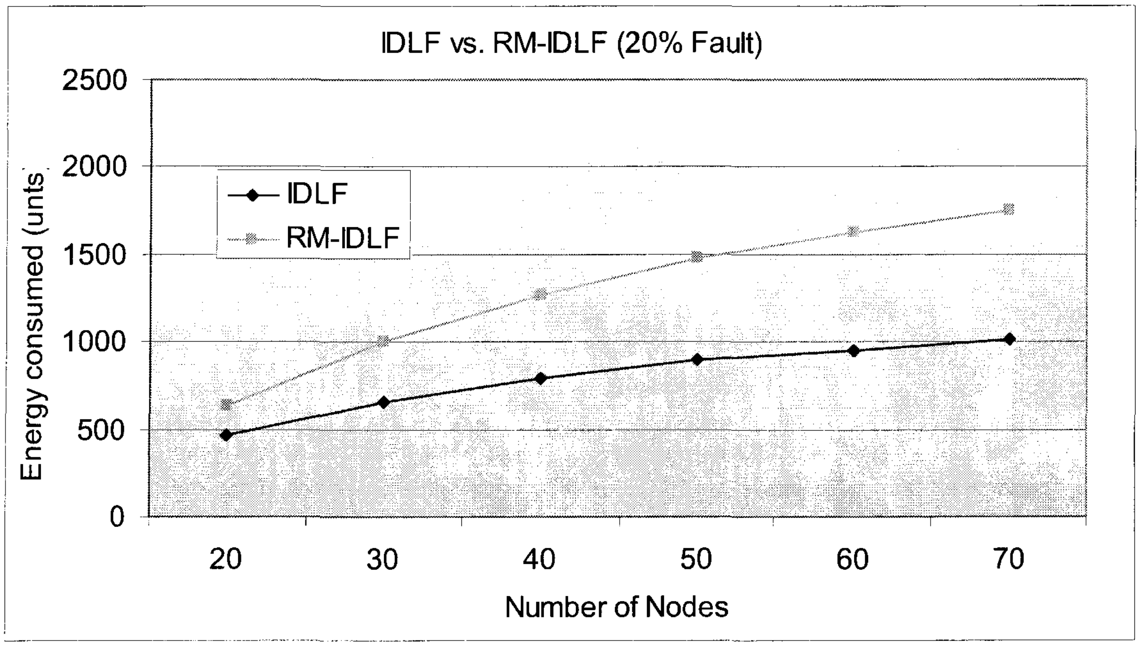


Figure 5.41 Consumed Energy vs. Number of nodes – (20% Fault)

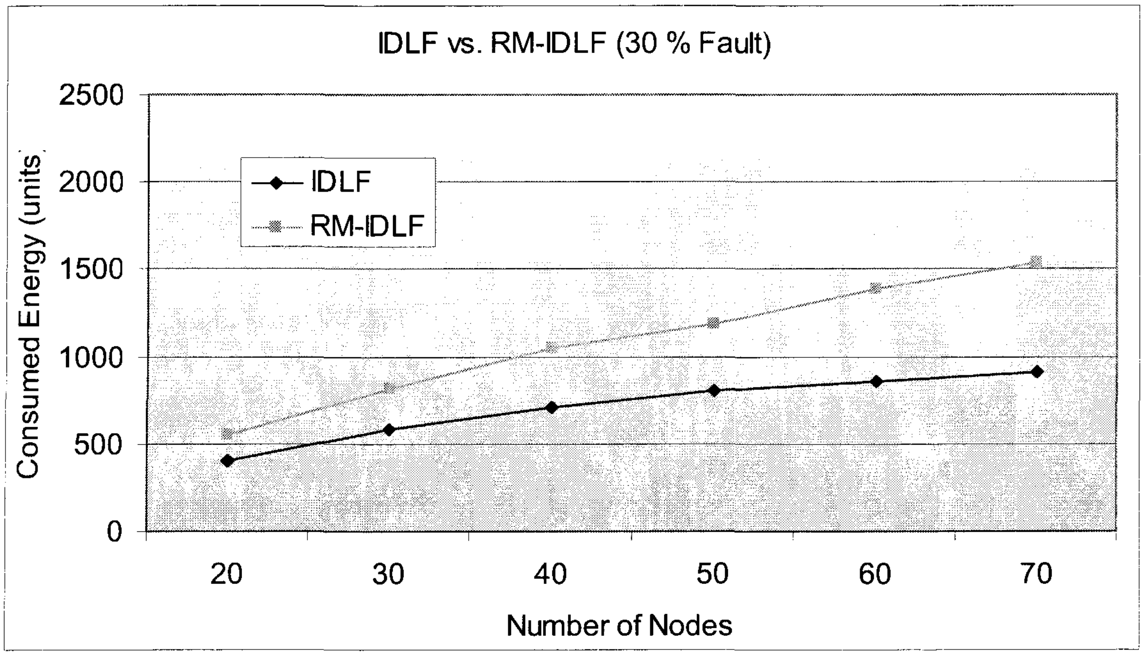


Figure 5.42 Consumed Energy vs. Number of nodes – (30% Fault)

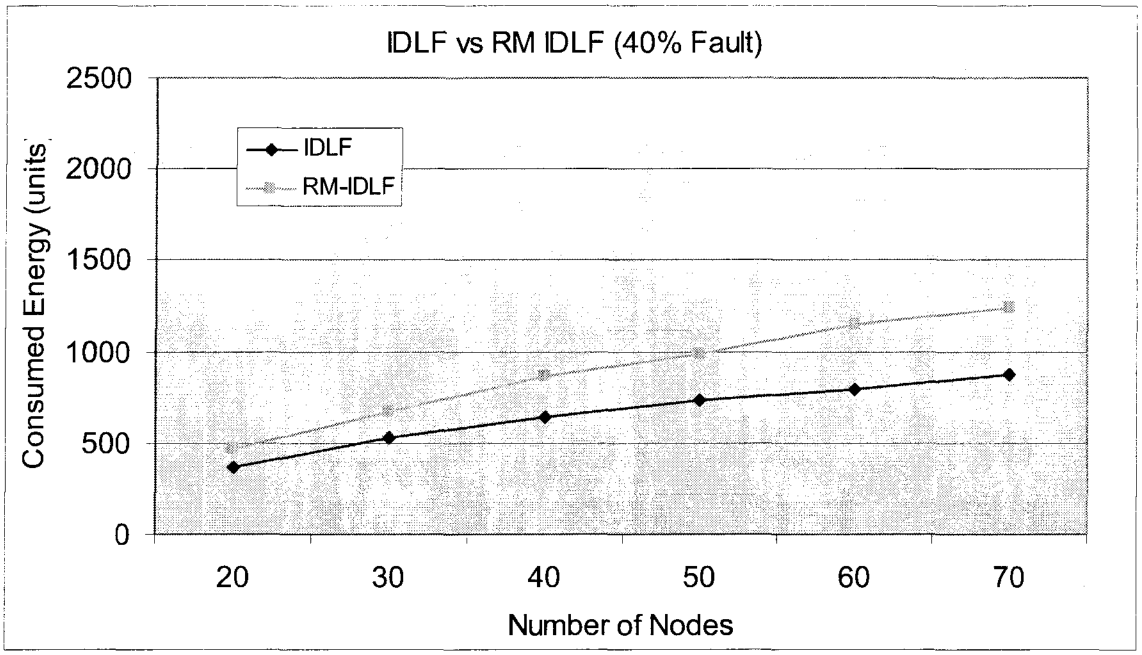


Figure 5.43 Consumed Energy vs. Number of nodes – (40% Fault)

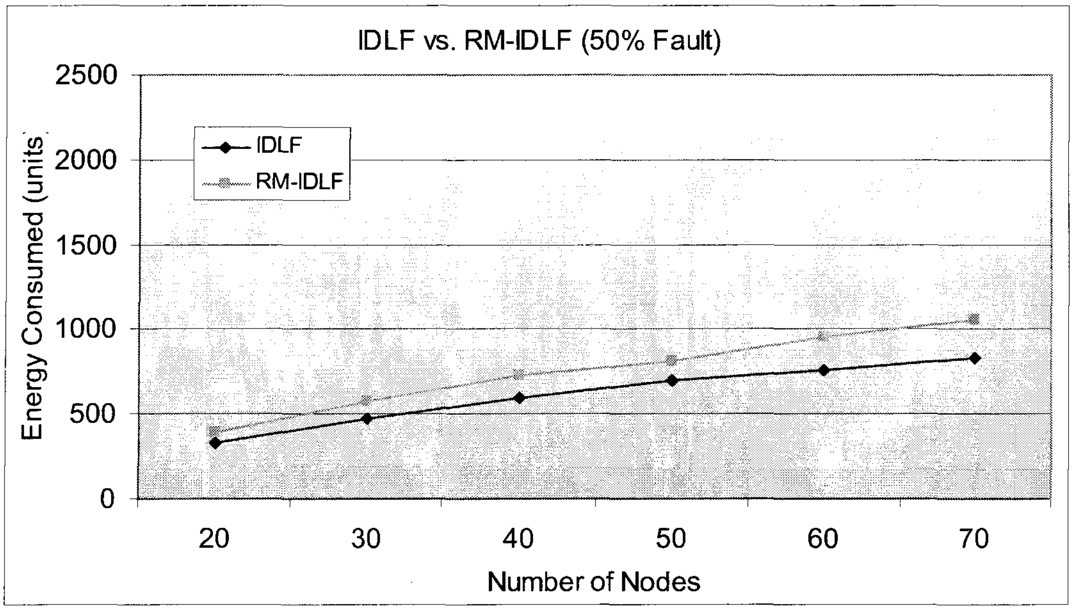


Figure 5.44 Consumed Energy vs. Number of nodes – (50% Fault)

5.6.1.3. Data packets delivered vs. Number of participating Sensor Nodes

Figure (5.45 – 5.50) shows the number of data packets arrived at a sink for a given simulation interval. We then vary the number of nodes and analyze the performance of each. For a given time interval, the number of data packets delivered to sink diminishes when the number of nodes in a network increases. This is because the average distance from a source to a sink increases, so it takes more time to deliver a data packet. In presence of faulty nodes, (Table 5.6) a descending pattern for data packet delivery is observed. Nonetheless, in RM-IDLF more data packets are delivered to sink. Occurrence of failure creates a void among the participating nodes, it takes more time for each data packet to reach to sink and consequently, for a given time, the number of data packets delivered is reduced.

