SECURE ENERGY-EFFICIENT CLUSTER-BASED ROUTING IN WIRELESS SENSOR NETWORKS

By

REGINA NAA DEDEI CRABBE

BSc. COMPUTER SCIENCE (HONS.)

A Thesis submitted to the Department of Computer Science, Kwame Nkrumah University of Science and Technology in partial fulfillment of the requirements for the degree of

MASTER OF PHILOSOPHY IN INFORMATION TECHNOLOGY

College of Science

June 2012

DECLARATION

I hereby declare that this submission is my own work towards the MPhil and that, to the best of my knowledge, it contains no material previously published by another person nor material which has been accepted for the award of any other degree of the University, except where due acknowledgement has been made in the text.

Regina Naa Dedei Crabbe & 20135391 .....…… ……

Student Name & ID Signature Date

Certified by:

Dr. M. Asante .....…… ……

Supervisor Signature Date

Mr. E. Ofori Oppong .....…… ……

Supervisor Signature Date

Certified by:

Dr. J. B. Heyfron-Acquah .....…… ……

Head of Department Signature Date

ABSTRACT

Wireless Sensor Networks are increasingly used for various applications, such as home monitoring, environmental monitoring, health monitoring, heavy industrial monitoring, and military monitoring. Sensors are low cost tiny devices with limited storage, computational capability and power. Since numerous sensors are usually deployed on remote and inaccessible places, the deployment and maintenance should be easy and scalable.

This study proposes a Secure Energy-efficient Cluster-based Data Routing Protocol for Wireless Sensor Networks based on Symmetric Key Cryptography. The network is divided into a number of clusters and a selected cluster head set within each cluster. The cluster head-set members are responsible for control and management of the network. On rotation basis, a cluster head-set member receives data from its cluster nodes and transmits the aggregated results to the distant base station. Communication between sensor and the sink takes place in four levels; sensor → cluster head → gateway → sink. Encryption of the sensed data is transmitted to the cluster head, which aggregated the data received from the sensor nodes of its cluster before forwarding to the next cluster head on the path or through its gateway to the sink or base station. Sensors do not participate in the routing scheme; their energy is conserved at each sensor node.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS AND ACCRONYMS

WSNs	Wireless Sensor Networks
LEACH	Low-Energy Adaptive Clustering Hierarchy
SPIN	Secure Positioning for Sensor Networks
WSNSF	Wireless Sensor Networks Security Framework
NS-2	Network Simulator Version -2
DSPs	Digital Signal Processors
RF	Radio Frequency
QoS	Quality of Service
RAM	Random Access Memory
MANETs	Mo-bile Adhoc Networks
CH	Cluster Head
TAG	Tiny Aggregation Approach
SQL	Database Query Language
BS	Base Station
S	Sink
PDDD	Pseudo-Distance Data Dissemination
POG	Partial Ordered Graph
TOG	Totally Ordered Graph
E-Span	Energy-aware Spanning Tree Algorithm
MLDA	Maximum Lifetime Data Aggregation
TPC	Two-Phase Clustering
MAC	Message Authentication Code
DES	Data Encryption Standard
3DES	Triple DES

RC6	Rivest Cipher Version -6
CID	Cluster ID
AES	Advanced Encryption Standard
LEAP	Lightweight Extensible Authentication Protocol
PIKE	Peer Intermediaries for Key Establishment in Sensor Networks
μTESLA	Micro Version of the Timed, Efficient, Streaming, Loss-tolerant Authentication
TRANS	Trust Routing for Location Aware Sensor Networks
DoS	Denial of Service
SSKS	Secure Symmetric-session Based Key Scheme
SECDRA	Secure Energy- efficient Cluster-based Data Routing Algorithm
SN	Sensor Node
GN	Gateway Node
CS	Cluster head Set
WSGNs	Wireless Sensor Gateway Networks
SECDRP	Secure Energy- efficient Cluster-based Data Routing Protocol
CBR	Constant Bit Rate
MATLAB	Matrix Laboratory
CPU	Central Processing Unit
TDMA	Time Division Multiple Access
ADCs	Analog to Digital Converters
DD	Directed Diffusion

AVG	Average
MIN	Minimum
MAX	Maximum
SUM	Summation
MHz	Megahertz

DEDICATION

This thesis is dedicated to my parents Mr. & Mrs. Samuel Alexander Leon Nii Daki Crabbe for the unique role they continue to play in my educational exploits and social life to the Glory of God.

ACKNOWLEDGEMENTS

I am most grateful to the Almighty God for his divine guidance and gift of life which have successfully taken me through this course.

I am thankful to my supervisor, Dr. M. Asante for his sage advice, insightful criticisms and suggestions, encouragement, and direction which contributed to the successful writing of this thesis.

I owe my lecturers at the Computer Science Department of Kwame Nkrumah University of Science and Technology, especially Mr. Oppong my deepest gratitude and appreciation for the knowledge imparted to me.

To Mr. Philip Peter Andoh of Ghana Immigration Service (GIS) I say thank you for your encouragement, love and support.

I am forever grateful to my God-father, Mr. Daniel Obuobi former head of Computer Science Department, UCC, for his care, love and keen interest he continues to show in my life.

Lastly, I would like to thank my siblings, Benedict Crabbe, Dorothy Crabbe, Reginald Ffoulkes Crabbe and Peggy Crabbe for their support, unfailing love and kindness which have brought me this far.

May God bless you all

CHAPTER ONE

1.1 INTRODUCTION

Over the past years wireless communication has become of such fundamental importance that a world without it is no longer imaginable for many of us. According to Culler and Hong (2004), Wireless Sensor Networks (WSNs) are emerging as both an important new tier in the IT (Information Technology) ecosystem and a rich domain of active research involving hardware and system design, networking, distributed algorithms, programming models, data management, security and social factors.

A sensor network consists of thousands, even millions of tiny devices equipped with signal processing circuits, microcontrollers, and wireless transmitters/receivers, in addition to embedded sensors. Nodes are randomly and densely deployed over the sensing field, leading therefore to a need for auto-organization capability. Today's sensors can monitor temperature, pressure, humidity, soil makeup, vehicular movement, noise levels, lighting conditions, the presence or absence of certain kinds of objects or substances, mechanical stress levels on attached objects, and other properties (Pathan et al., 2006).

According to Sinha and Chandrakasan (2001), the key limitations of wireless sensor networks are storage, power and processing. The key challenge in sensor networks is to maximize the lifetime of sensor nodes due to the fact that it is not feasible to replace the batteries of thousands of sensor nodes. Therefore, computational operations of nodes and communication protocols must be made as energy efficient as possible. Among these protocols data transmission protocols have much more importance in terms of energy, since the energy required for data

transmission takes 70% of the total energy consumption of a wireless sensor network (Kumar & Chee-Yee, 2003).

Security is an inseparable part of any system. Different people have defined security in different ways. Wikipedia (2011) states that security is the condition of being protected against danger or loss. The US Department of Commerce defines security as a condition that results from the establishment and maintenance of protective measures that ensure a state of inviolability from hostile acts or influences. It is a common fact that security is always of great importance and due to the wireless nature of communication in sensor network applications, various security threats may occur. These threats and attacks could pose serious problems to the wireless sensor devices. Concerns regarding security and privacy have been raised by some authors such as Hanna & Hailes (2011). They have emphasized that if the issues associated with security and privacy are not honestly debated in a reasoned and open ways, there is a risk that, there will be a public backlash which will result in mistrust and consequently the technology will not be used for the many valuable applications where it can provide significant benefits.

1.2 RESEARCH FIELD AND SUBJECT OF STUDY

The study focuses on wireless sensor networks. The study investigates the cluster-based data routing in wireless sensor networks. It further researches on secure energy-efficient cluster-based data routing protocols in wireless sensor networks.

1.3 RESEARCH OBJECTIVES

The main objective of this study is to propose a framework to establish Secure Energy-efficient Cluster-based Data Routing in wireless sensor network.

The specific objectives of the study are to:

- Find out the data routing issues in Wireless Sensor Networks (WSNs).
- Find out the existing data aggregation approaches and protocols in Wireless Sensor Networks (WSNs).
- Establish secure energy-efficient cluster-based data routing from source to sink so that data can be transmitted in a more secured manner with less energy consumption.
- Propose a secure energy-efficient protocol that performs secure data routing using hierarchical clustering and a cryptographic algorithm, with resource rich dynamic cluster head.

1.4 RESEARCH PROBLEM AND RESEARCH QUESTIONS

1.4.1 RESEARCH PROBLEM

In recent times, wireless sensor networks have drawn a lot of attention due to their broad application potentials. Sensor nodes in the network are characterized by severely constrained energy resources and communicational capabilities. Sensor networks aggravate the security and privacy problems because they make large volumes of information easily available through remote access. Hence, adversaries need not be physically present to maintain surveillance. Due to limited capabilities of sensor nodes which are storage, power and processing, providing security and privacy against these attacks are challenging issues to sensor networks.

These security and privacy issues have led to breaches of applications in Wireless Sensor Networks in corporate organizations and homes among others. This

according to Kargl et al., 2008, has led to life threatening situations such as legal issues, attacks in health monitoring in more detailed manner viz. eavesdropping on medical data, modification of medical data, forging of alarms on medical data, denial of service, location tracking of users, activity tracking of users, physical tampering with devices and jamming attacks.

1.4.2 RESEARCH QUESTIONS

- What are the data routing issues in Wireless Sensor Networks (WSNs)?
- What are the existing data aggregation approaches and protocols in Wireless Sensor Networks (WSNs)?
- What are the existing data routing algorithms in Wireless Sensor Networks (WSNs)?
- What are the existing secure energy-efficient algorithms that perform secure data routing using hierarchical clustering and a cryptographic algorithm in WSNs?

1.5 RESEARCH BACKGROUND AND JUSTIFICATION

1.5.1 BACKGROUND TO THE RESEARCH

1.5.1.1 Wireless Sensor Network

Wireless sensor networks are potentially one of the most important technologies of this century. Recent advancement in wireless communications and electronics has enabled the development of low-cost, low-power, multifunctional miniature devices for use in remote sensing applications. The combination of these factors has improved the viability of utilizing a sensor network consisting of a large

number of intelligent sensors, enabling the collection, processing analysis and dissemination of valuable information gathered in a variety of environments.

A WSN is usually composed of hundreds or thousands of sensor nodes. These sensor nodes are often densely deployed in a sensor field and have the capability to collect data and route data back to a base station (BS). A sensor consists of four basic parts: a sensing unit, a processing unit, a transceiver unit, and a power unit (Akyildiz et al., 2002). It may also have additional application dependent components such as a location finding system, power generator, and mobilisers (Fig. 1.1). Sensing units are usually composed of two subunits: sensors and analog-to-digital converters (ADCs). The ADCs convert the analog signals produced by the sensors to digital signals based on the observed phenomenon. The processing unit, which is generally associated with a small storage unit, manages the procedures that make the sensor node collaborate with the other nodes.