Table 5.6 Percentage of data packets delivered in RM-IDLF in comparison with IDLF by varying the number of sensor nodes

Number of Nodes	Fault Free		10%		20%	
	IDLF	RM-IDLF	IDLF	RM-IDLF	IDLF	RM-IDLF
20	100	100	71.7	82.5	54.3	66.1
30	98.4	98.9	64.5	81.5	44.1	66.7
40	87.3	91.2	52.4	72.9	37	58.5
50	73.2	77.6	41.9	60.9	27.6	48.9
60	56.3	68.1	30.2	51.9	16	41.4
70	43.4	55.4	22.4	39.1	14	32.8
Number of Nodes	30%		40%		50%	
	IDLF	RM-IDLF	IDLF	RM-IDLF	IDLF	RM-IDLF
20	41.7	56.4	31.2	41.7	25.3	34.8
30	32.5	49.9	24.9	37.5	17	27
40	27.2	47	18.1	34.1	14.4	23.1
50	19.1	39.3	15.1	29	11.4	19.1
60	14.7	34.3	11	24.2	7.9	16.3
70	8.5	28.9	7.5	20.8	7.6	14.7

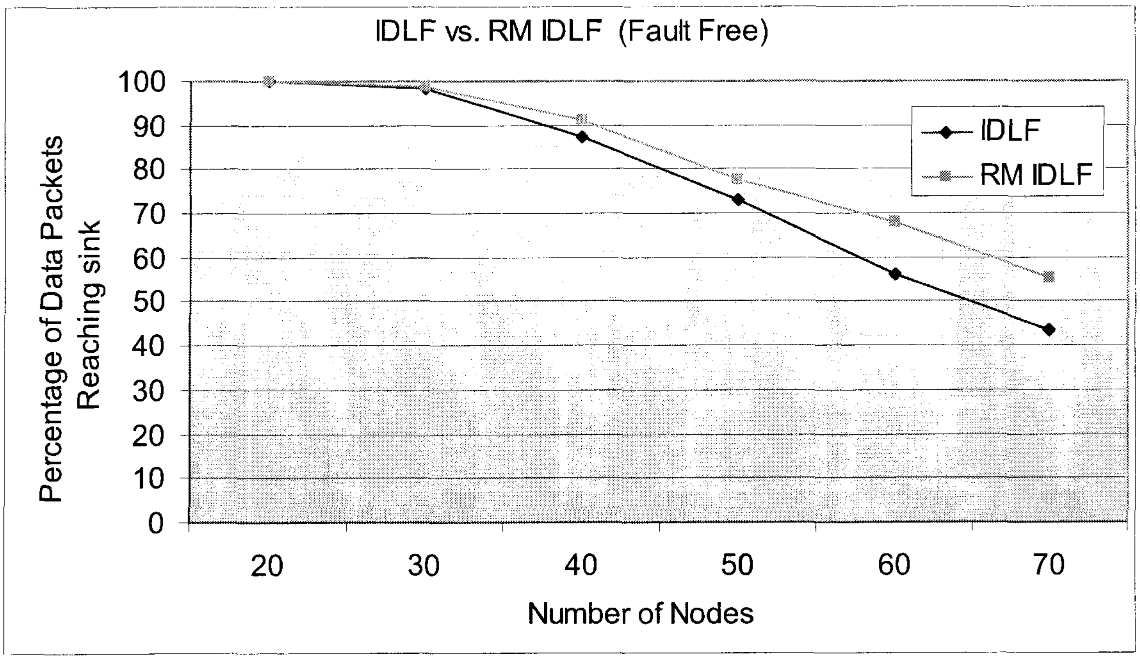


Figure 5.45 Percentage of Data Packets Reaching Sink vs. Number of nodes

(Fault free)

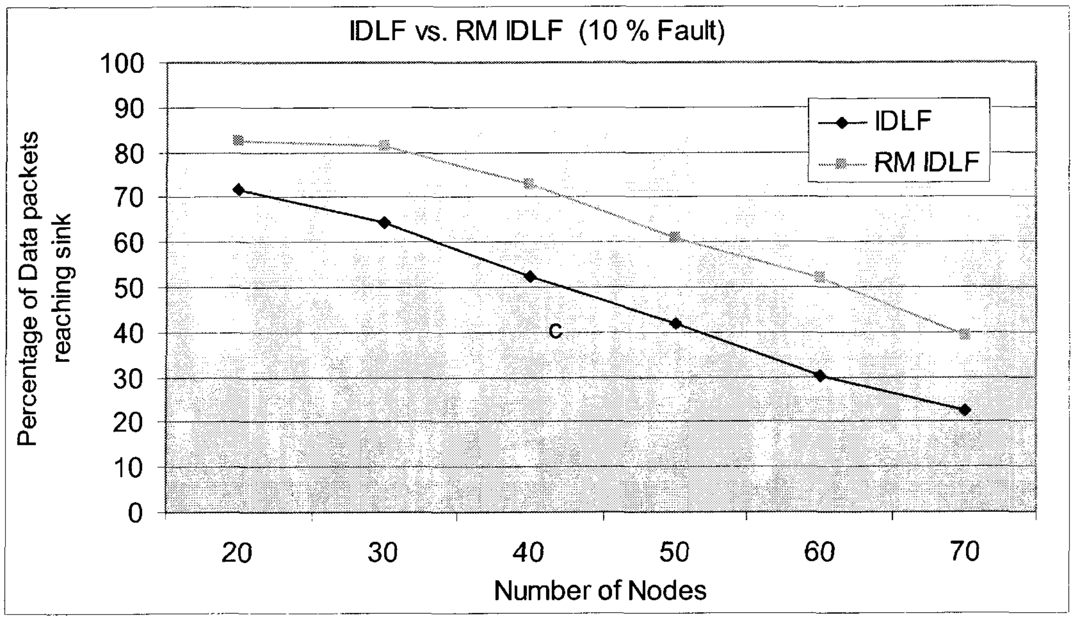


Figure 5.46 Percentage of Data Packets Reaching Sink vs. Number of nodes

(10% Fault)

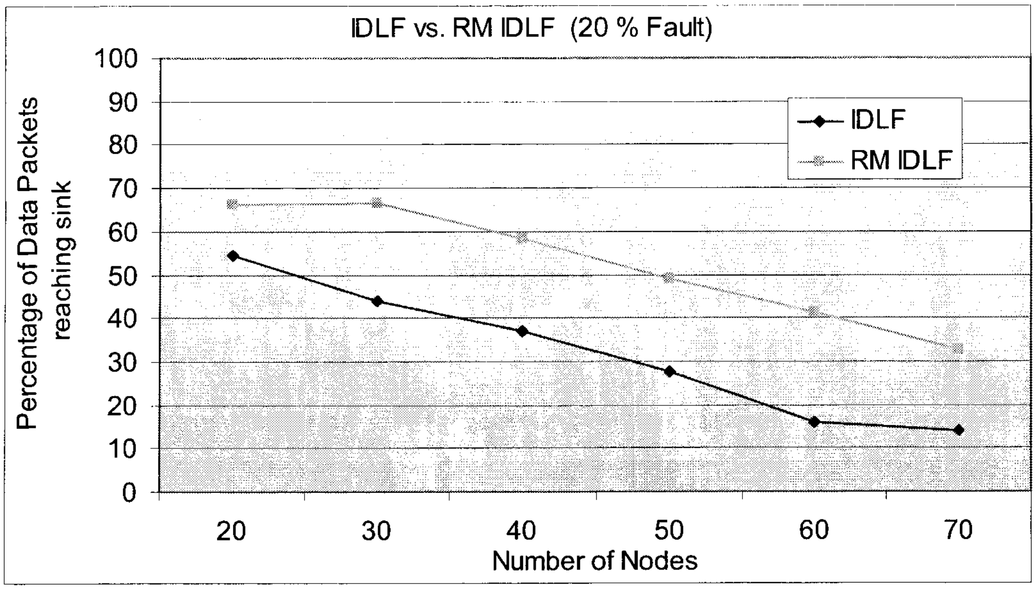


Figure 5.47 Percentage of Data Packets Reaching Sink vs. Number of nodes

(20% Fault)

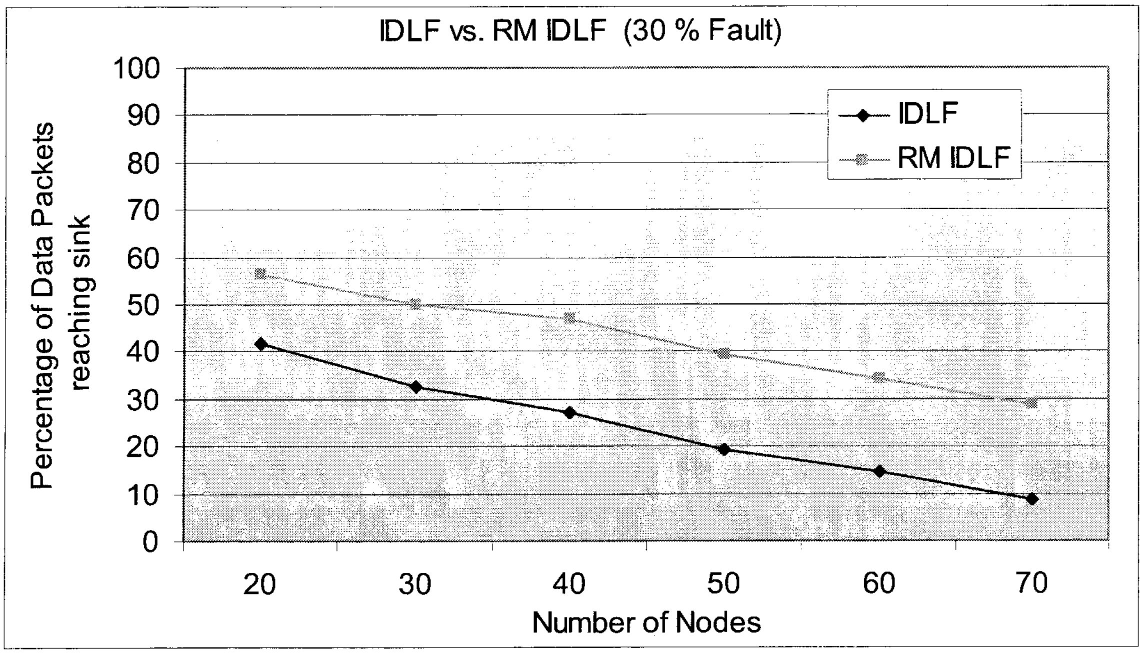


Figure 5.48 Percentage of Data Packets Reaching Sink vs. Number of nodes

(30% Fault)

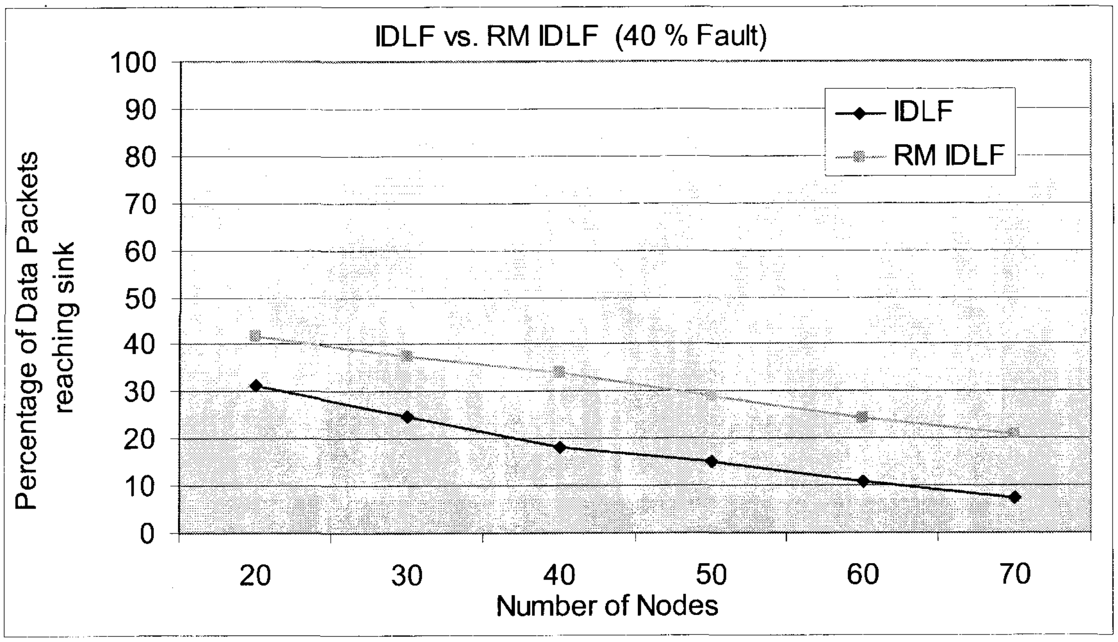


Figure 5.49 Percentage of Data Packets Reaching Sink vs. Number of nodes

(40% Fault)