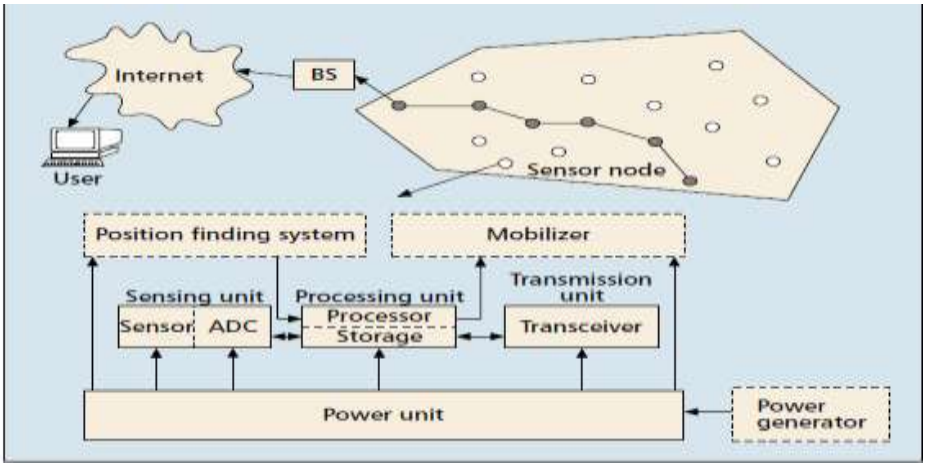


Figure 1.1: The Component of a Sensor Node (Akyildiz et al., 2002)

A transceiver unit connects the node to the network. One of the most important units is the power unit. A power unit may be finite (e.g., a single battery) or may be supported by power scavenging devices (e.g., solar cells). Most of the

sensor network routing techniques and sensing tasks require knowledge of location, which is provided by a location finding system. Finally, a mobiliser may sometimes be needed to move the sensor node, depending on the application.

1.5.1.2 Protocol Stack in Sensor Nodes

The protocol stack used in sensor nodes contains physical, data link, network, transport, and application layers defined as follows (Akyildiz et al., 2002):

- Physical layer: responsible for frequency selection, carrier frequency generation, signal deflection, modulation, and data encryption.
- Data link layer: responsible for the multiplexing of data streams, data frame detection, medium access, and error control; as well as ensuring reliable point-to-point and point-to-multipoint connections.
- Network layer: responsible for specifying the assignment of addresses and how packets are forwarded.
- Transport layer: responsible for specifying how the reliable transport of packets will take place.
- Application layer: responsible for specifying how the data are requested and provided for both individual sensor nodes and interactions with the end user.

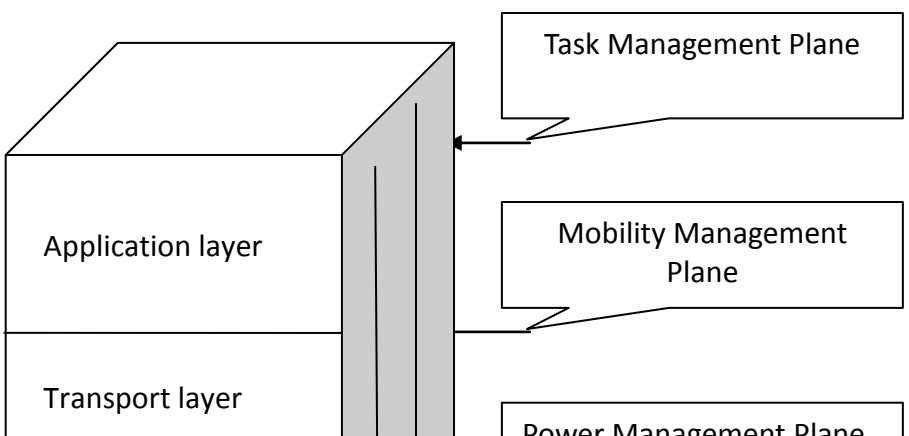


Figure 1.2: The Protocol Stack in Sensor Nodes (Akyildiz et al., 2002)

1.5.1.3 Application Potentials of WSNs

Depending upon the requirement and characteristics of system, wide variety of applications are there which require constant monitoring and detection of specific event.

- Military Applications: Sensor networks are applied very successfully in the military sensing. WSN can be an integral part of military command, control, communications, computing, intelligence, surveillance, reconnaissance and targeting systems, detection of mass destruction and explosion and enemy movement, biological, nuclear and chemical attack detection reconnaissance and military situation awareness.
- Environmental Applications: Nowadays sensor networks are also widely applied in habitat monitoring, agriculture research such as sensing of pesticide, soil moisture, PH levels, habitat exploration of animals, bush burning and flood detection, traffic control and ocean monitoring which includes monitoring of fishes.

- Structural health monitoring: Health monitoring is a very hot research topic for industry and academia. The amount of raw data that can be gathered and transported for such application is of the order of 1-10 Mbps. Thus only useful information is transmitted by using complex algorithms like wavelet transformation, auto regressive models et cetera.
- Heavy industrial monitoring: Industrial applications require highly reliable operation in harsh environment, in warehousing, industrial applications, manufacturing monitoring, industrial automation and factory process control.
- Health or Medical Applications: Sensor networks are also widely used in health care delivery such as monitoring of patient's physiological data, blood pressure or heart beat rate, drug administration control, unconsciousness detection, exercise monitoring and noninvasive health monitoring.
- Home Application: In home application, sensor node can be embedded into furniture and home appliances, monitoring of product quality, managing and monitoring inventory system and automatically controls the temperature and airflow of the room (Toh, 2002).

1.5.1.4 Design Challenges of WSNs

Wireless sensor network uses a wide variety of application and to impact these applications in real world environments, we need more efficient protocols and algorithms. Designing a new protocol or algorithm addresses some challenges which are needed to be clearly understood (Kasimoglui & Akyildiz, 2004). These challenges are summarized below:

- Scalable and flexible architecture: In the sensor network the number of sensor nodes deployed may be in order of hundreds, thousands or millions so that we

can easily extend the network size. The communication protocols must be designed in such a manner that deploying many nodes in the network does not affect clustering and routing. In other words, the network must preserve its stability. Introducing more nodes into the network means that additional communication messages will be exchanged, so that these nodes are integrated into the existing network. The density of network can be calculated as:-

$$\mu(R) = (N \times R^2)/A$$

Where

N = number of scattered sensors node in the region A

R = radio transmission range

$\mu(R)$ = number of nodes within the transmission radius of each node

- Error-prone wireless medium: Since sensor networks can be deployed in different situations, the requirements of each different application may vary significantly. It should be consider that the wireless medium can be greatly affected by noisy environments. An attacker interferes knowingly and causes enough noise to affect the communication.
- Fault tolerance and adaptability: Fault tolerance means to maintain sensor network functionalities without any interruption due to failure of sensor node. This is because in sensor network every node has limited power of energy and so the failure of single node does not affect the overall task of the sensor network. Adaptable protocols can establish new links in case of node failure or link congestion. Network can be able to adapt by changing its connectivity in case of any fault. In that case, well-efficient routing algorithm is applied to change the overall configuration of network.

- Node Deployment: Sensor network can be deployed randomly in geographical area. After deployment, they can be maintained automatically without human presence. In sensor network, node deployment falls into two categories which is either a dense deployment or a sparse deployment. In dense deployment we have relatively high number of sensor nodes in the targeted field whiles in a sparse deployment we have fewer nodes and it is used when the cost of sensor nodes increases and prohibits the use of dense deployment. The dense deployment is used when it is important to detect every moment or when there are multiple sensors for covering an area.
- Real-time: Achieving Real-time in WSN is difficult to maintain. It must support maximum bandwidth, minimum delay and several Quality of Service (QoS) parameters. This issue can affect time synchronization algorithm.
- Power Consumption: Wireless sensor node is a microelectronic device. It means it is equipped with a limited number of power source. Nodes are dependent on battery for their power. Hence power conservation and power management are important issues in wireless sensor network. Due to this reason researchers are focusing on the design of power aware protocols and algorithm for sensor networks.
- Short Range Transmission: WSNs consider the short transmission range in order to reduce the possibility of being eavesdropped. As in long range transmission we need high transmission power due to the point to point transmission between the nodes to reach the destination which increases the chance of being eavesdropped.
- Hardware design: Energy-efficiency should be considered seriously when designing any hardware of sensor network. Hardware such as micro-controller,

power control, and communication unit should be designed to consume less energy.

- Limited computational power and memory size: These factors affect WSN in the sense that each node stores the data individually and sometimes more than one node store same data and transfer the data to the base station where processing of data take place. This wastes the power and the storing capacity of nodes. Therefore effective routing schemes and protocols must be developed to eliminate data redundancy and maximize the network lifetime.
- Operating environment: Sensor nodes are deployed densely, either very close or inside the phenomenon which is to be observed. These nodes may work under-busy interaction, at the bottom of an ocean, in the interior of a large machinery, on the surface of an ocean during a tornado, in a home or large building and in large warehouse.
- Simplicity: Simplicity is an important point in the wireless sensor network. Sensor nodes are small in nature and for this reason there is restriction on the utilization of energy as they are energy dependent. Therefore the computing and communicating software used in the nodes should be computation efficient and less in size than the traditional software in the network.
- Quality of Service (QoS): It means data should be delivered within a period of time. Some real time sensor applications are based on time and it means that, if data should not be delivered on time from the moment it is sensed; the data will become unusable, e.g. fire detection requires good quality of service.
- Unattended operation: In wireless sensor network, nodes are deployed randomly, without any topology. Once these nodes are deployed they do not require human intervention. Hence the nodes are responsible for reconfiguration in case of any

modification that is, addition of new nodes or failure of any node. Nodes are independent of each other so the maintenance needs to be autonomous.

- System Architecture: There is no stable, unified and mature networking and system architecture to build different applications. Most of the applications and research prototypes are integrated in order to maximise performance.
- Security: Security is a very important parameter in sensor network since sensor networks are data centric. This means that, there is no particular id associated with sensor nodes and for this reason an attacker can easily insert himself into the network and steal the important data by becoming part of network without the knowledge of sensor nodes of the network. This makes it difficult to identify whether the information is authenticated or not (Sharma, Tyagi & Bhadana, 2010).

1.5.1.5 Wireless Sensor Networks vs. Traditional Wireless Networks

There are many existing protocols, techniques and concepts from traditional wireless network, such as cellular network, mobile ad-hoc network, wireless local area network and Bluetooth are applicable and still used in wireless sensor network. There are also many fundamental differences which led to the need for new protocols and techniques (Garg et al., 2006). Some of the most important characteristic differences are summarized below:

• Number of nodes in wireless sensor network is much higher than any traditional wireless network. Possibly a sensor network has to scale the number of nodes to thousands. Moreover a sensor network might need to extend the monitored area and has to increase the number of nodes from time to time. This needs a highly scalable solution to ensure sensor network operations without any problem.

• Due to large number of sensor nodes, addresses are not assigned to the sensor nodes. Sensor networks are not address-centric; instead they are data-centric network. Operations in sensor networks are centered on data instead of individual sensor node. As a result sensor nodes require collaborative efforts.

• Sensor nodes mainly use a broadcast communication paradigm, whereas most ad hoc networks are on point-to-point communications.

• Sensor nodes are much cheaper than nodes in ad-hoc networks.

• Wireless sensor networks are environment-driven. While data is generated by humans in traditional networks, the sensor network generates data when the environment changes. As a result the traffic pattern changes dramatically from time to time. Sensor networks are mainly used to collect information while MANETs (Mobile Ad-hoc Networks) are designed for distributed computing rather than information gathering.

• A unique characteristic of wireless sensor network is the correlated data problem. Data collected by neighboring sensor nodes are often quite similar which make it possible to the development of routing and aggregation techniques that can reduce redundancy and improve energy efficiency. It has also been observed that the environmental quantities changes very slowly and some consecutive readings sense temporally correlated data. This advantageous feature can be exploited to develop an energy efficient data gathering and aggregation techniques. Thus, unlike traditional networks, where the focus is on maximizing channel throughput or minimizing node deployment, the major consideration in a sensor network is to extend the system lifetime as well as the system security.

1.5.2 JUSTIFICATION OF THE RESEARCH

It is a common fact that security is always of great importance in all human endeavour and due to the wireless nature of communication in sensor networks applications, various security threats may occur. These threats and attacks could pose serious problems to people using the wireless sensor devices. This study seeks to establish these threats and attacks and counter security measures in wireless sensor networks, and propose a framework for the purpose of establishing secure routing in WSNs.

This study may be useful to corporate organisations that use Wireless Sensor Networks in that the findings might help improve upon secure data routing in WSNs. The findings might also be useful to developers and/or manufacturers to improve upon the wireless sensor devices. It is also envisaged that the study findings might enhance teaching and research activities in the educational sector and other related fields.

1.6 SUMMARY AND PRESENTATION OF THESIS

The study is categorised into five chapters as follows:

Chapter One deals with the introduction which highlights among others, the Research Field and Subject of Study, Research Objectives, Research Problem and Research Questions, Research Background and Justification, and Summary and Presentation of Thesis. Chapter Two focuses on the literature review relating to the study.

Chapter Three introduces and describes the new proposed protocol for data routing in cluster-based wireless sensor networks. Chapter Four presents the performance analysis of the proposed protocol. It also provides the comparison results, discussions

and findings. Finally, Conclusions and Recommendations are given in Chapter Five and the scope of future enhancements is also incorporated.

CHAPTER TWO

REVIEW OF LITERATURE

2.1 Introduction

Advancement in sensor technology has led to the production of wireless sensors which are capable of sensing and reporting various real-world phenomena in a time sensitive manner. A WSN is made up of a large number of low cost sensor nodes with processing and communication capabilities. Even though sensors are small devices with limited power supply, a WSN should operate autonomously for a long period of time in most applications. In order to better manage energy consumption and increase the whole network lifetime, suitable solutions are required at all layers of the networking protocol stack. According to Heinzelman, et al., (2000), energy aware routing protocols at the network layer have received a great deal of attention since it is well established that, wireless communication is the major source of energy consumption in WSN.

This study focuses on data-routing problems in energy constrained sensor networks. The main goal of data-routing algorithms is to gather and aggregate data in an energy efficient manner so that network lifetime is enhanced. Data routing techniques can significantly help to conserve the limited energy resource by eliminating data redundancy and minimising the number of data transmission. For this reason, data routing techniques in WSNs are broadly investigated in this literature. This chapter presents a survey of data-routing algorithms and some security related parameters in wireless sensor networks.

2.2 In-Network Data Aggregation

Fasolo et al. (2007) defined the in-network aggregation process as follows: ”In-network aggregation is the global process of gathering and routing information through a multi-hop network, processing data at intermediate nodes with the objective of reducing resource consumption (in particular energy), thereby increasing network lifetime.” In-network aggregation deals with this distributed processing of data within the network. In this scheme, the sensor network is divided into pre-defined set of regions. Each region is responsible for sensing and reporting events that occurs inside the region to the sink node. In a typical sensor network scenario, different nodes collect data from the environment and then send it to some central node or sink which analyzes and processes the data and then send it to the application. In the in-Network data aggregation, data produced by different node can be jointly processed and forwarded to the sink node.

There are two approaches for in-network aggregation which are

- With size reduction
- Without size reduction.

With size reduction

In-network aggregation with size reduction refers to the process of combining and compressing the data packets received by a node from its neighbors in order to reduce the packet length to be transmitted or forwarded towards a sink or base station. As an example, consider the situation when a node receives two packets which have a spatial correlated data. In this case it is worthless to send both packets. Instead of that, one should apply any function like Average (AVG), Maximum (MAX), and Minimum (MIN) and then send a single packet.

Without size reduction

In-network aggregation without size reduction refers to the process of merging data packets received from different neighbors into a single data packet but without processing the value of data. For example, two packets may contain different physical quantities (like temperature and humidity) and they can be merged into a single packet by keeping both values intact but keeping a single header. This approach preserves the value of data and thus transmits more bits in the network but still reduces the overhead by keeping single header. These two approaches depend on many factors like the type of application, data rate, and network characteristics and so on. There is also a trade-off between energy consumption and precision of data for the two approaches.

In in-network aggregation, the sensor with the most critical information aggregates the data packets and sends the fused data to the sink. Each sensor transmits its signal strength to its neighbors. If a neighbor has higher signal strength, the sender stops transmitting the packets. After receiving packets from all their neighbors, the node that has the highest signal strength becomes the data aggregator. The in-network aggregation scheme is best suited for environments where events are highly localized. In Figure 2.1 of the in-network data aggregation scheme, the numbers indicate the signal strengths detected by the sensors. The arrows indicate the exchange of signal strengths between neighboring nodes.

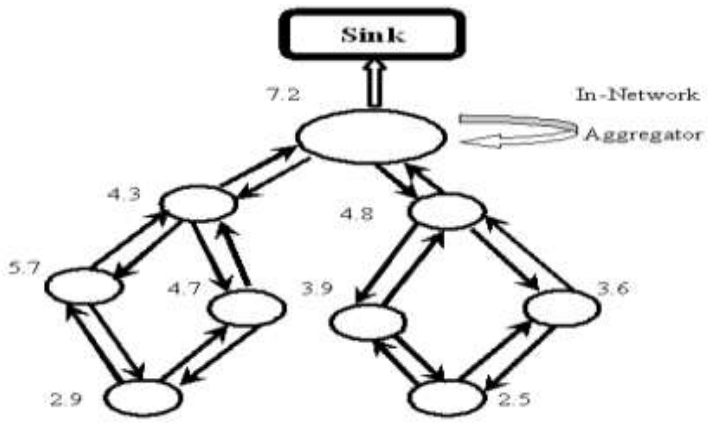


Figure 2.1: In-Network Architecture (Fasolo et al., 2007)

2.2.1 Grid-Based Data Aggregation

Vaidhyanathan et al(2004) have proposed grid based data-aggregation schemes which are based on dividing the regions monitored by a sensor network into several grids. In a grid-based data aggregation, a set of sensors is assigned as data aggregators in fixed regions of the sensor network. The sensors in a particular grid transmit the data directly to the data aggregator of that grid. Hence, the sensors within a grid do not communicate with each other. In a grid-based data aggregation, the data aggregator is fixed in each grid and it aggregates the data from all the sensors within the grid. This is similar to cluster-based data aggregation in which the cluster heads are fixed. Grid based data aggregation is suitable for mobile environments such as military surveillance and weather forecasting and adapts to dynamic changes in the network and event mobility.

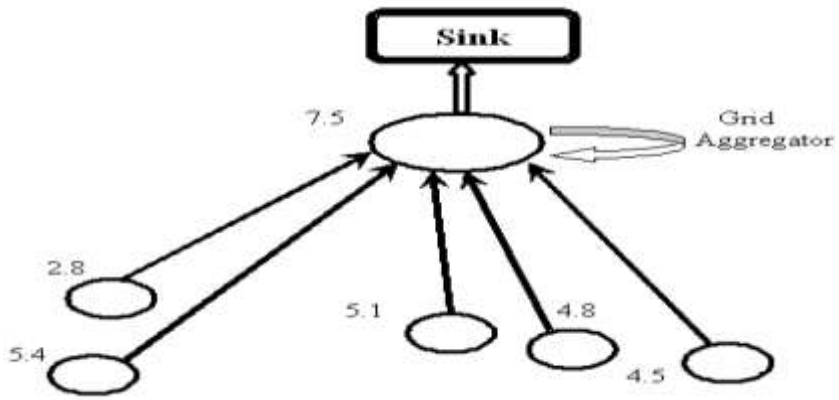


Figure 2.2: A Grid-based Data Aggregation Architecture

(Vaidhyanathan et al., 2004)

The arrows indicate the transmission of data from sensors to the grid aggregator. A typical grid-based data aggregation scheme in Fig 2.2 shows that in a grid-based data aggregation, all sensors directly transmit data to a predetermined grid aggregator. After collecting all data from other sensors, then aggregator which fuses the data sends only the critical information to the sink nodes. Thus, a grid-based scheme reduces the traffic in mobile environment and makes sure the critical data is transmitted to the sink. However, the grid-based scheme does not perform well when events are highly localized and mostly immobile in nature.

2.2.2 Tree-Based Data Aggregation

The simplest way to routing data is to organise the nodes in a hierarchical manner and then select some nodes as the aggregation point or aggregators. The tree-based approach performs aggregation by constructing an aggregation tree (Lee and Wong, 2005), which could be a minimum spanning tree, rooted at sink and source nodes are considered as leaves. Each node has a parent node to forward its data. Flow of data starts from the leaf nodes up to the sink and therein the aggregation is done by parent nodes. The way this approach operates has some drawbacks. As it is known like any wireless network the wireless sensor networks are also not free from failures. In case of packet loss at any level of tree, the data will be lost not only for a single level but for a whole related sub-tree as well. In spite of the high cost of maintaining tree structure in dynamic networks and scarce robustness of the system, this approach is very much suitable for designing optimal aggregation technique and energy-efficient techniques. Madden et al (2002) proposed a data-centric protocol which is based on aggregation trees, known as Tiny Aggregation (TAG) approach. TAG works in two phases: distribution phase and collection phase. In a distribution

phase, TAG organizes nodes into a routing tree rooted at the sink. The tree formation starts with broadcasting a message from the sink at specific level or distance from the root. When a node receives this message it sets its own level to be the level of message plus one and elects the parent as the node from which it receives the message. After that, the node rebroadcast this message with its own level. This process continues until all nodes elect their parent. After tree formation, the sink sends queries to all sensor nodes in the network. TAG uses database query language (SQL) for selection and aggregation functions. In collection phase, data is forwarded and aggregated from the leaf nodes to the root. A parent node has to wait for the data from all its child nodes before it can send its aggregate up the tree. Apart from the simple aggregation function provided by SQL (e.g.: COUNT, MIN, MAX, SUM, and AVG), TAG also partitions aggregators according to the duplicate sensitivity, exemplary and summary, and monotonic properties. Though TAG periodically refreshes the tree structure of network most of the tree-based schemes are inefficient for dynamic network.