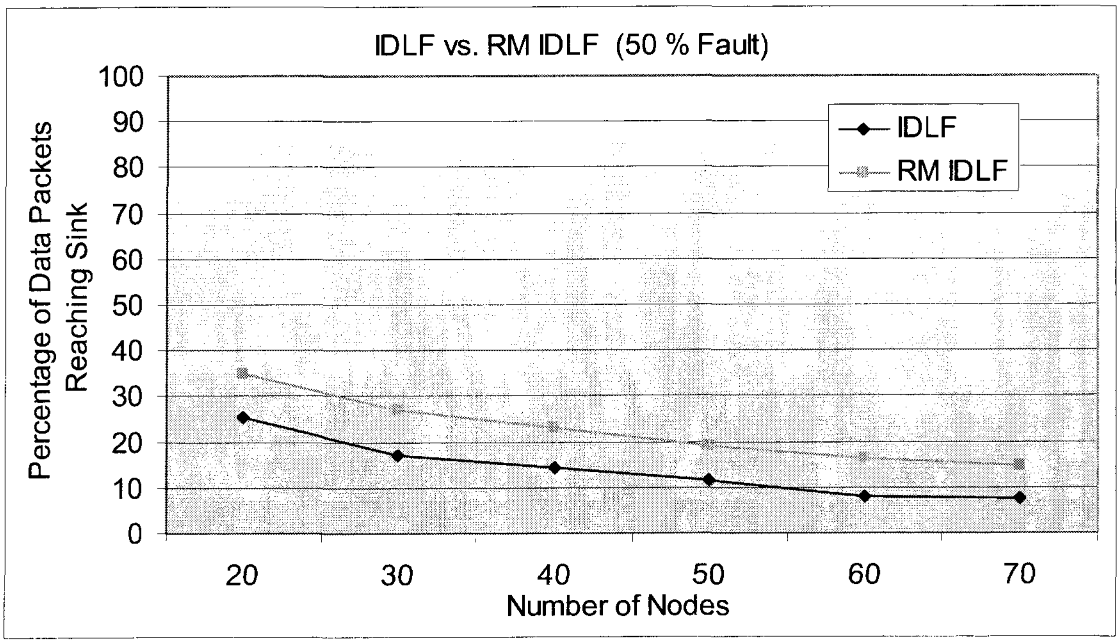


Figure 5.50 Percentage of Data Packets Reaching Sink vs. Number of nodes

(50% Fault)

5.6.1.4. Time to reach sink vs. Number of participating Sensor nodes

Table 5.7 shows the average of 1000 simulation for both RM-IDLF and IDLF. The experimental results explain that data packets take more time to traverse to sink when the number of nodes increases in the sensor network. For fewer nodes, due to unavailability alternate paths, it is more likely that labels will not reach a sink. Therefore, the phenomenon of dropping labels happens more frequently than request and data dropping. For 20 nodes, under the fault free condition (figure 5.51), it takes 157.57 time for RM-IDLF to reach sink as compared to 315.15 units for 70 nodes.

In case of fewer nodes in the network, the chances of data packets not reaching sink is high due to unavailability of the neighboring nodes. This causes less time for the data packets to reach the sink. As the number of nodes increase, the average hop distance from source to sink also increase and the possibility of creation of an alternate path increases as well. Thus, the network takes more time to deliver the data packet. In presence of faulty nodes, a descending pattern for energy consumption is observed, figures 5.52- 5.56. Occurrence of failure creates a void among the participating nodes, thereby abandoning the data packets and eventually tearing down the path. More the number of faulty nodes exists in the network, the less is the time taken for the data packets to reach sink.

Table 5.7 Average time to reach the sink in RM-IDLF in comparison with IDLF by varying the number of sensor nodes

Number of Nodes	Fault Free		10%		20%	
	IDLF	RM-IDLF	IDLF	RM-IDLF	IDLF	RM-IDLF
20	149.26	157.57	130.56	153.41	111.29	137.83
30	192.13	197.72	166.07	199.51	135.32	190.19
40	229.33	235.47	190.28	238.24	149.97	226.81
50	264.21	263.64	228.12	269.41	174.87	268.93
60	302.41	292.82	235.28	289.08	199.56	268.40
70	322.26	315.15	278.68	291.29	209.44	283.14
Number of Nodes	30%		40%		50%	
	IDLF	RM-IDLF	IDLF	RM-IDLF	IDLF	RM-IDLF
20	91.39	131.13	80.15	113.97	62.54	90.44
30	108.30	176.48	94.79	149.22	70.12	126.04
40	124.70	214.90	100.60	185.98	72.73	151.27
50	133.38	230.17	87.13	194.99	73.75	147.80
60	128.79	250.45	91.93	243.93	72.10	192.36
70	135.33	272.55	83.33	243.10	63.45	191.70

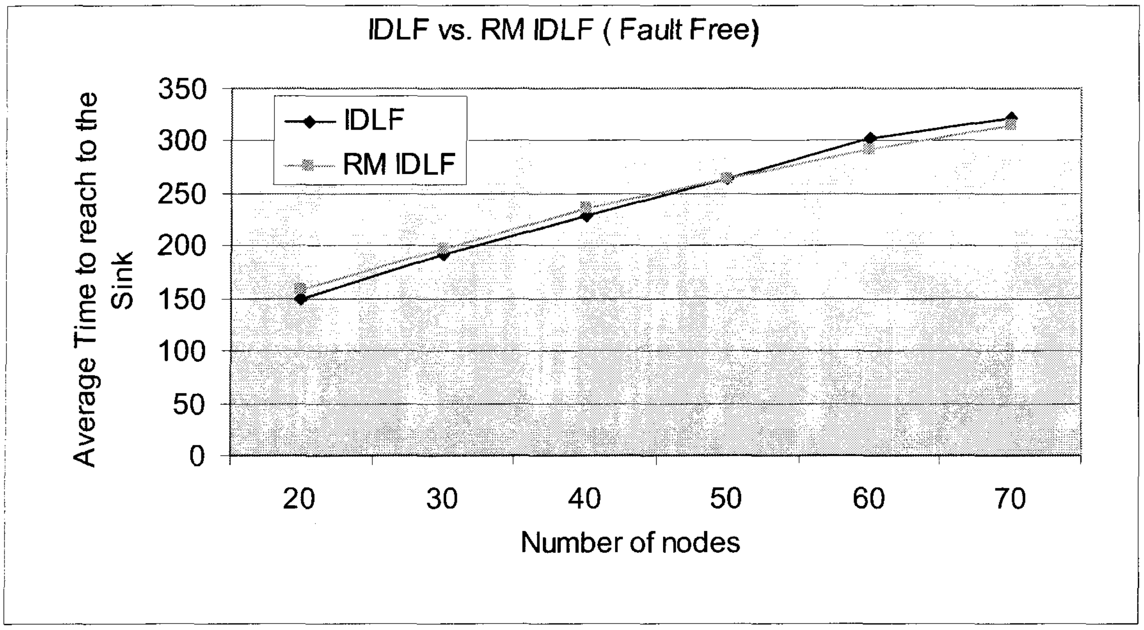


Figure 5.51 Average time to reach to sink vs. number of nodes (Fault free)

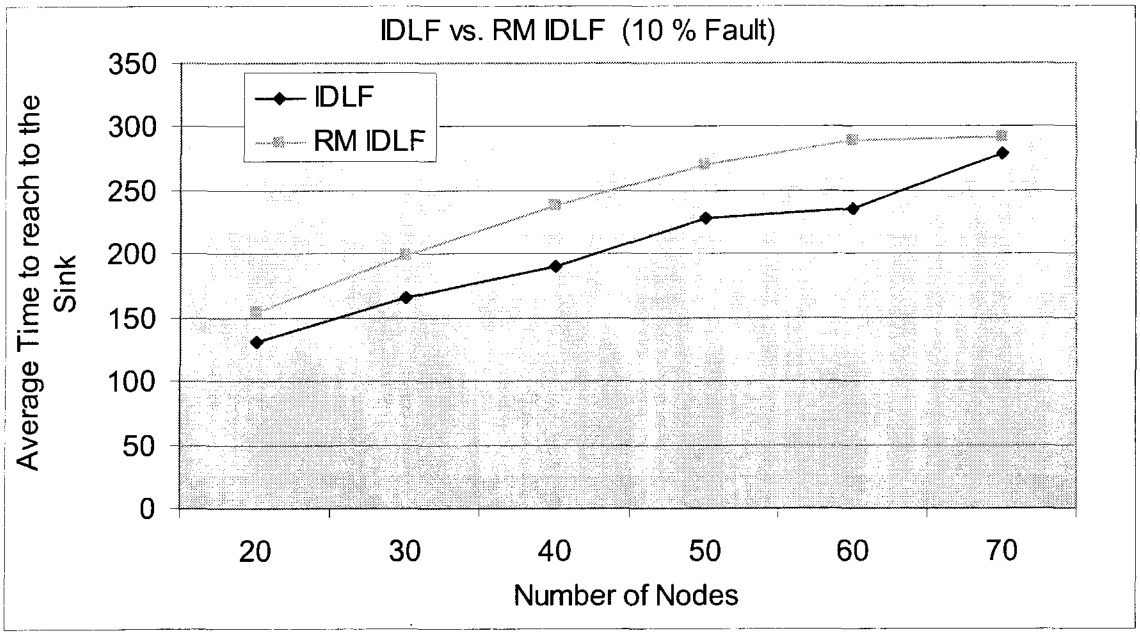


Figure 5.52 Average time to reach to sink vs. number of nodes – (10% Fault)

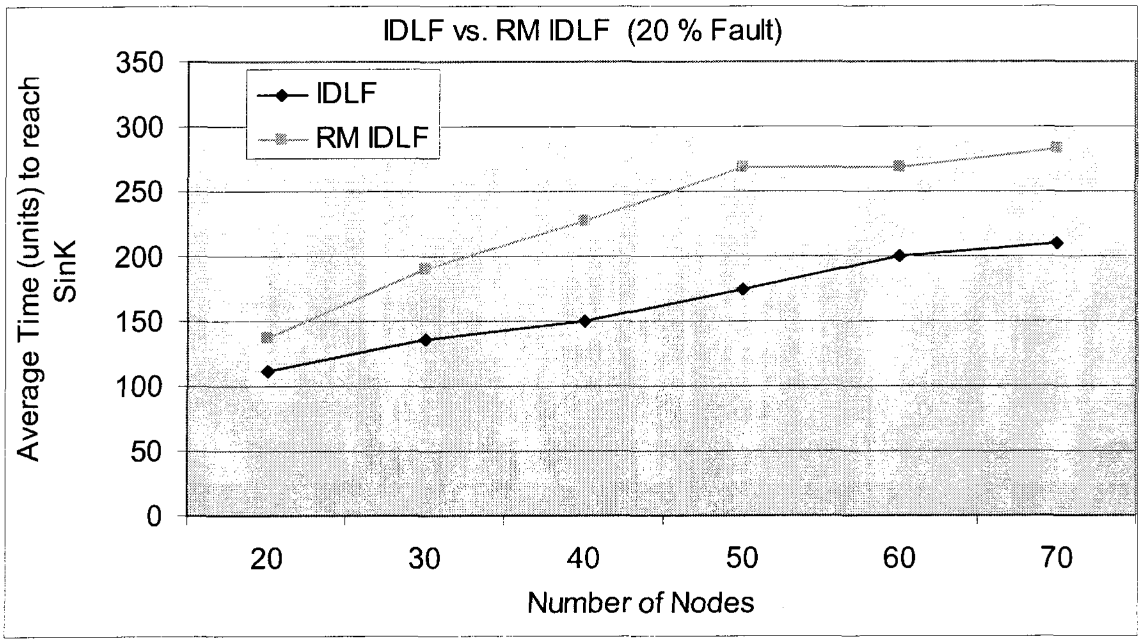


Figure 5.53 Average time to reach to sink vs. number of nodes – (20% Fault)