Intanagonwiwat, Govindan & Estrin (2000) proposed a reactive data-centric protocol for applications where the sink requested some specific information by flooding, known as directed diffusion paradigm. The directed diffusion paradigm combines data coming from different sources and en-route them by eliminating redundancy, minimizing the number of data transmissions; thus maximizing network lifetime. Directed diffusion consists of several elements: interests, data messages, gradients, and reinforcements.

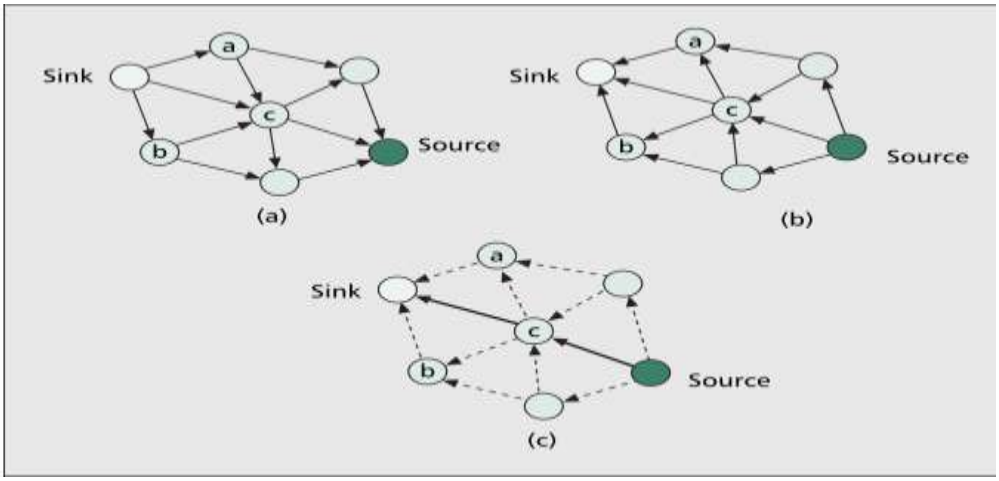


Figure 2.3: A Tree-Based Data Routing Architecture (Intanagonwiwat, Govindan & Estrin, 2000)

Figure 2.3 Simplified scheme for directed diffusion. (a) Interest propagation. (b) Initial gradients setup. (c) Data delivery along reinforced path.

The base station (BS) requests for data by broadcasting an interest message which contains a description of a sensing task.

(a) This interest message propagates through the network hop-by-hop and each node also broadcasts interest message to its neighbor. As interest message propagates throughout the network, gradients are setup by every node within the network.

(b) The gradient direction is set towards the neighboring node from which the interest is received. This process continues until gradients are setup from source node to base station. Loops are not checked at this stage but they are removed at a later stage.

(c) After this path, information flow is formed and then best paths are reinforced to prevent further flooding according to a local rule. Data aggregation took place on the way of different paths from different sources to the base station or sink. The base station periodically refreshes and resends the interest message as soon as it starts to

receive data from sources to provide reliability. The problem with directed diffusion is that it may not be applied to applications (e.g. environmental monitoring) that require continuous data delivery to the base station. This is because query driven on demand data model may not help in this regard. Also matching data to queries might require some extra overhead at the sensor nodes. Mobility of sink nodes can also degrade the performance as the path from sources to sinks cannot be updated until the next interest message is flooded throughout the network. If frequent flooding is introduced then also too much overhead of bandwidth and battery power will be introduced. Furthermore, exploratory data follows all possible paths in the network following the gradients which lead to unnecessary communications overhead.

Lee et al (2005) proposed a new low-control-overhead data dissemination scheme, which they called as pseudo-distanced at dissemination (PDDD), for efficiently disseminating data from all the sensor nodes to the mobile sink. Some assumptions have been made. They are:

(1) all source nodes maintain routes to mobile sink node,

(2) no periodically messaging for topological changes due to mobile sink node,

(3) all link are bi-directional and no control messages are lost,

(4) mobile sink nodes have unlimited battery power, so no need to care about battery efficiency of sink node, and

(5) network partitioning is not considered.

Data dissemination process interest message, sensor nodes do not send exploratory data and do not wait for reinforcement message because each sensor node already has routes to the sink node. After getting interest message, adjacent nodes set a parent-child relationship using pseudo-distance of each node and finally, a partial ordered graph (POG) has been built. Optimal data dissemination is achieved in terms

of path length by forwarding packets to a parent node until topology is unchanged. Then each sensor node is assigned a level of a corresponding sink node with pseudo-distance. In order to overcome the shortcoming of POG, the author used totally ordered graph (TOG) in place of POG. The problem identified in this approach is that due to mobility of sink node all sensor nodes have to maintain routes and for any change in topology nodes have to again change route accordingly which led to energy waste.

Lee et al (2005) proposed an energy-aware spanning tree algorithm for data aggregation, referred as E-Span. E-Span is a distributed protocol in which source node that has highest residual energy is chosen as root. Other source nodes choose their parent based on residual energy and distance to the root. The protocol uses configuration message to exchange information of node i.e., residual energy and distance to the root. Each node performs single-hop broadcast operation to send packets. Single-hop broadcast refers to the operation of sending a packet to all single-hop neighbors.

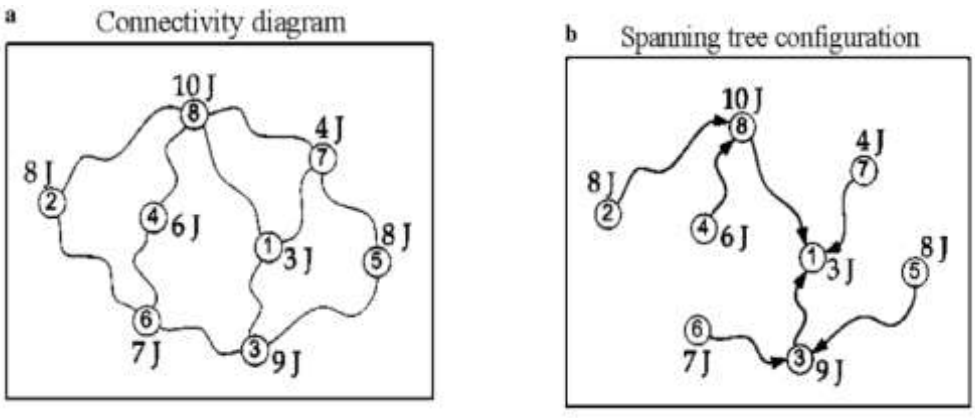


Figure 2.4: E-span Protocol Architecture (Lee et al., 2005)

2.2.3 Cluster-Based Data Aggregation

Another scheme to organise the network in hierarchical manner is cluster-based approach. In cluster-based approach, whole network is divided into several clusters. Each cluster has a cluster-head which is selected among cluster members. Cluster-heads perform the role of aggregator which aggregate data received from cluster members locally and then transmit the result to the sink. The advantages and disadvantages of the cluster-based approaches are very much similar to tree-based approaches.

Dasgupta, Kalpakis & Namjoshi (2003) proposed a maximum lifetime data aggregation (MLDA) algorithm which finds data gathering schedule, provides location for sensors and base station, data packet size, and energy for each sensor. A data gathering schedule specifies how data packets are collected from sensors and transmitted to the base station for each round. A schedule can be thought of as a collection of aggregation trees. They proposed heuristic-greedy clustering-based MLDA based on MLDA algorithm. In this case they partitioned the network into clusters and referred to each cluster as super-sensor. They then computed maximum lifetime schedule for the super-sensors and then used this schedule to construct aggregation trees for the sensors.

Choi and Das (2004) presented a two-phase clustering (TPC) scheme. Phase 1 of this scheme creates clusters with a cluster-head and each node within that cluster forms a direct link with cluster-head. Phase I of this scheme is similar to various schemes used for clustering but differ in a way that the cluster-head rotation is localized and is done based on the remaining energy level of the sensor nodes which minimises time variance of sensors and this leads to energy saving from unnecessary cluster-head rotation. In phase II, each node within the cluster searches for a neighbor closer than the cluster-head which is called data relay point and sets up a

data relay link. Now the sensor nodes within a cluster either uses direct link or data and relay link to send their data to cluster head which is an energy efficient scheme. The data relay point aggregates data at forwarding time to another data relay point or cluster-head. In case of high network density, TPC phase II will set up unnecessary data relay link between neighbors as closely deployed sensor will sense same data and this leads to a waste of energy.

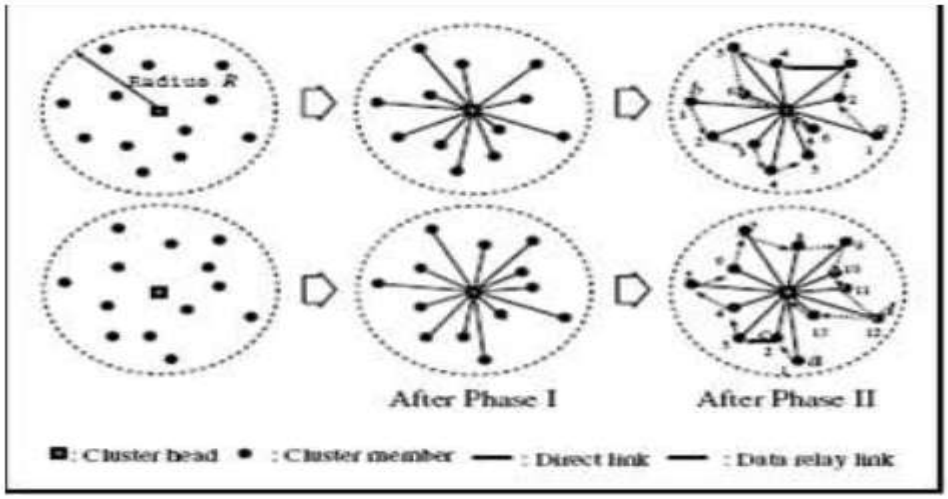


Figure 2.5: Illustration of Two Phase Clustering (Choi and Das, 2004)

Cam et al (2005) present energy efficient and secure pattern based data aggregation protocol which is designed for clustered environment. In the conventional method data is aggregated at a cluster-head and a cluster-head eliminates redundancy by checking the content of data. This protocol is such that instead of sending raw data to cluster-head, the cluster members send corresponding pattern codes to the cluster-head for data aggregation. If multiple nodes send the same pattern code, then only one of them is finally selected to send actual data to the cluster-head. For pattern matching, authors present a pattern comparison algorithm.