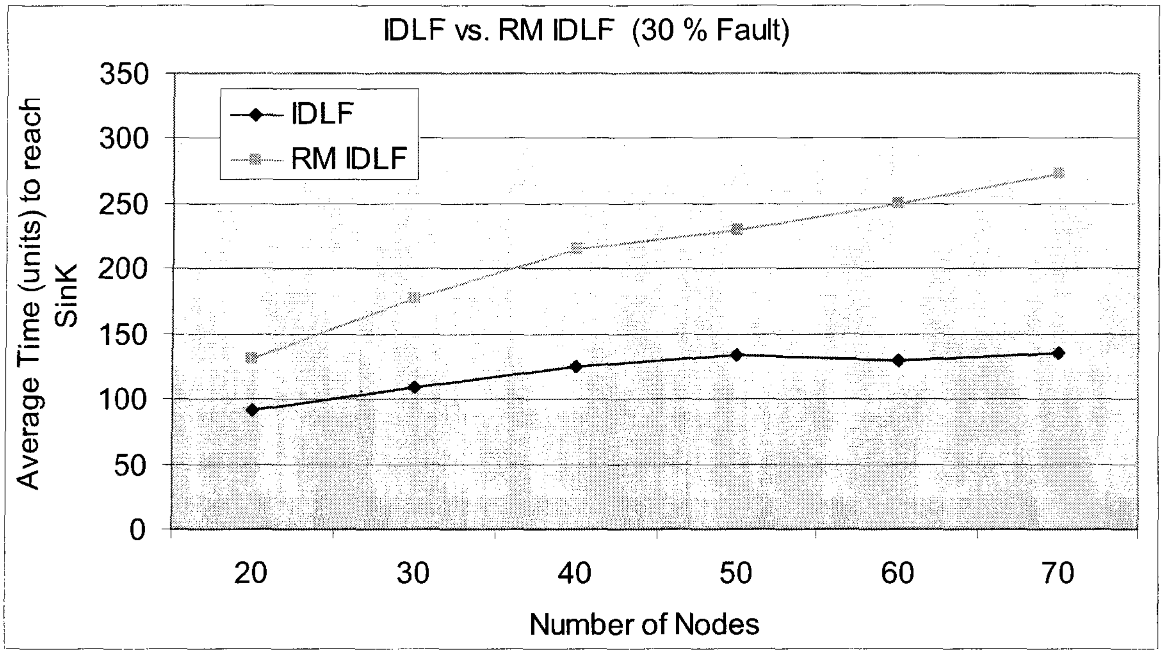


Figure 5.54 Average time to reach to sink vs. number of nodes – (30% Fault)

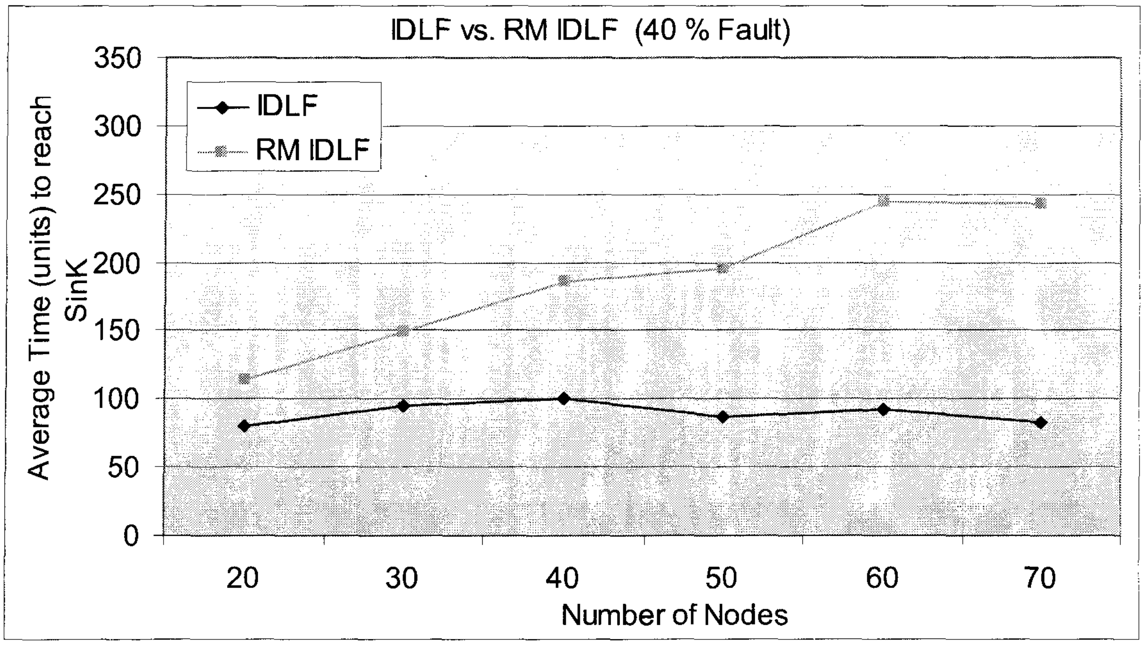


Figure 5.55 Average time to reach to sink vs. number of nodes – (40% Fault)

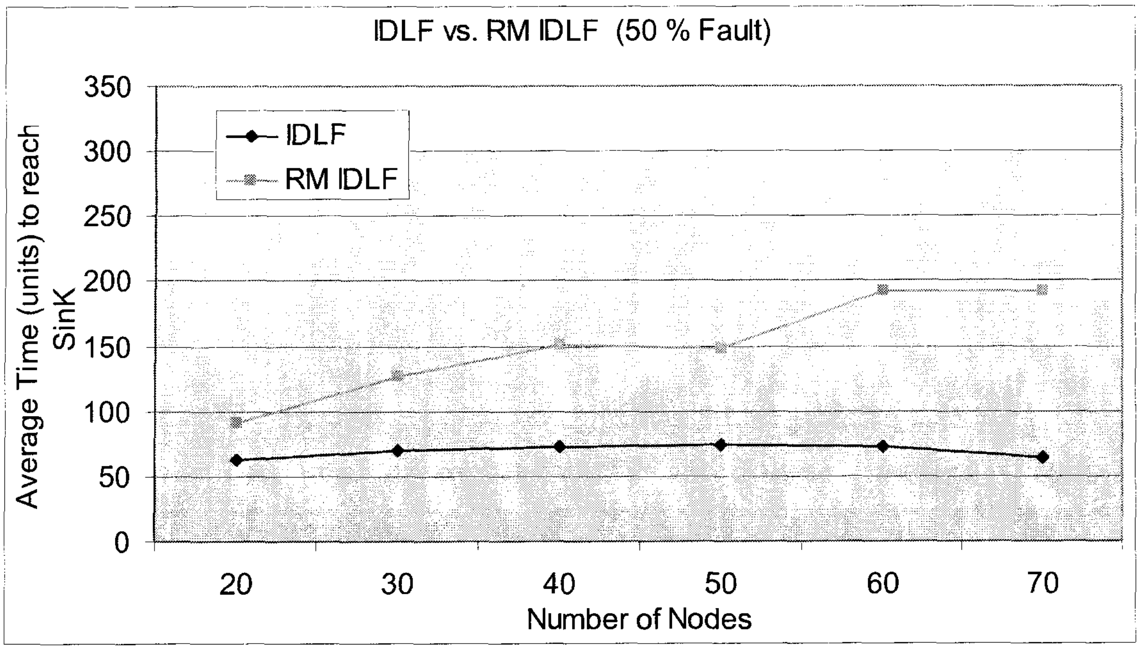


Figure 5.56 Average time to reach to sink vs. number of nodes – (50% Fault)

5.6.2 Comparison of Label Dissemination Scheme with Flooding and SPIN

In this section, we compare the performance of the RM-IDLF algorithm with flooding and SPIN on the basis of (a) total energy consumed (b) the percentage of data packets delivered by entire sensor network and (c) time it takes for data packets to reach sink node. We study each result under both fault-less and faulty situation. Each comparison involves fixed 70 nodes at the simulation time 1000 units. Figure 5.57 show that for fault free network RM-IDLF consumes 1.81 times more energy than IDLF. SPIN consumes 3.98 times more energy than RM-IDLF and Flooding consumes 1.55 times more than SPIN. As the fault percentage in the network increases, the total energy consumption decreases due to dropping of data packets. However the results show that the Flooding and SPIN still consumes more energy than RM-IDLF and IDLF. For 50% fault in the network RM-IDLF consumes 1.23 times more energy than IDLF. SPIN consumes 2.34 times more energy than RM-IDLF and Flooding consumes 2.28 times more than SPIN. Figure 5.58 shows the data packets reaching sink vs. the fault percentage. Under fault free scenario, SPIN delivers 1.18 times more data packets than flooding. SPIN delivers 1.55 times more data packets than RM-IDLF and RM-IDLF delivers 1.24 times more data packets than IDLF. With increasing number of faulty nodes within the sensor network the data packets reaching to sink decreases. For 50% fault in the network SPIN delivers 1.33 times more data packets than flooding. SPIN delivers 3.05 times more data packets than RM-IDLF and RM-IDLF delivers 2.22 times more data packets than IDLF. Figure 5.59 shows the average time to reach to the sink for each scheme. As the fault in the system increases, the average time to reach to the sink decreases. It should be noted at this point that in the case of flooding and SPIN, the increased number of data packets reaching the sink is attributed to the higher energy consumption than IDLF and RM-IDLF.

Table 5.8 Energy to packet ratio for each algorithm

	Energy to Packet Ratio			
	Flooding	SPIN	IDLF	RM-IDLF
Fault free	17.31	9.46	2.52	3.67
10 Percent	17.49	8.49	4.29	3.83
20 Percent	17.48	7.71	8.69	3.94
30 Percent	18.22	6.73	9.85	4.10
40 Percent	18.71	5.89	10.87	5.13
50 Percent	16.29	5.35	12.59	5.21

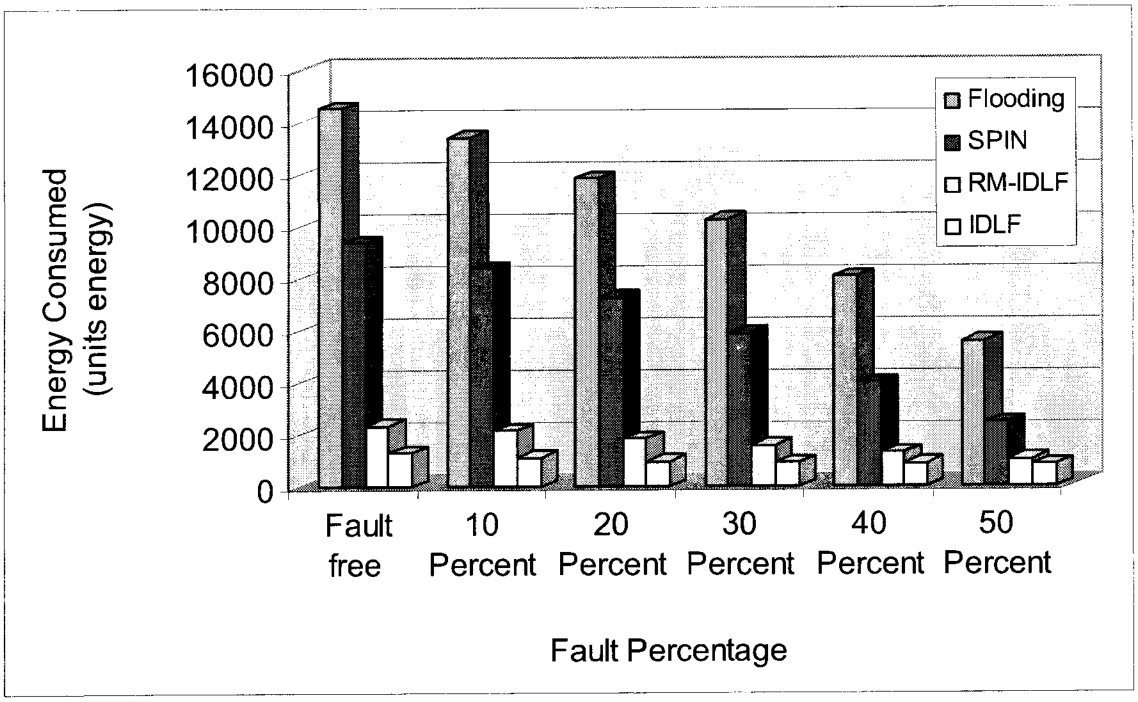


Figure 5.57 Average Energy consumed vs. Fault percentage

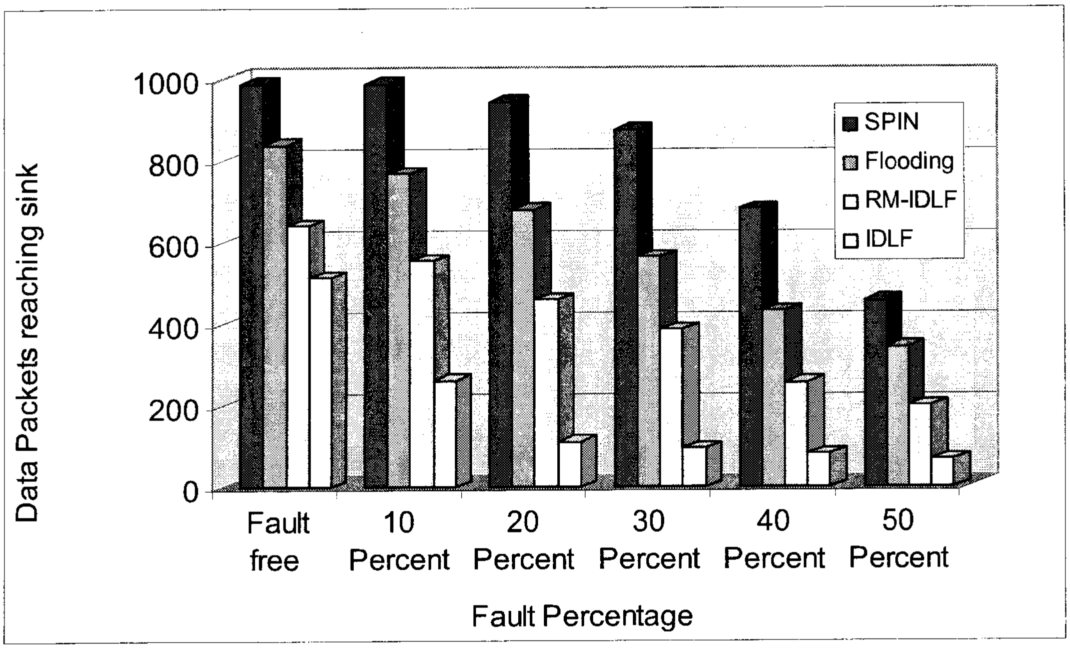


Figure 5.58 Data Packets reaching sink vs. Fault percentage

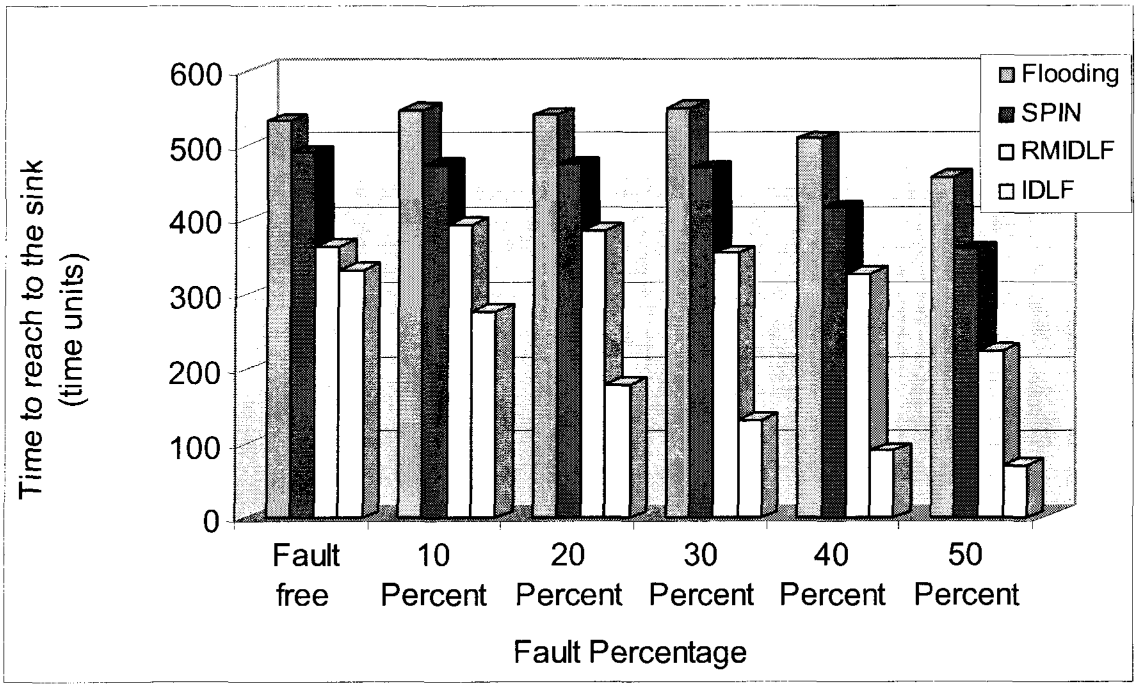


Figure 5.59 Time to reach sink vs. Fault percentage

Table 5.8 shows the ratio of *energy consumed* to *data packets reached at sink node*. For the fault less network, the energy to packet ratio for RM-IDLF is 3.67 as compared to 2.52 for IDLF, 9.46 for SPIN and 17.31 for the flooding. For the 50% fault in the network, the energy to packet ratio for RM-IDLF is 5.21 as compared to 12.59 for IDLF, 5.35 for SPIN and 16.29 for the flooding. SPIN performs better than IDLF when fault percentage increases. The reason for this behavior is that IDLF is a point to point data transmission and depends on the availability of a single path. SPIN on the other hand disseminate the data within the network using negotiation. Also, SPIN performs better than flooding in terms of energy consumption, which is true, because it is designed to prevent implosion and overlap in flooding.

CHAPTER 6

CONCLUDING NOTE

6.1 Conclusions

We have devised energy efficient routing framework for wireless sensor networks in this dissertation. The first study in the dissertation addresses the simple routing scheme with a collaborative flow of information packet/s from the source to sink. We introduce Information Dissemination via Label Forwarding (IDLF). This forwarding algorithm is a reactive and on-demand routing paradigm for distributed sensing applications. IDLF introduce a point to point data transmission where the source initiates the routing scheme and disseminates the information toward the sink (destination) node. Prior to transmission of actual data packet/s, a data tunnel is formed followed by the source node issuing small label information to its neighbors locally. These labels are in turn disseminated in the network. By using small size labels, IDLF avoids generation of unnecessary network traffic and transmission of duplicate packets to nodes. The label path ensures that a data packet is transmitted to the sink node without wasting energy on transmitting a data packet to redundant nodes. We also implemented directional flooding. In directional forwarding, sensor nodes narrow the range of broadcasting data packets based on location information about a sink to reduce transmission energy. This method is based on the assumption that each node in the sensor network has the knowledge of the probable location of the sink node. This scheme can lessen the involvement of the neighboring nodes, to which a label has to be disseminated.

For the energy efficiency, we designed and implemented discrete energy efficient schemes in conjunction to IDLF (a) Minimum transmission around the sink and (b) setting up a battery threshold value. The effective utilization of neighboring sensor nodes is seen in the first scheme. Sensor nodes located one-hop away from the sink transmit information packet only to the sink. Sensor nodes below a battery threshold value do not participate in the data dissemination process to prevent dropping important data packets. Setting up a battery threshold ensures that data packets will not be dropped after the sensor node's battery level falls below the threshold value. Minimum transmission around the sink prevents fast energy dissipation of the neighboring nodes to the sink. Finally, directional forwarding is applied to IDLF.

One of the challenges in designing a routing protocol for wireless sensor networks is to the find the most reliable path from the source to destination node, i.e. which path should deliver the data packets without retransmitting or discovering a new path. Secondly, a routing protocol for wireless sensor network should be well aware of sensor limitations. It should also take into consideration, the unique aspects of various applications running over wireless sensor networks, such as monitoring applications or acquisition of the sensitive data etc. For the reliable information dissemination, we designed and developed Reliable Information Dissemination by Label forwarding - RM-IDLF. Similar to IDLF, RM-IDLF also incorporates point to point data transmission where the source initiates the routing scheme and disseminates the information toward the sink (destination) node. We study the impact of faulty nodes on the performance of label dissemination framework. We employ an alternate disjoint path. This alternate path scheme (RM-IDLF) has a higher path cost in terms of energy consumption. However, under the faulty nodes scenario, it proves to be more reliable in terms of data packet delivery to sink than the single path scheme (IDLF). Additionally, in single path routing, as the path fails, the sink stops receiving the data packets due to absence of a backup route. In multipath routing, we took one alternate path. However, depending on the application and importance of the data delivery, the number of paths can be increased to make the system more robust. Another point of interest in this framework is the study of trade-offs between the achieved routing reliability using multiple disjoint path routing and extra energy consumption due to the use of additional path/s. We used an alternate disjoint path for ease of understanding. This alternate path scheme (RM-IDLF) may have a higher path cost in terms of energy consumption, but is more reliable in terms of data packet delivery to sink than the single path scheme (IDLF). In the latter scheme, the protocol establishes multiple (alternate) disjoint path/s from source to destination with negligible control overhead to balance load due to heavy data traffic among intermediate nodes from source to destination.

We conclude that IDLF is more suitable for disseminating information point-to-point in wireless sensor networks than flooding and SPIN. By applying directional forwarding, the average energy consumed by transmitting one data packet from a source to a sink is halved in all three routing protocols. The simulation results of other three energy management schemes also show significant improvement in total energy consumed by transmitting data. In addition to less energy consumption, when the battery threshold value scheme is applied on sensor nodes, the sensor network drops fewer number of data packets than the network without the threshold value. RM-IDLF outperforms IDLF, SPIN and Flooding in energy to packet ratio. This scheme is more suitable for large number of nodes. The overhead involving the creation of an alternate path is to be optimized with the success of receiving the information.

6.2 Future Directions

In near future sensor networks sensor networks will be an essential element in most industries, health care, environmental, agriculture and home applications. To make sensor networks truly advantageous for common applications, they must be reliable, robust, energy efficient and resistive to topology changes. Although the commercially available sensor nodes are very cheap, but designing the infrastructure and application usage cost should be minimal. Collecting data and routing appropriate and needed data to the end user is a challenging issue in such a wireless battery operated small sensor networks. Sensor information is data centric and using traditional network protocols are not always appropriate or sufficient. Power consumption is still the primary issue in the research for sensor networks. While it is often understood that sensor nodes are driven by batteries, other energy sources such as solar power may offer an unlimited power resource to a changing classification of the nodes. Since the sensor nodes can then receive and transmit packets without expending battery power, routing via these nodes is appealing. There are still a lot of studies has to be conduced in every field of sensor networks. In our simulation, we assumed that there is only one source node in a sensor network at one time. However, the network could have multiple sources at the same time.

Formation of multipaths increases the probability of data reaching the sink. More research is needed to optimize the multiple paths over the reliability of the network. We would like to implement the data dissemination algorithm on a FPGA hardware and work on improving the reliability of the system.