2.3 Constraints of WSN

A wireless sensor network is a special network which has many constraints compared to a traditional computer network. Due to these constraints it is difficult to directly employ the existing security approaches to the area of wireless sensor networks. Therefore, to develop useful security mechanisms while borrowing the ideas from the current security techniques, it is necessary to know and understand these constraints first (Carman, Krus, & Matt, 2000).

1. Limited Memory and Storage Space: A sensor is a tiny device with only a small amount of memory and storage space for the code. In order to build an effective security mechanism, it is necessary to limit the code size of the security algorithm. For example, one common sensor type (TelosB) has a 16-bit, 8 MHz RISC CPU with only 10K RAM, 48K program memory, and 1024K flash Storage.

2. Power Limitation: Energy is the biggest constraint that affects wireless sensor capabilities. For when sensor nodes are deployed in a sensor network, they cannot be easily replaced (high operating cost) or recharged (high cost of sensors).

3. Unreliable Communication: Normally the packet-based routing of the sensor network is connectionless and thus inherently unreliable. Packets may get damaged due to channel errors or dropped at highly congested nodes. The result is that, the packets either get lost or missing in the process.

2.4 Security Requirements of a Sensor Network

• Data Confidentiality: In many applications, nodes communicate highly sensitive data, (e.g., key distribution) therefore it is extremely important to build a secure communication channel in a wireless sensor network. The adversary can change the data, so as to send the sensor network into disarray. For example, a malicious node may add some fragments or manipulate the data within a packet.

• Data Freshness**:** Data freshness suggests that the data is current, and it ensures that no old messages have been replayed. This requirement is especially important when there are shared-key strategies employed in the design.

• Self-Organization: A wireless sensor network is typically an ad-hoc network, which requires every sensor node to be independent of the others and flexible enough to be self-organizing and self-healing according to different situations.

• Time Synchronization: sensors may wish to compute the end-to end delay of a packet as it travels between two pairs of wise sensors. A more collaborative sensor network may require group synchronization for tracking applications, etc.

• Secure Localization: A sensor network will rely on its ability to accurately and automatically locate each sensor in the network. A sensor network designed to identify faults will need accurate location information in order to pinpoint the location of a fault. For large sensor networks, the SPINE (Secure Positioning for sensor Networks) algorithm is used. It is a three phase algorithm based on verifiable multilateration.

• Authentication: Data authentication allows a receiver to verify that the data really is sent by the claimed sender. In the case of two-party communication, data authentication can be achieved through a purely symmetric mechanism: the sender and the receiver share a secret key to compute the message authentication code (MAC) of all communicated data.

2.5 Attacks on WSNs

• Denial of Service Attack: "Any event that diminishes or eliminates a network's capacity to perform its expected function" (Wood & Stankovic, 2002).

• Jamming: To jam a node or set of nodes, in this case, is simply the transmission of a radio signal that interferes with the radio frequencies being used by the sensor network.

• The Sybil Attack: Sybil attack is defined as a "malicious device illegitimately taking on multiple identities" (Newsome et al., 2004). It was originally described as an attack able to defeat the redundancy mechanisms of distributed data storage systems in peer-to-peer networks. In addition to defeating distributed data storage systems, the Sybil attack is also effective against routing algorithms, data aggregation, voting, fair resource allocation and foiling misbehavior detection. For instance, in a sensor network voting scheme, the Sybil attack might utilize multiple identities to generate additional "votes."

• Node Replication Attacks: An attacker seeks to add a node to an existing sensor network by copying (replicating) the node ID (Identity) of an existing sensor node (Karlof & Wagner, 2003). A node replicated in this fashion can severely disrupt a sensor network's performance; packets can be corrupted or even misrouted. This can result in a disconnected network, false sensor readings, etc.

• Attacks against Privacy: Monitoring and eavesdropping: Eavesdropping occurs when an adversary listens to the data and easily discovers the communication contents. When the traffic conveys the control information about the sensor network configuration, which contains potentially more detailed information than accessible through the location server, the eavesdropping can act effectively against the privacy protection (Chan, Perrig & Song, 2003).

2.6 Defensive Measures

Key Establishment

Traditionally, key establishment is done using one of the many public-key protocols. One of the more common public-key protocols is the Diffie-Hellman public key protocol, but the protocols are incompatible in low power devices such as wireless sensor networks. This is due largely to the fact that typical key exchange techniques use asymmetric cryptography, also called public key cryptography. In this case, it is necessary to maintain two mathematically related keys, one of which is made public while the other is kept private. This allows data to be encrypted with the public key and decrypted only with the private key. The problem with asymmetric cryptography, in a wireless sensor network, is that it is typically too computationally intensive for the individual nodes in a sensor network. This is true in the general case. (Burman, 2007), (Price & Kosaka, 2004), (Jolly, 2003), (Blaβ, 2006) show that it is feasible with the right selection of algorithms.

Symmetric schemes utilize a single shared key known only between the two communicating hosts. This shared key is used for both encrypting and decrypting data. The traditional example of symmetric cryptography is DES (Data Encryption Standard). The use of DES, however, is quite limited due to the fact that it can be broken relatively easily. Other symmetric cryptography systems have been proposed including 3DES (Triple DES), RC6, AES, and so on (Burman, 2007), (Abdul-Elminaam, Abdul-Kader & Hadhoud, 2008).

Key Establishment and Associated Protocols: -The Lightweight Extensible Authentication Protocol (LEAP) (Jolly, 2003), (Devasenapathy, Suat & Nair, 2003) takes an approach that utilizes multiple keying mechanisms. Their observation is that no single security requirement accurately suites all types of communication in a wireless sensor network. Therefore, four different keys are used depending on whom

the sensor node is communicating with. Sensors are preloaded with an initial key from which further keys can be established.

In Peer Intermediaries for Key Establishment (PIKE) (Chan & Perrig, 2005), a mechanism for establishing a key between two sensor nodes is based on the common trust of a third node somewhere within the sensor network. The nodes and their shared keys are spread over the network such that for any two nodes A and B, there is a node C that shares a key with both A and B. Therefore, the key establishment protocol between A and B can be securely routed through C.

Perrig et al., 2002 proposed a key-chain distribution system for their Micro Version of the Timed, Efficient, Streaming, Loss-tolerant Authentication (µTESLA) secure broadcast protocol (Liu & Ning, 2004). The basic idea of the µTESLA system is to achieve asymmetric cryptography by delaying the disclosure of the symmetric keys. In this case a sender will broadcast a message generated with a secret key. After a certain period of time, the sender will disclose the secret key. The receiver is responsible for buffering the packet until the secret key has been disclosed. After the disclosure, the receiver can authenticate the packet, provided that the packet was received before the key was disclosed.

2.7 Types of Attacks and their Defensive Mechanisms

A. Defending Against Attacks on Routing Protocols

Techniques for Securing the Routing Protocol: - According to Liu and Ning (2004), the Trust Routing for Location Aware Sensor Networks (TRANS) routing protocol is designed to be used in data centric networks. It also makes use of a loose-time synchronization asymmetric cryptographic scheme to ensure message confidentiality. In their implementation, µTESLA is used to ensure message

authentication and confidentiality. Using μTESLA, TRANS is able to ensure that a message is sent along a path of trusted nodes while also using location aware routing. The strategy is for the base station to broadcast an encrypted message to all of its neighbors. Only those neighbors who are trusted will possess the shared key necessary to decrypt the message. The trusted neighbor(s) then adds its location (for the return trip), encrypts the new message with its own shared key and forwards the message to its neighbor that is closest to the destination. Once the message reaches the destination, the recipient is able to authenticate the source (base station) using the MAC (Message Authentication Code) that will correspond to the base station. To acknowledge or reply to the message, the destination node forwards a return message along the same trusted path from which the first message was received (Liu & Ning, 2004).

B. Defending Against DoS (Denial of Service) Attacks

One strategy in defending against the classic jamming attack is to identify the jammed part of the sensor network and effectively route around the unavailable portion. Wood and Stankovic (Newsome et al., 2004) described a two phase approach where the nodes along the perimeter of the jammed region report their status to their neighbors who then collaboratively define the jammed region and simply route around it. To handle jamming at the MAC layer (Medium Access Control Layer), nodes might utilize a MAC admission control that is rate limiting. This would allow the network to ignore those requests designed to exhaust the power reserved for a node.

C. A Wormhole Attack

A malicious node eavesdrops on a packet or series of packets, tunnels them through the sensor network to another malicious node, and then replays the packets.

This can be done to misrepresent the distance between the two colluding nodes. It can also be used to generally disrupt the routing protocol by misleading the neighbor discovery process. Additional hardware, such as a directional antenna (Karlof & Wagner, 2003) is used to defend against wormhole attacks.

Using a visualization approach to identify wormholes, it first computes distance estimation between all neighbor sensors, including possible existing wormholes. Using multi-dimensional scaling, they then compute a virtual layout of the sensor network. A surface smoothing strategy is then used to adjust for round-off errors in the multi-dimensional scaling. Finally, the shape of the resulting virtual network is analyzed. If a wormhole exists within the network, the shape of the virtual network will bend and curve towards the offending nodes. Using this strategy the nodes that participate in the wormhole can be identified and removed from the network. If a network does not contain a wormhole, the virtual network will appear flat (Song, Przydatek & Perrig, 2003).

E. Defending Against the Sybil Attack

The network needs some mechanism to validate that a particular identity is the only identity being held by a given physical node (Newsome et al., 2004). The two methods to validate identities are:

1. Direct Validation: -In direct validation a trusted node directly tests whether the joining identity is valid. Direct validation techniques include a radio resource test. In the radio test, a node assigns each of its neighbors a different channel on which to communicate. The node then randomly chooses a channel and listens to the transmission on the channel. If the node detects a

transmission on the channel it is assumed that the node transmitting on the channel is a physical node. Similarly, if the node does not detect a transmission on the specified channel, the node assumes that the identity assigned to the channel is not a physical identity.

2. Indirect Validation: -In the indirect validation, another trusted node is allowed to vouch for (or against) the validity of a joining node (Song et al., 2004).

F. Detecting Node Replication Attacks

In Karlof and Wagner (2003), Parno, Perrig, & Gligor (2005) described two algorithms:

1. Randomized multicast: - It determines multicast by randomly choosing the witnesses. In the event that a node is replicated, two sets of witness nodes are chosen. Assuming a network of size n, if each node derives pn witnesses then the birthday paradox suggests that there will likely be at least one collision. In the event that a collision is detected, the offending nodes can easily be revoked by propagating a revocation throughout the network. The communication cost of the randomized multicast algorithm is still O (n2) - too high for large networks.
2. Line-selected Multicast: - It is based upon rumor routing (Karlof and Wagner, 2003). The idea is that a location claim traveling from source to destination will also travel through several intermediate nodes. If each of these nodes records the location claim, then the path of the location claim through the network can be thought of as a line segment. In this case, the destination of the location claim is one of the randomly chosen witnesses as the location claim routes through the network towards a witness node; the intermediate

sensors check the claim. If the claim results in an intersection of a line segment then the nodes originating from the conflicting claims are revoked. The line selected multicast algorithm reduces the communication cost to O(npn) as long as each line segment is of length O(pn) nodes. The storage cost of the line-selected multicast algorithm is O(pn).