APPENDIX

Initialization of Algorithm

Setup Stage Algorithm

```
- Acquire the number of sensor nodes and duration of simulation
- Locate sensor nodes in the sensor field randomly
- Find neighboring nodes and store the information
```

Information Exchange Stage Algorithm

```
- Until the simulation time expires
    If it is the beginning of simulation or the data reached to sink
      Assign a new source node
      Record the source information to the source node
- Add sets of pair nodes, between which a label is transmitted (source to neighbors), to the Waiting List
-Check all elements in the Waiting List
   If pair of nodes finished exchanging and processing information
     Delete the pair's info from the Waiting List
-If the information was a Label
```

```
          Copy the label information to the receiving node
          Add sets of pair nodes, between which a label is transmitted to the Waiting List.
          Exclude the neighbors which are the sender of the Label and the source.
  -If the new sending node is neighbor to the sink, transmit the label only to the sink
             If the receiving node was the sink, prepare to transmit a request to send
            (REQ) signal
 - If the information was a REQ
                  -Save the sender node's ID to the receiver
                  -Add pair nodes, between which a REQ signal is exchanged, to the
                  Waiting List
 -If the receiving node was the source, prepare to transmit a Data
      If the information was a Data
          If the receiving node was the sink
             Empty the Waiting List
            Clear the Label cache in the each node
        If the receiving node was not the sink
           Add pair nodes, between which a Data signal is   exchanged, to the Waiting List


  - Update the Waiting List (Sort its' index numbers)
        Until there is no more pair nodes in the Waiting List
                  Randomly select the pair nodes, between which the information will  be
                 exchanged, from the Waiting List
```

```
        The selected pair's transmission range must not interfere with pairs'
        transmission ranges, which have been already selected before
      - Decrement the simulation time
```

Pseudo code for IDLF scheme

```
1) Output 1--> Total number of data packets reaching the sink (destination) node
   Output 2--> Total Energy Consumed in the network (output1.txt) and (output2.txt)
2) Ask User to Enter
      (a) The number of nodes (numofnodes)
      (b) Simulation time (simtime)
3) Initialize..... {Function Initialize}
      (a) the Grid (10 x 10) map
      (b) Each node's parameter----> x_loc, y_loc, power, neighbors, and labelcache
          (defined in structset.h)
4) Allocate Node.... {Function AllocateNode}
      (a) Allocate Sink at (0, 0)
      (b) Allocate all nodes (numofnodes) by
          - getting a random number
          - check if the location is not assigned yet
          - check if location is within the transmission range of the existing node
  (Note: the transmission range is r1 and a node can travel to 8 grinds around itself)
      (c) If yes- node is allocated
```

(d) If not- node is not allocated, find another location

5) Neighbor Discovery Phase.... *{Function NeighborDisc}*

(a) Find nodes within the r1 range

(b) Check if the x- axis is in the simulation area (within the grid)

(c) Check if the y- axis is in the simulation area (within the grid)

(d) Find location to store the node information

(e) Int D_type → 1=label ; 2=Req ;3=data

6) Data Propagation .... *{Function DataProp}*

(a) Maintain a list of node pair waiting for transmission – *(wlist)*

(b) Assign a Source Node *(AssignSource)* - make sure the assigned source node is not sink node and after each event *(i.e. after the data packet reaches to sink)* select a random source

(c) Check if sink is the neighbor to source, If yes then only send label to sink

(d) Transmit label - *{Function AddtoWList}* If sink is not the neighbor to source, then send label to the neighboring nodes

(e) Check WaitingList- If waiting label – stop transmission also

(f) Check if a receiver already has the label- *(CheckContention)* If the label in label cache has the same source and start time, the receiver already has it – and then we are not sending label to that neighbor

(g) If newly arrived label is not in cache save label in cache and transmit to further neighbors

(h) If receiving node is sink, transmit *Request packet* else If receiving node is not sink transmit *Label packet*

(i) Save label to node's label cache and Reach at sink

(j)Req transmission →Find the corresponding data in label cache→ Found the label in cache→save the child Id in node's label cache→Add sets of pair nodes to Waiting List→Check if the receiver node is source

(h)Data transmission→ Find the corresponding data in label cache→ Found the label in cache→Add sets of pair nodes to Waiting List→Decrement transmission time in the list→If the data reaches to the sink, empty the list

Pseudo Code for the RM-IDLF

1) Output 1--> Whether Data packet reaching the sink (destination) node- YES/NO

Output 2--> Total Energy Consumed in the network *(output1.txt)" (output2.txt)"*

i.e. the energy consumed for data propagation from the source to sink

[Continuous simulations over the time period provides the overall evaluation of the scheme]

2) Ask User to Enter

(a) The number of nodes *(numofnodes)*

(b) Simulation time *(simtime)*

(c) The failure Rate * (in Percentage)- i.e. 20% means , out of total selected nodes say 30, we are failing 20% of the nodes for the simulation- (Note : failure of node is occurring, just before the data transmission, WORST CASE)

3) Initialize..... {Function Initialize}

(a) Grid (10 x 10) map

(b) Each node's parameter----> x_loc, y_loc, power, neighbors, labelcache

(defined in structset.h)

4) Allocate Node....{Function AllocateNode}

(a) Allocate Sink at (0, 0)

(b) Allocate all nodes *(numofnodes)* by

- getting a random number
- check if the location is not assigned yet
- check if location is within the transmission range of the existing node

(Note: the transmission range is r1 and a node can travel to 8 grinds around itself)

(c) If yes- node is allocated

(d) If not- node is not allocated, find another location till you allocate

5) Neighbor Discovery Phase.... {Function *NeighborDisc}*

(a) Find nodes within the r1 range

(b) Check if the x- axis is in the simulation area (within the grid)

(c) Check if the y- axis is in the simulation area (within the grid)

(d)Find location to store the node information

(e) Int D_type → 1=label ; 2=Req ;3=data

6) Data Propagation .... *{Function DataProp}*

(a) Maintain a list of node pair waiting for transmission – *(wlist)*

(b) Assign a Source Node *(AssignSource)* - make sure the assigned source node is not sink node [2]

(c) Check if sink is the neighbor to source, If yes then only send label to sink

(d) Transmit label - *{Function AddtoWList}* if sink is not the neighbor to source, then send label to the neighboring nodes

(e) Check WaitingList- If waiting label – stop transmission also

(f) Check if a receiver already has the label- *(CheckContention)*

(g) If newly arrived label is not in cache save label in cache and transmit to further neighbors

(h) If receiving node is sink, transmit Request packet else receiving node is not sink, transmit Label packet

(i) Save label to node's label cache and Reach at sink

(j) Also, wait for X amount of time and let another label path (an alternate path be created → and follow steps (l) and (m)

(k) Physically Destroy the nodes (Depending on the percentage, the user selects) - Nodes are as good as empty slots, we cannot expect any communication from the destroyed nodes (NodeDestroy)* in other words, source and Sink will not be affected at all by the node failure. Fail node before the actual data transmission.

(l)Req transmission →Find the corresponding data in label cache→ Found the label in cache→Save the child Id in node's label cache→Add sets of pair nodes to Waiting List→Check if the receiver node is source

---

[2] [In IDLF after each event (i.e. after the data packet reaches to sink, we select a random source; In Multi-path IDLF, we are stopping after the data reaches the sink, start a new simulation, its because now (after intentionally failing some nodes) we are more concerned about whether the data packet is reaching the sink or not]

(m)Data transmission→ Find the corresponding data in label cache→ Found the label in cache→Add sets of pair nodes to Waiting List→Decrement transmission time in the list→If the data reaches to the sink, empty the list

BIBLIOGRAPHY

[1] Mark Weiser, Some Computer Science Problems in Ubiquitous Computing, Communications of the ACM, July 1993. (Reprinted as Ubiquitous Computing, Nikkei Electronics, December 6, 1993, pages. 137-143.)

[2] D. Culler, D. Estrin, and M. Srivastava, “Guest Editors’ Introduction: Overview of Sensor Network,” Computer, Volume: 37 Issue: 8, pp. 41-49, August 2004

[3] Mohammad Ilyas, Imad Mahgoub, "The Handbook of Ad Hoc Wireless Networks" CRC Press 2002, Hardcover

[4] G. J. Pottie and W. J. Kaiser, “Wireless Integrated Network Sensors,” Communications of ACM, 43(5), May 2000, pp. 51-58.

[5] S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman, “A Taxonomy of Wireless Micro-Sensor Network Models,” ACM Mobile Computing and Communications Review (MC2R), Volume 6, Number 2, April 2002, pp. 28-36.

[6] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, “Next century challenges: Scalable coordination in sensor networks,” In Proceedings of International Conference MOBICOM, pp. 263-270, 1999.

[7] D. Estrin, L. Girod, G. Pottie and M. Srivastava, Instrumenting the world with wireless sensor networks, Proc. of the International Conference on Acoustics, Speech and Signal Processing, (ICASSP 2001)

[8] P. Rentala, R. Musunuri, S. Gandham, U. Saxena , University of Texas, Dallas

Survey on sensor networks, Technical Report- UTDCS-33-02, University of Texas at Dallas, 2002.

[9] Tilak, S., Abu-Ghazaleh, N.B. and Heinzelman, W., Taxonomy of Wireless Sensor Network Communication Models. Mobile Computing and Communication Review, April 2002, Vol. 6, No. 2.

[10] MICA Wireless measurement Systems, Crossbow Technology Inc. http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA.pdf, Document part number: 6020-0041-01

[11] Hsu, V. S., Kahn, J. M., and Pister, K. S. J. (1998), "Wireless Communications for SMART DUST," Electronics Research Laboratory Technical Memorandum Number M98/2, University of California, Berkeley

[12] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Emerging Challenges: Mobile Networking for 'Smart Dust,'" J. Communications and Networks, Vol. 2, No. 3, September 2000.

[13] B. Warneke, M. Last, B. Leibowitz, and K. S. J. Pister, "Smart Dust: Communicating with a Cubic-Millimeter Computer," Computer Magazine, January 2001

[14] Doug Smart Dust: ISRC Technology Briefing, March 2005, http://www.uhisrc.com/FTB/Smart Dust/Smart Dust brief 5B1D-Doug2005.doc

[15] SMART DUST : Autonomous sensing and communication in a cubic millimeter Working group,http://www-bsac.eecs.berkeley.edu/~pister/SmartDust/,University of California Berkeley

[16] L. Doherty, B. A. Warneke, B. E. Boser, and K. S. J. Pister, "Energy and

Performance Considerations for Smart Dust," Int. J. Parallel Distributed Systems and Networks, Vol. 4, No. 3, 2001.

[17] B. Warneke, B. Atwood, and K. S. J. Pister, Smart Dust Mote Forerunners, Proceedings of International Conference on Micro-electromechanical Systems, Interlaken, Switzerland, January 2001.

[18] B. Warneke, M. Scott, B. Leibowitz, L. Zhou, C. Bellew, J. M. Kahn, B. E. Boser, and K. S. J. Pister, "An Autonomous 16 mm3 Solar-Powered Node for Distributed Wireless Sensor Networks," IEEE Int. Conf. Sensors, Orlando, FL, June 2002.

[19] S.E.Hollar, COTS Dust, http://www-bsac.eecs.berkeley.edu/archive/ users/hollar-seth/publications/cotsdust.pdf, December 2000

[20] S.E.Hollar, Macro-Motes, http://www-sac.eecs.berkeley.edu/archive/users/hollar-seth/macro_motes/macromotes.html, April 2004

[21] S.E.Hollar, Matthew Last, Laser Turret Test-bed http://www.bsac.eecs.berkeley.edu/ ~mattlast/research/laser_turret.html

[22] Crossbow - Wireless Sensor Networks Product page http://www.xbow.com/products/Wireless_Sensor_Networks.htm, April 2004

[23] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," Computer Networks, 2002, pp. 393-422.

[24] I. F Akyildiz, MC Vuran, OB Akan, W Su, "–Wireless Sensor Networks: A Survey revisited" - Computer Networks Journal (Elsevier), 2005

[25] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan , Energy-Efficient Communication Protocol for Wireless Microsensor Networks,

Proceedings of the 33rd Hawaii International Conference on System Sciences – 2000

[26] V Rajendran, K Obraczka, JJ Garcia-Luna-Aceves ,Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks - ACM SenSys 2003, 2003

[27] Samir Goel and Tomasz Imielinski, Etiquette Protocol for Ultra Low Power Operation in Sensor Networks. ACM Transactions on Sensor Networks-March 2005

[28] T. van Dam and K. Langendoen, An adaptive energy efficient MAC protocol for wireless sensor networks. In Proc. 1st ACM Conf. Embedded Networked Sensor Systems (SenSys 2003), pages 171–180. ACM Press, New York, November 2003.

[29] C. K. Toh, Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks, Communications Magazine, IEEE, Volume: 39, Issue: 6, pp. 138-147, June 2001.

[30] K. Scott and N. Bambos, "Routing and channel assignment for low power transmission in PCS," 5th IEEE International Conference on Universal Personal Communications, Volume: 2, pp. 498 - 502, October 1996.

[31] B. Krishnamachari, Y. Mourtada, and S. Wicker, The Energy-robustness Tradeoff for Routing in Wireless Sensor Networks, Autonomous Networks Research Group Technical Report ANRG TR 02-001, September 2002.

[32] Kay Römer, Oliver Kasten, Friedemann Mattern, Middleware Challenges for Wireless Sensor Networks, ACM Mobile Computing and Communication Review, Vol. 6, No. 4, pp. 59-61, October 2002

[33] A. Howard, M. J. Mataric, and G. S. Sukhatme, an Incremental Self-Deployment Algorithm for Mobile Sensor Networks," Autonomous Robots, Special Issue on Intelligent Embedded Systems, 13(2), September 2002, pp. 113-126.

[34] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan and K. Saluja, Sensor deployment strategy for target detection, WSNA 2002, pp. 42-48.

[35] V. Mhatre and C. Rosenberg, "Energy and cost optimizations in wireless sensor networks: A survey," in Performance Evaluation and Planning Methods for the Next Generation Internet (A. Girard, B. Sanso, and F. Vazquez-Abad, eds.), 25th Anniversary Volumes of GERAD (Group for Research in Decision Analysis), Kluwer Academic Publishers, May 2005.

[36] K. Xu, X. Hong, M. Gerla, H. Ly, D. Lihui Gu; "Landmark Routing in Large Wireless Battlefield Networks using UAVs"; Military Communications Conference (MILCOM), 2001; Vol. 1; October 2001; pp. 230-234.

[37] Y. Ge, L. Lamont, L. Villasenor. "Improving Scalability of Heterogeneous Wireless Networks with Hierarchical OLSR", the OLSR Interop & Workshop, San Diego, USA, August 2004.

[38] U. Roedig, A. Barroso, and C. J. Sreenan, Determination of Aggregation Points in Wireless Sensor Networks. In Proceedings of the 30th Euromicro Conference (EUROMICRO2004), Rennes, France, IEEE Computer Society Press, August 2004, pp. 503–510.

[39] T. Yang, T. He and J. Stankovic, Differentiated Surveillance for Sensor Networks, Sensys 2003, November 2003.

[40] GPS-less low cost outdoor localization for very small devices, Nirupama Bulusu, John Heidemann and Deborah Estrin, IEEE Wireless Communications, Vol 7. No.5, pp. 27-34, Oct 2000.

[41] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," In Proceedings of 5th ACM/IEEE Mobicom Conference, August 1999.

[42] B. Krishnamachari, Y. Mourtada, and S. Wicker, The Energy-robustness Tradeoff for Routing in Wireless Sensor Networks, Autonomous Networks Research Group Technical Report ANRG TR 02-001, September 2002.

[43] H. Lim and C. Kim, Multicast Tree Construction and Flooding in Wireless Ad Hoc Networks, In Proceedings of ACM Modeling, Analysis, and Simulation of Wireless and Mobile Systems, pp. 61-68, 2000.

[44]. S. Brennan, A. Mielke, D. Torney, "Radiation Detection with Distributed Sensor Networks", Computer, Vol. 37, No. 8, August 2004, pp. 57-59.

[45] M. Maroti, G. Simon, A. Ledeczi, J. Sztipanovits, "Shooter Localization in Urban Terrain", Computer, Vol. 37, No. 8, August 2004, pp. 60-61.

[46]. K. Martinez, J. Hart, R. Ong, Environmental Sensor Networks, Computer, Vol. 37, No. 8, August 2004, pp. 50-56.

[47] J. Burrell, T. Brooke, R. Beckwith, "Vineyard Computing: Sensor Networks in Agricultural Production", IEEE Pervasive Computing, Vol. 3, No. 1, January – March 2004, pp. 38-45.

[48] Adrian Perrig, John Stankovic, David Wagner, "Security in wireless sensor networks", ACM-SPECIAL ISSUE: Wireless sensor networks, Volume 47, Issue 6 (June 2004), Pages: 53 - 57

[49] A. F. Harris III and R. Kravets, "Pincher: A Power-Saver Wireless Communication Protocol," UIUCDCS-R-2003-2368, December 2003, pp. 1-12.

[50] C. K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," IEEE Communications Magazine, Volume: 39 Issue: 6, June 2001 pp. 138-147

[51] C. Toh, H. Cobb, and D. Scott, Performance evaluation of battery-life-aware routing schemes for wireless ad hoc networks, IEEE International Conference on Communications, June 2001, Volume 9, Page(s):2824 - 2829

[52] M. Khan, G. Pandurangan, and B.Bhargava, Energy-Efficient Routing Schemes for Wireless Sensor Networks, Technical Report CSD TR 03-013, Dept. of Computer Science, Purdue University, 2003, pp. 1-12.

[53] S. Banerjee and A. Misra, "Minimum energy paths for reliable communication in multi-hop wireless networks," Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing Lausanne, Switzerland Pages: 146 – 156, 2002

[54] Jamil Ibriq, Imad Mahgoub,"Cluster-Based Routing in Wireless Sensor Networks: Issues and Challenges", Proceedings of the 2004 Symposium on Performance Evaluation of Computer and Telecommunication Systems- SPECTS, July 25-29 2004

[55] Narayanan Sadagopan, Bhaskar Krishnamachari, and Ahmed Helmy, "Active Query Forwarding in Sensor Networks (ACQUIRE),"Journal of Ad Hoc Networks, Elsevier, Vol 3, Issue 1, pp. 91-113, January 2005.

[56] Stephanie Lindsey, C. S. Raghavendra, "PEGASIS: Power Efficient Gathering in Sensor Information Systems," 2002 IEEE Aerospace Conference, March 2002.

[57] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann and Fabio Silva, Directed Diffusion for Wireless Sensor Networking IEEE/ACM transactions on networking, vol. 11, no. 1, February 2003

[58] Koustuv Dasgupta, Konstantinos Kalpakis, Parag Namjoshi "An Efficient Clustering--based Heuristic for Data Gathering and Aggregation in Sensor Networks". IEEE Wireless Communications and Networking Conference (WCNC), New Orleans, Louisiana, USA, March 16-20, 2003.

[59] Jamal N. Al-Karaki Ahmed E. Kamal, Routing Techniques in Wireless Sensor Networks: A Survey", IEEE Wireless Communications, Volume: 11, Issue: 6 6- 28, Dec. 2004

[60] H. Sabbineni and K. Chakrabarty, “Location-Aided Flooding: An Energy-Efficient Data Dissemination Protocol for Wireless Sensor Networks,” IEEE Transactions on Computers, Volume 54, Issue 1, pp. 36-46, January 2005.

[61] J. Kulik, W. R. Heinzelman, and H. Balakrishnan, "Negotiation-based protocols for disseminating information in wireless sensor networks," Wireless Networks, Volume: 8, pp. 169-185, 2002.

[62] Archana Bharathidasan and Vijay Anand Sai Ponduru,"Sensor networks: An overview" Department of Computer Science University of California, Davis, UCDAVIS 2003.

[63] Gunjan Khanna, Saurabh Bagchi, Yu-Sung Wu," Fault Tolerant Energy Aware Data Dissemination Protocol in Sensor Networks " 2004 International Conference on Dependable Systems and Networks (DSN'04) 2004 Florence, Italy, pp 795

[64] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," Proceedings of ACM MobiCom '00, Boston, MA, 2000, pp. 56-67.

[65] D. Braginsky and D. Estrin, "Rumor Routing Algorithm for Sensor Networks," in the Proceedings of the First Workshop on Sensor Networks and Applications (WSNA), Atlanta, GA, October 2002.

[66] Yong Yao and J. E. Gehrke, The Cougar Approach to in-Network Query Processing in Sensor Networks. Sigmod Record, Volume 31, Number 3, September 2002.

[67] N. Sadagopan et al., The ACQUIRE mechanism for efficient querying in sensor networks, in the Proceedings of the First International Workshop on Sensor Network Protocol and Applications, Anchorage, Alaska, May 2003.

[68] S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman, "A Taxonomy of Wireless Micro-Sensor Network Models," ACM Mobile Computing and Communications Review (MC2R), Volume 6, Number 2, April 2002,pp. 28-36

[69] Stephanie Lindsey, C. S. Raghavendra, "PEGASIS: Power Efficient Gathering in Sensor Information Systems", 2002 IEEE Aerospace Conference, Volume 3, 9-16 March 2002 Page(s):3-1125 - 3-1130

[70] Rajgopal Kannan Ram Kalidindi S. S. Iyengar," Energy and rate based MAC protocol for wireless sensor networks", ACM SPECIAL ISSUE: Special section on sensor network technology and sensor data management 2003 Pp: 60 – 65, Volume 32, Issue 4

[71] Chansu Yu, S Moh, B Lee, H Y Youn, "Energy Efficient and Robust Multicast Protocol for Mobile Ad Hoc Networks "2002 Pacific Rim Intl Symposium on Dependable Computing (PRDC'02) 2002 Tsukuba, Japan, pp 145

[72] K. Scott and N. Bambos, "Routing and channel assignment for low power transmission in PCS," 5th IEEE International Conference on Universal Personal Communications, Volume 2, 29 Sept.-2 Oct. 1996 Page(s):498 - 502

[73] K. Dongkyun, J. Park, C-K Toh, Y. Choi, "Power-aware Route Maintenance Protocol for Mobile Ad Hoc Network,". 10th International Conference on Telecommunications, 2003 ICT 2003. Volume 1, 23 Feb.-1 March 2003 Page(s):501 - 506

[74] A. Manjeshwar and D. P. Agrawal, TEEN: A Protocol for Enhanced Efficiency in Wireless Sensor Networks, Proceedings of the 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001.

[75] A. Manjeshwar and D. P. Agrawal, "APTEEN: A Hybrid Protocol for Efficient Routing and Comprehensive Information Retrieval in Wireless Sensor Networks,"

in the Proceedings of the 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile computing, Ft. Lauderdale, FL, April 2002.

[76] L. Subramanian and R. H. Katz, "An Architecture for Building Self Configurable Systems, in the Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing, Boston, MA, August 2000.

[77] A. Savvides, C-C Han, and M. Srivastava, Dynamic fine-grained localization in Ad-Hoc networks of sensors, Proceedings of the Seventh ACM Annual International Conference on Mobile Computing and Networking (MobiCom), July 2001. Pages 166-179

[78] S. Capkun, M. Hamdi, J. Hubaux, GPS-free positioning in mobile ad-hoc networks, Proceedings of the 34th Annual Hawaii International Conference on System Sciences, 2001 pp. 3481-3490.