G. Defending Against Attacks on Sensor Privacy

Anonymity Mechanisms: Anonymity mechanisms depersonalise the data before the data is released, which present an alternative to privacy policy-based access control. Secure communication channel using secure communication protocols, such as SPINS (Secure Positioning for Sensor Networks) (Zhou, 2006), the eavesdropping and active attack scan be prevented.

H. Secure Data Aggregation

An aggregator is responsible for collecting the raw data from a subset of nodes and processing/aggregating the raw data from the nodes into more usable data (Song, Przydatek & Perrig, 2003).

I. Aggregate-commit-prove Technique: - This technique comprises three phases. Aggregate: The aggregator collects data from the sensors and computes the aggregation result according to a specific aggregate function. Each sensor shares a key with the aggregator. This allows the aggregator to verify that the sensor reading is authentic.

II. Commit Phase: The aggregator is responsible for committing to the collected data. This commitment ensures that the aggregator actually uses the data collected from the sensors. One way to perform this commitment is to use a Merkle hash-tree construction. Using this technique the aggregator computes a hash of each input value and the internal nodes are computed as the hash of

their children concatenation. The commitment is the root value. The hashing is used to ensure that the aggregator cannot change any input values after having hashed them.

III. Proving Phase: The aggregator is charged with proving the results to the user. The aggregator first communicates the aggregation result and the commitment. The aggregator then uses an interactive proof to prove the correctness of the results. This requires two steps.

(1) The user/home server checks to ensure that the committed data is a good representation of the data values in the sensor network.

(2) The user/home server decides whether the aggregator is true or not. This can be done by checking whether or not the aggregation result is close to the committed result. The interactive proof differs depending on the aggregation function that is being used.

2.8 Conclusion

In this chapter, the research reviewed the two types of network architecture which are the flat network architecture and hierarchical network architecture. The study further reviewed the types of routing aggregation protocols which are the In-network, Grid-based, Tree-based and Cluster-based routing protocols. The study finally discussed the constraints of sensor security, security requirements in WSNs, attacks on WSNs and their defensive measures.

CHAPTER THREE

WIRELESS SENSOR NETWORK SECURITY FRAMEWORK (WSNSF)

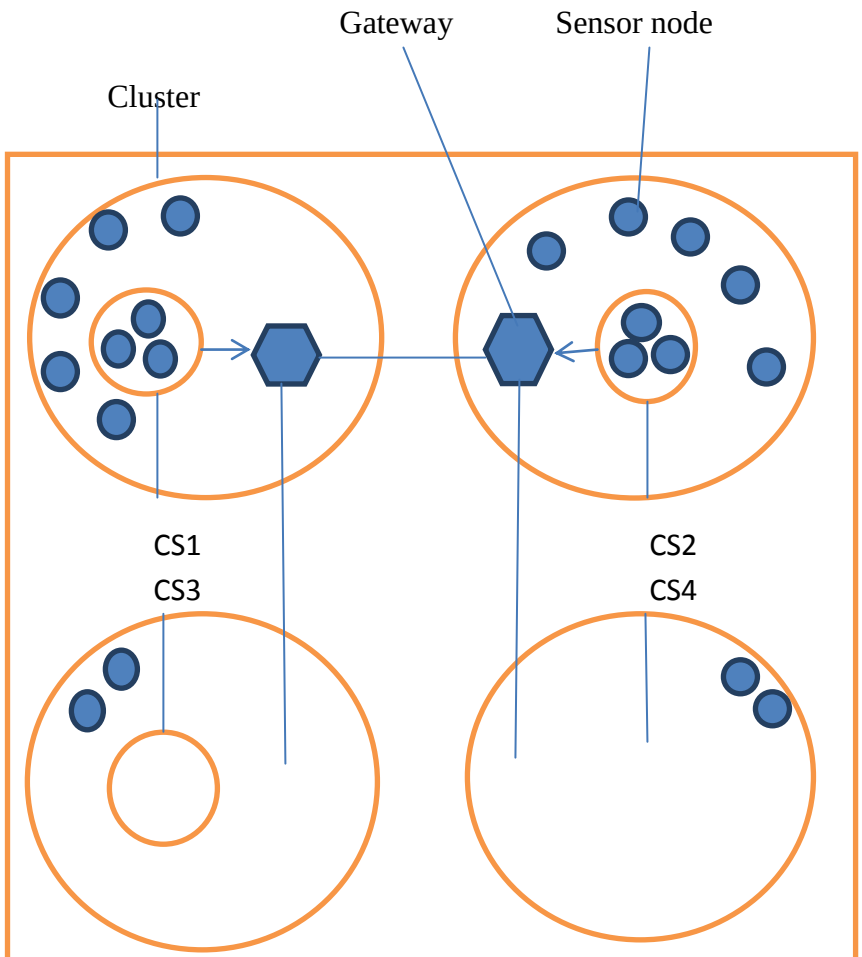
3.1 Introduction

The proposed framework called Wireless Sensor Network Security Framework (WSNSF) consists of three interacting components that can be used to design energy-efficient security protocols that are adaptive to the environment: a Symmetric-Session based Key Scheme (SSKS) (blowfish encryption/decryption), Secure Energy-Efficient Cluster-based Data Routing Algorithm (SECDRA) and Error Detection Mechanism. Each of these components can achieve certain level of

security and energy efficient routing in the wireless sensor networks. WSNSF takes into consideration the communication and computation limitations of sensor networks. While there is always a trade-off between security and performance, experimental results prove that the proposed framework can achieve energy efficient routing and high degree of security with negligible overheads.

3.2 Systems Model

A description of a three-level system model in the wireless sensor network comprising the Sensor Nodes (SN), Cluster head Set (CS) nodes which consist of several sensor nodes, Distributed Gateway Nodes (GN) and Sink (Base Station) are shown in Figure 3.1 The whole network is divided into clusters and each cluster comprises of one CS and a GN that controls several SNs. The four different levels of the WSNs are given below.



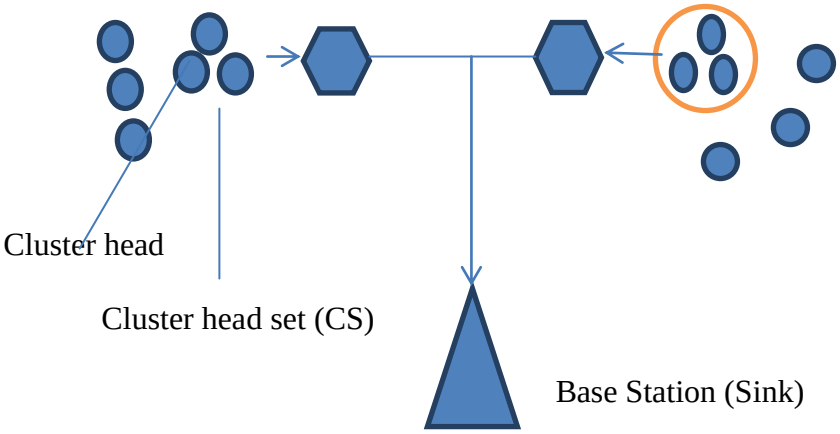


Figure 3.1: Four-level WSNGs Architecture

Based on different hardware constraints and the applications of WSNs, the sensor nodes are classified into three categories which are the generic, special-purpose and the high-bandwidth sensors.

Level-1: These are the set of generic sensor nodes (SN) like Mica Motes (Ding et al., 2004) that are deployed in hundreds or thousands in a specific monitoring area. The whole monitoring area is divided into certain clusters which can be formed based on selected cluster selection algorithms and based on the number and type of sensors for different applications (Khan, Bhargava & Pankaj, 2007), (Sahoo & Sun, 2005) and (Chandrakasan, Heinzelman & Balakrishnan, 2000). Their functions are simple, specific and are usually operated independently. They sense the medium, collect the raw data and forward it to the second level. The hardware specifications of such nodes are shown in Table 3.1.

Table 3.1: Prototype of generic-sensor nodes (Mica Mote)

Processor	8-bit, 4 MHz
Memory	8 KB flash

	512 bytes RAM 512 bytes EEPROM
Radio	916 MHz radio
Data Rate	10 Kbps

Level-2: These are some special-purpose sensor nodes like Spec 2003 (Ding et al., 2004), of which limited number is deployed in the monitoring region. In each cluster, there exists only one cluster head from the cluster head-set which collects raw data from the SNs of its cluster to its Gateway node (GN). These nodes are more powerful in computation and energy than the SNs and their respective prototypes are presented in Table 3.2.

Each Cluster Head (CH) of the network has unique identity (ID) and its assignment is based on the number of clusters. CH can track events or targets using the sensors of its own cluster and prepare the final report using data fusion and aggregation techniques and then forwards the fused data via the GN to the higher level (Sink).

Table 3.2: Prototype of special-purpose sensor nodes (Spec 2003)

Processor	4-8 MHz Custom 8-bit
Memory	0.1 Mb flash memory 3K-4Kb RAM
Radio	50-100Kbps
Data Rate	20 Kbps

Level-3: These are generic sensor nodes (SN) like Mica Motes (Ding et al., 2004) that are deployed in every cluster to forward data from the cluster head to the Sink

for processing. Their functions are simple, specific and are usually operated independently. These types of sensors are used because the gateway nodes do not perform any aggregation and therefore do not require any powerful sensor node like the special purpose sensor node or the high-bandwidth sensor node. The hardware specifications of such nodes are shown in Table 3.1.

Level-4: The high-bandwidth sensing and communication nodes like RSC Wins-Hidra Nodes (Ding et al., 2004) form the fourth level of the network and are known as the Sink of the Wireless Sensor Gateway Networks (WSGNs). According to Ding et al (2004) the operating characteristics of such nodes are illustrated in Table 3.3. These nodes have relatively powerful processing, memory and transmission capacity and have long battery life. These sinks and the user or the controlling center are connected via wireless such as internet and satellite.

Table 3.3: Prototype of high-bandwidth sensing nodes (RSC Wins-Hidra Nodes)

Processor	Intel Strong ARM 1100@133 MHz 150 MIPS
Memory	4 MB Flash memory 1MB SRAM
Radio	3 wire RS-232
Data Rate	100 Kbps

3.3 Definition of Terms

Functions of Sink (Base Station)

1. The sink decrypts the data packet and checks the integrity of the packet
2. Whenever the session expires it generates a new session key
3. It encrypts the new session key with another new generated session key and forwards it to the gateway.

Functions of the Cluster head Set (CS)

1. The Cluster head Set consists of Cluster head nodes which are selected during election phase.
2. It is responsible for transmitting messages to the distant base station.
3. At one time, only one member of the cluster head-set is active and the remaining head-set members are in sleep mode.
4. Its task of transmission to the base station is uniformly distributed among all the cluster head-set members.

Functions of the Cluster Head (CH)

1. Appends logical time stamp and its own ID on received data packets from all single hop to its cluster sensor nodes.
2. Aggregates the data packets by applying the redundancy factor and routes them to sink.

Functions of Gateway (GN)

1. Receives a new session key from Sink
2. It forwards the encrypted data packets to the Sink for processing.
3. Sends control packets (session keys) to its cluster head.

Functions of Sensor Node (SN) or Non-Cluster Head

1. Encrypts the data packet using blowfish algorithms.
2. Sends data packets to its cluster head (CH) node.
3. Receives control packets from its cluster head (CH) node.
4. Updates the session key based on control packets.

3.4 A Symmetric-Session based Key Scheme (SSKS)

In the design every sensor node has a session key at the time of deployments. Initially the sensor node encrypts the sensed data by applying the Blowfish Algorithm, which makes the data transmission more secured. It sends encrypted data to the cluster head and then from the cluster head to the gateway and finally to the sink.

The advantage of this technique is that, it increases communication security and requires very less energy as compared to other cryptography algorithms. After completing a current session, the sink will generate a new session key using a pseudorandom function (f) and current session key and sends it to the corresponding gateway. The new session key is broadcast to its cluster sensors by the CH, for data encryption of the new session. This communication processes of the session key changes dynamically for every session by the sink.

In the propose algorithm, the Constant Bit Rate (CBR) (Blaze, 1996) protocol that is used to provide data authentication is granted by using periodically changing user specific session keys. These session keys are generated from the sink to the cluster Head via its gateway and then the cluster head broadcasts the key to its cluster sensor nodes to be used in the next session.

3.4.1 Blowfish Algorithm

It is a symmetric (i.e. uses the same secret key for both encryption and decryption) block cipher (encrypts data in 8-byte blocks) that uses a variable-length key, from 32 bits (4 bytes) bits to 448 bits (56 bytes). The algorithm consists of two parts: a key expansion part and a data-encryption part. Key expansion converts a variable key of at least 4 and at most 56 bytes into several sub-key arrays totaling 4168 bytes. Blowfish has 16 rounds. Each round consists of a key-dependent permutation, and a key and data-dependent substitution. All operations are XORs and additions on 32-bit words. The only additional operations are four indexed array data lookups per round. Blowfish uses a large number of sub-keys. These keys must be pre-computed before any data encryption or decryption (Schneier, 1993).

Sharma (2009) evaluated the performance of different symmetric cryptographic algorithms and found out that AES algorithm (Tamimi, 2008) is a very fast algorithm but requires at least 800-byte memory space for lookup tables, DES also uses large lookup tables and its throughput is very less hence weak DES, which makes it an insecure block cipher. RC6 (El-Fishawy, 2007) is a small algorithm, but it is slower than blowfish. It was found that, in encryption and decryption Blowfish is better than the other mentioned algorithms in throughput and power consumption. Thus, analysis concluded that Blowfish has better performance than other general encryption algorithms in term of the battery and time consumption.

3.5 Secure Data Routing Algorithm (SECDRA): Routing

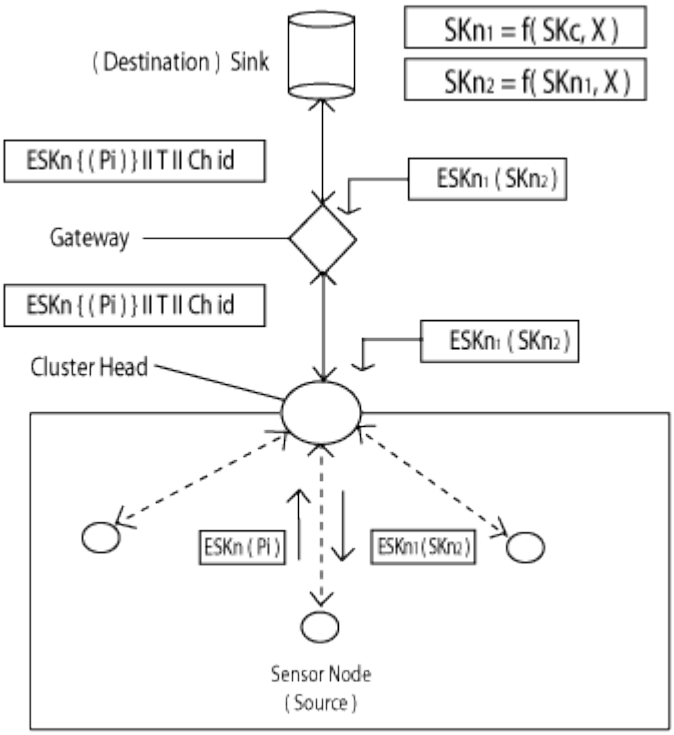


Figure 3.2: Secure transmission in WSNGs

Notation uses in SECDRA

Notation		Description
P*i*	→	Packet
E ( )	→	Encryption function
SK	→	Session Key
S	→	Sink
Ch1, Ch2, Ch3	→	Cluster Head IDs
G	→	Gateway
D ( )	→	Decryption function
f ( )	→	Pseudorandom function
T*GS*	→	Logical time stamp (sequence- numbers) of the gateways

T*SG*	→	Logical time stamp (sequence - numbers) of the Sink
SK*n*	→	New session key
SK*c*	→	Current session key
X	→	Random number
| |	→	Concatenation operator

Routing Algorithm

Start

There are four types of communications in the proposed scheme:-

a) Sensor → Cluster Head

b) Cluster Head → Gateway

c) Gateway → Gateway

d) Gateway → Sink

Secure communication in each of these schemes is explained below

1. Sensor to Cluster Head: - A Sensor node S*i* encrypts the packet P*i* using current session key SK, which is built-in at the time of sensors deployments and sends it to its Cluster Head (Ch).

Si →Ch

E*SK* (P*i*)

2. Cluster Head to Gateway: - The following actions are performed at the Ch:

(i) Ch concatenates the encrypted packets received from the sensor nodes within their respective clusters.

(ii) Increments the value of logical time stamps (T*GS*) by one and appends it to the concatenated packets.

(iii) Concatenates its own ID and sends it to the next cluster head via its gateway on the path to the Sink.

Ch→G

{ESK(Pn)} | |TGS | |Ch_ID | | Ch1 } | |. . . . | | {{(ES1K1(Pd)} | | TGS | |Ch_ID | | Ch2)} | |. . . . | |{{(ES2K2(Px)} | | TGS | |Ch_ID | | Ch3)} where

where SK(Pn) is the encrypted packet from the sensor node n belonging to the cluster, of Cluster head Ch1,

(ES1K1(Pd) | |TGS | |Ch_ID | | Ch2) are encrypted packets received from cluster head Ch2, (ES2K2(Px) | | TGS | |Ch_ID | | Ch3):- are encrypted packet received from the cluster head Ch3.

TGS: - Time stamp belonging to each cluster head.

Ch_ID: - Cluster Head ID

3. Gateway to Gateway: All encrypted packets from the various cluster heads are sent to their respective gateway nodes to forward to the sink. The gateways also allow communication between different cluster nodes.

Gn1→Gn2→Gn3

4. Gateway to Sink: The Gateway sends the concatenated encrypted packets to the sink.

G→Sink

{ESK(Pn)} | | TGS | |Ch_ID | | Gn } | |. . . . | | {{ES1K1(Pd)} | | TGS | |Ch_ID | | Gd}| |. . . | | {{(ES2K2(Px)} | | TGS | |Ch_ID | | Gx)}

The sink processes the received packets before it goes to the end user.

5. The following actions are performed by the sink on receiving packets from the gateway:

(i) For a credible time stamp, sink decrypts the encrypted packets using the current session key, DSK {{ESK(Pn)} | | TGS | |CH_ID | | Gn| |. . . . | | {{(ES1K1(Pd) } | | TGS | |CH_ID | | Gd} | |. . . . | | {{(ES2K2(Px)} | |TGS | |Ch_ID | | Ch3)}

if (TGS ≥TSG), the time stamp is credible and data is authentic (to obtain the original message Px). That is DSK ESK (Px) →Px,

if (TGS ≤TSG), then the sink either discards the packet or sends a retransmission request to CH through its gateway.

(ii) Checks the time stamp credibility first, sink extracts CH ID from packet. For a valid CH ID, it checks the time stamp credibility by comparing the sequence number TGS appended by the CH with the latest value of its logical time stamp TSG,

(iii) Verify CH IDs in the packets.

6. On expiry of current session, sink increments the value of TSG by 1, and generates the new session key using the pseudorandom function (f) and current session key. The new session key is a function of current session and a random number (x).

New Session SKn_1= f(SKc,x) where x is a random number .

7. Another new session key SKn_2 is generated using the pseudorandom function (f) and the new session key SKn_1.

8. Session key is updated for the next session. Session key is updated as follows:

(i) Sink encrypts the new session key(SKn_2) using the new session key(SKn_1) and sends it to the corresponding gateway, $ESKn_1$ (SKn_2)

(ii) Gateway sends the new session key(SKn_2) to the Cluster head to broadcast the new session key(SKn_2) to the sensor nodes in its cluster,

(iii) Sensor nodes update its session keys, with the new session key.

9. End

3.6 Communication stages in a Cluster of a Wireless Sensor Network

A cluster head is a sensor node that transmits an aggregated sensor data to the distant base station. Non-cluster heads are sensor nodes that transmit the collected data to their respective cluster heads. Each cluster has a cluster head-set that consists of several virtual cluster heads; however, only one cluster head-set member is active at one time.

An iteration consists of two stages: an election phase and a data transfer phase (Hussain & Matin, 2005). In an election phase, the cluster head-sets are chosen for the pre-determined number of clusters. In the data transfer phase, the members of the cluster head-set transmit aggregated data to the base station. Each data transfer phase consists of several epochs or sessions. Each member of a cluster head-set becomes a cluster head once during an epoch. A round consists of several iterations. In one round, each sensor node becomes a member of the cluster head-set for one time. Figure 3.3 shows the communication stages.