[79] Y. Xu, J. Heidemann, D. Estrin, Geography-informed Energy Conservation for Ad-hoc Routing," In Proceedings of the seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking 2001, pp. 70-84.

[80] K. Akkaya and M. Younis, A Survey of Routing Protocols in Wireless Sensor Networks, Elsevier Ad Hoc Network Journal

[81] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in the Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'01), Rome, Italy, July 2001.

[82] V. Rodoplu and T.H. Ming, Minimum energy mobile wireless networks, IEEE Areas in Communications, Vol. 17, pp. 1333-1344, 1999

[83] L. Li and J. Y Halpern, Minimum energy mobile wireless networks revisited, in the Proceedings of IEEE International Conference on Communications (ICC'01), Helsinki, Finland, June 2001

[84] G. G. Finn, Routing and Addressing Problems in Large Metropolitan-Scale Inter-networks, ISI res. rep. ISU/RR-87-180, Mar.1987

[85] E. Kranakis, H. Singh, and J. Urrutia, "Compass Routing on Geometric Networks," in Proc. 11th Canadian Conference of Computational Geometry, Aug. 1999.

[86] P. Bose et al., Routing with Guaranteed Delivery in Ad Hoc Wireless Networks, In Proceedings 3rd International Workshop, Discrete Algorithms Methods Mobile Computation, Seattle, WA, Aug. 20, 1999, pp. 48–55; also in ACM/Kluwer Wireless Networks., vol. 7, no. 6,Nov. 2001, pp. 609–616.

[87] V. Rodoplu and T.H. Ming, "Minimum energy mobile wireless networks," IEEE Journal of Selected Areas in Communications, Vol. 17, No. 8, pp. 1333-1344, 1999.

[88] L. Li, and J. Y. Halpern, Minimum Energy Mobile Wireless Networks Revisited, IEEE International Conference on Communications (ICC) Helsinki, Finland, June 2001. Vol. 1, pp. 278-283

[89] Y. Yu, D. Estrin, and R. Govindan, Geographical and Energy-Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks", UCLA

Computer Science Department Technical Report, UCLA-CSD TR-01-0023, May 2001.

[90] J.W. Suurballe, The single-source, all terminals problem for disjoint paths, unpublished technical memorandum, Bell Laboratories (1982).

[91] J.W. Suurballe and R.E. Tarjan, A quick method for finding shortest pairs of disjoint paths, Networks 14 (1984) 325–336.

[92] A. Srinivas, E. Modiano, Minimum energy disjoint path routing in wireless ad-hoc network, Proceedings of ACM MOBI COM'2003, pp. 122-133.

[93] G. Gupta, M. Younis, Fault-Tolerant clustering of wireless sensor networks, Proceedings of IEEE WCNC-2003, pp. 1579-1584

[94] B. Hao, J. Tang and G. Xue, “Fault-tolerant relay node placement in wireless sensor networks: formulation and approximation”, HPSR'04: IEEE Workshop on High Performance Switching and Routing; April 19-21, 2004, Phoenix, Arizona; pp. 246-250

[95] C. Rahul, J. Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks", IEEE Wireless Communications and Networking Conference (WCNC), vol.1, March 17-21, 2002, Orlando, FL, pp. 350-355.

[96] Q. Li, J. Aslam & D. Rus, Hierarchical Power-aware Routing in Sensor Networks, Proceedings of the DIMACS Workshop on Pervasive Networking, May, 2001.

[97] S. Dulman, T. Nieberg, J. Wu, P. Havinga,"Trade-Offs between Traffic Overhead and Reliability in Multipath Routing for Wireless Sensor Networks", WCNC Workshop, New Orleans, Louisiana, USA, March 2003.

[98] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, Highly-resilient, energy-efficient multipath routing in wireless sensor networks", ACM SIGMOBILE Mobile Computing and Communications Review, vol. 5, no. 4, pp. 1125, 2001.

[99] K. Sohrabi, J. Pottie, Protocols for self-organization of a wireless sensor network, IEEE Personal Communications, Volume 7, Issue 5, pp 16-27, 2000.

[100] C. Shen, C. Srisathapornphat, C. Jaikaeo, Sensor information networking architecture and applications, IEEE Personal Communications, August 2001, pp. 52–59

[101] G. Hoblos, M. Staroswiecki, A. Aitouche, Optimal design of fault tolerant sensor networks, IEEE International Conference on Control Applications, Anchorage, AK, September 2000, pp. 467–472.

[102] L. Lamport, R. Shostak, and M. Pease, "The Byzantine Generals Problem," ACM Transaction on Programming Languages and Systems, vol. 4, no. 3, pp. 382–401, July 1982.

[103] Thomas Clouqueur, Parameswaran Ramanathan, Kewal K. Saluja and Kuang-Ching Wang, "Value-Fusion versus Decision-Fusion for Fault-tolerance in Collaborative Target Detection in Sensor Networks", Proceedings of the 4th Ann. Conf. on Information Fusion, pp. TuC2/25-TuC2/30, Aug. 2001. (PS, PDF)

[104] K Marzullo," Implementing fault tolerant sensors", TR89-997, Department of computer Science, Cornell University, May 1989

[105] L. Prasad, S.S.Iyengar, R. L. Kashyap and R. N. Madan, "Functional Characterization of Fault tolerant Integration in Distributed Sensor Networks", IEEE Transactions on SMC, Vol.21, No.5, Sept.-October 1991, pp. 1082 - 1087.

[106] S.S.Iyengar, R. L. Kashyap, L. Prasad, and R. N. Madan, "Functional Characterization of Sensor Integration in Distributed Sensor Networks", Proceedings of Fifth International Parallel Processing Symposium, April - May 1991.

[107] Chih-Wei, Peng-Jun Wan, Xiang-Yang Li, Ophir Frieder, "Fault Tolerant Sensor Networks with Bernoulli Nodes", IEEE Wireless Commnication and Networking Conference (WCNC), New Orleans, Louisiana, March 2003.

[108] F. Koushanfar, S. Slijepcevic, M. Potkonjak, A. Sangiovanni-Vincentelli, Error-Tolerant Multi-Modal Sensor Fusion, IEEE CAS Workshop on Wireless Communication and Networking, Pasadena, CA, Sep. 5-6, 2002.

[109] C. Schurgers, G. Kulkarni, and M. B. Srivastava, "Distributed On-Demand Address Assignment in Wireless Sensor Networks," IEEE Transactions on Parallel and Distributed Systems, Volume 13, Issue10, pp. 1156-1165, October 2002.

[110] Hull, Bret, Kyle Jamieson, Hari Balakrishnan, "Bandwidth Management in Wireless Sensor Networks", Networks and Mobile Systems 2003, MIT-LCS-TR-909

[111] P.R. Kumar, "New technological vistas for systems and control: the example of wireless networks," IEEE Control Systems Magazine, pp. 24-37, Feb. 2001. F.L. Lewis, Optimal Estimation, Wiley, New York, 1986.

[112] P. Gupta and P.R. Kumar, The capacity of wireless networks, IEEE Transactions on Information Theory, vol. 46, pp. 388-404, Mar. 2000.

[113] Manual of Network Simulator NS-2 http://www.isi.edu/nsnam/ns/.

[114] W. J. Kaiser, WINS-NG 1.0 Transceiver Power Dissipation Specifications, Sensoria Corp.

[115] Jain Neha, Energy Aware and Adaptive Routing Protocols in Wireless Sensor Networks, PhD Dissertation, University of Cincinnati, Engineering : Computer Science & Engineering, 2004.

[116] J. Chang and L. Tassiulas, Energy Conserving Routing in Wireless Ad-hoc Networks, Proceedings of IEEE Infocom, pp. 22-31, Mar. 2000, Tel-Aviv, Israel.

[117] N. Jain, D. K. Madathil, and D. P. Agrawal, "Energy Aware Multi-Path Routing for Uniform Resource Utilization in Sensor Networks," Proceedings of the IPSN'03 International Workshop on Information Processing in Sensor Networks, Palo Alto, CA, 22nd April, 2004.

[118] Kimura Naoto, Information dissemination by label forwarding, MS Thesis, University of Nevada Las Vegas, -Joint work with the author.

[119] S. D. Servetto and G. Barrenechea, "Constrained Random Walks on Random Graphs: Routing Algorithms for Large Scale Wireless Sensor Networks," In Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, pp. 12-21, September2002, Atlanta, Georgia.

[120] S. Lee and M. Gerla, "Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks," Proceedings of the International Conference on Communications (ICC 2001),pp. 3201-3205, 2001

[121] R.C. Shah and J. Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks, "In Proc. of IEEE Wireless Communications and Networking Conference (WCNC), March 2002, Orlando, Florida.

[122] Reliability modeling in Sensor Networks, Bein D, Jolly V, Kumar B, Latifi S, International Journal of Information Technology Special Issue, Volume 11, no. 2, 2005 pp 1-8

[123] D. Nadig, S.S. Iyengar, A new architecture for distributed sensor integration, Proceedings of IEEE Southeastcon'93, Charlotte, NC, April 1993.

[124] Minimum energy fault tolerant sensor networks Djukic, P.; Valaee, S.; Global Telecommunications Conference Workshops, 2004. GlobeCom workshop 2004, IEEE 29 Nov.-3 Dec. 2004 Page(s):22 - 26

VITA

Graduate College
University of Nevada, Las Vegas

Vasu Jolly

Home Address:
772 Wigan Pier Dr
Henderson, NV 89015

Degrees:
Bachelor of Engineering,
Electrical and Electronics Engineering, 1999
Nagpur University, India

Master of Science Degree in Electrical Engineering
Department of Electrical and Computer Engineering
Howard R. Hughes College of Engineering
University of Nevada Las Vegas
December 2002

Dissertation Title: Reliable and Resource Aware Framework for Data Dissemination in Wireless Sensor Networks

Dissertation Examination Committee:
Chairperson, Dr. Shahram Latifi, Ph.D. (Supervisor)
Committee Member, Dr. Henry Selvaraj, Ph.D.
Committee Member, Dr. Emma Regentova, Ph.D.
Committee Member, Dr. Sahjendra Singh, Ph.D.
Graduate Faculty Representative, Dr Wolfgang Bein, Ph.D.

Publications (Relevant to Dissertation)

Power Related Issues in Wireless sensor networks, *Vasu Jolly, N kimura, P Srimani, S Latifi, appeared under a Special issue: Resource Management in Wireless and Adhoc mobile networks in the journal of Microprocessors and Microsystems Vol 28/8, 2004 pp 439-445 ".*

Reliability modeling in Sensor Networks, *Bein D, Jolly V, Kumar B, Latifi S, International Journal of Information Technology Special Issue, Volume 11, no. 2, 2005 pp 1-8*

Data Dissemination with Resource management in Wireless Sensor Networks, *V Jolly, N Kimura, S Latifi, Proceedings of third IASTED International Conference on Communications And Computer Networks-CCN 2005 Oct 24-26, 2005 Marina del Rey, CA, USA, Vol 1, pp-12-17.*

Resource Management Schemes in Wireless Sensor Networks, *Vasu Jolly, Naoto Kimura and Shahram Latifi, Information Systems: New Generations - ISNG 2005, Las Vegas 2005*

Reliability Issues in multimodal fusion sensors, *Vasu Jolly, Bein D, Kumar B, Latifi S, Intl Conference on Parallel and Distributed Processing Techniques and Applications PDPTA 2004*

Submissions

Energy-efficient routing in Wireless Sensor Networks based on Data Reduction, *N Kimura, V Jolly and S Latifi, Journal of Combinatorial Optimization.*

Energy Restrained Data Dissemination in Wireless Sensor Networks, *Vasu Jolly, N Kimura, S Latifi, Submitted to International Journal of Distributed Sensor Networks*