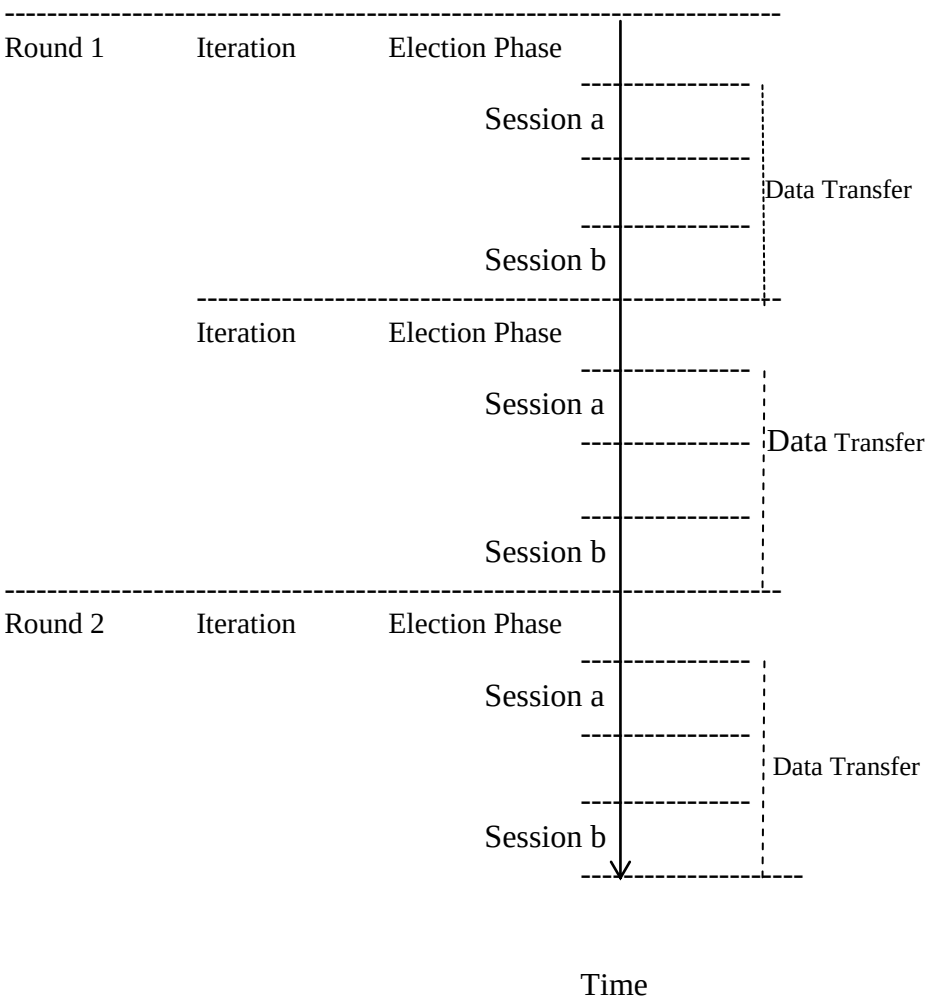


Figure 3.3: Communication stages in a cluster of a wireless sensor network (Hussain & Matin, 2005)

3.6.1 Election Phase

In the proposed model, the number of clusters, c, is pre-determined for the wireless sensor network. At the start, a set of cluster heads are chosen on random basis. These cluster heads send a short range advertisement broadcast message. The sensor nodes receive the advertisements and choose their cluster heads based on the signal strengths of the advertisement messages. Each sensor node sends an acknowledgment message to its cluster head. Moreover, for each iteration, the cluster heads choose a set of associates based on the signal analysis of the acknowledgments.

A cluster head-set consists of a cluster head and the associates. The cluster head-set, which is responsible to send messages to the base station, is chosen for one iteration of a round. In an epoch or session of an iteration, each member of the cluster head-set becomes a cluster head. All the cluster head-set members share the same time slot to transmit their frames. Based on uniform rotation, a schedule is created for the cluster head-set members for their frame transmissions; only the active cluster head transmits a frame to the base station. Moreover, a schedule is created for the data acquisition and data transfer time intervals for the sensor nodes that are not members of the cluster head-set.

3.6.2 Data Transfer Phase

Once clusters, cluster head-sets, and TDMA-based schedules are formed, data transmission begins. The non-cluster head nodes collect the sensor data and transmit the data to the cluster head, in their allotted time slots. The cluster head node must keep its radio turned on to receive the data from the nodes in the cluster. The associate members of the cluster head-set remain in the sleep mode and do not receive any messages. After, some pre-determined time interval, the next associate becomes a cluster head and the current cluster head becomes a passive cluster head-set member.

At the end of an epoch or session, all the cluster head-set members become a cluster head for once. There can be several epochs in an iteration. At the end of an iteration, the cluster head-set members become non-candidate members and a new cluster head-set is chosen for the next iteration. Finally, at the end of a round, all the nodes become non-candidate members. At this stage, a new round is started and all the nodes become candidate members.

3.6.3 States of a Sensor Node

Different states of a sensor node in a wireless sensor network are shown in Figure 3.4. The damaged or malfunctioning sensor states are not considered. Each sensor node joins the network as a *candidate*. At the start of each iteration, a fixed number of sensor nodes is chosen as cluster heads; these chosen cluster heads acquire the *active* state. By the end of the election phase, a few nodes are selected as members of the cluster head-sets; these nodes acquire *associate* state. At the end of an election phase, one member of a cluster head-set is in the active state and the remaining cluster head-set members are in associate state.

In a session of a data transfer stage, the active sensor node transmits a frame to the base station and goes into the *passive associate* state. Moreover, the associate, which is the next in the schedule to transmit to the base station, acquires the active state. During a session or an epoch, the cluster head-set members are distributed as follows: one member is in active state, a few members are in associate state, and a few members are in passive associate state. During the transmission of the last frame of a session (epoch), one member is active and the remaining members are passive associates; there is no member in an associate state. Then, at the start of the next session, all the cluster head-set members become associates and one of them is chosen to acquire the active state.

At the end of an iteration, all the cluster head-set members acquire the *non-candidate* state. The members in non-candidate state are not eligible to become a member of a cluster head-set. At the start of a new round, all non-candidate sensor nodes acquire candidate state; a new round starts when all the nodes acquire non-candidate state.

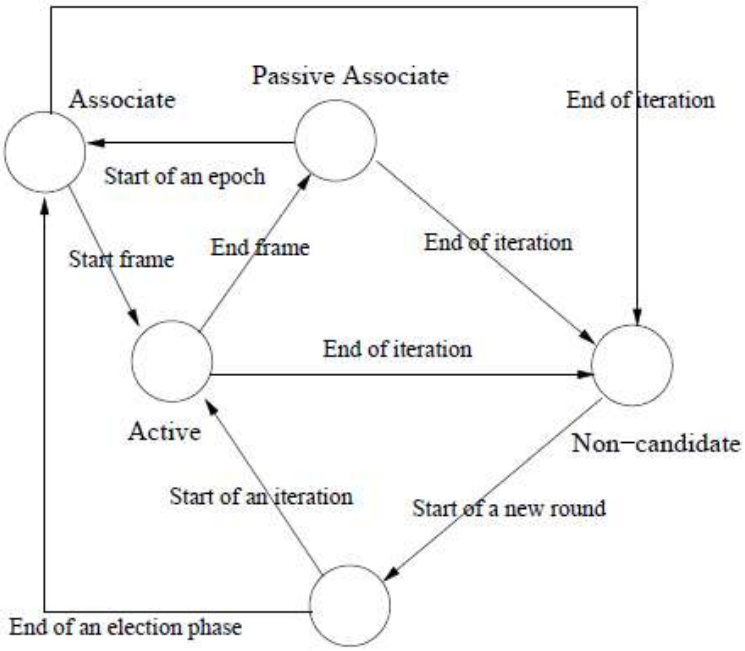


Figure 3.4: States of a sensor node in a wireless sensor network (Hussain & Matin, 2005)

3.7 Error Detection Mechanism

In the model, whenever packet reaches the sink, then the sink tries to decrypt the packet using session key and checks the contents of the packet with the logical time stamp and the corresponding cluster head id. If packet is altered or losses any content of the packet in the communication process, the sink will discard the packet or re-send the transmission request to the corresponding gateway, thus the sort of errors can be detected.

3.8 Conclusion

In this chapter, a new framework for secure energy efficient data routing protocol (SECDRP) is proposed. The proposed framework uses Blowfish encryption and decryption algorithm, Cluster-based Aggregation on secure energy efficient

algorithms and low energy Wireless Sensor Gateway Networks (WSGNs). The entire framework is based on a three level architecture for energy constrained sensor nodes at lower level (generic sensors), sizeable numbers of energy rich Cluster head and Gateway at the middle level (special-purpose sensors), and a sink (base station) which monitors the activities of sensor field at the upper level (high-bandwidth sensors). In the propose scheme a cluster head-set is selected which contains several nodes that controls and manages the cluster nodes. Communication between sensor nodes and the sink is secured as the sensor data is encrypted using symmetric key cryptography. The propose scheme has two communication stages which are the election phase and data transfer phase. It also explains the different states of the sensor nodes. Again, the propose scheme has an error detection mechanism to check the packets during transmission.

CHAPTER FOUR

RESULTS AND DISCUSSIONS

Performance analysis of Wireless Sensor Network Secure Framework (WSNSF) Architecture

4.1 Introduction

The proposed Wireless Sensor Network Secure Framework (WSNSF) provides a secure data routing environment for three-level Wireless Sensor Gateway Networks (WSGNs). MATLAB is used for the simulation of the propose Protocol.

Radio communication model is also used in the quantitative analysis of the propose Protocol (SECDRP).

4.2 Simulation Platform

MATLAB precisely MATLAB 7.11.0 (2010b) was selected as a tool for the simulation of the propose protocol. MATLAB is a high-level technical computing language and an interactive environment for algorithm development, data visualization, data analysis, and numerical computation. MATLAB can solve technical computing problems faster than with traditional programming languages, such as C, C++, and FORTRAN. It contains hundreds of commands to do mathematics. It can be used to draw graph functions, solve equations, perform statistical tests, and much more. It is a high-level programming language that can communicate with other programming languages like FORTRAN and C.

4.3 Quantitative Analysis

This section describes a radio communication model that is used in the quantitative analysis of the protocol (SECDRP).

The qualitative analysis will determine the following:

- The optimum number of clusters
- The energy dissipation
- Number of frames
- Time for message transfer

4.4 Radio Communication Model

The researcher use a radio model as described in (Chandrakasan, Heinzelman & Balakrishnan, 2000), where for a shorter distance such as single-hop transmission, for instance direct data transfer from sensor node to cluster-head, the energy consumed by a transmit amplifier is proportional to d^2, where d is distance.

However, for a longer distance transmission, such as multi-hop transmission from a sensor node to the sink, the energy consumed is proportional to d^4. Using the given radio model, the energy consumed to transmit l-bit message for a longer distance, d, is given by:

$$E_T = E_e + E_L d^4 \qquad (4.1)$$

Where

E_T = Energy consumed to transmit

E_e = Energy consumed in the electronics circuit to transmit or receive the signal

E_L = Energy consumed by the amplifier to transmit at a longer distance

d^4 = long distance

In the same way, the energy consumed to transmit l-bit message for a shorter distance is given by:

$$E_T = E_e + E_S d^2 \qquad (4.2)$$

Where

E_T = Energy consumed to transmit

E_e = Energy consumed in the electronics circuit to transmit or receive the signal

E_s = Energy consumed by the amplifier to transmit at a shorter distance

d^2 = short distance

Moreover, the energy consumed to receive l-bit message is given by:

$$E_R = E_e + E_{BF} \quad (4.3)$$

Where

E_R = Energy consumed to receive

E_e = Energy consumed in the electronics circuit to transmit or receive the signal

E_{BF} = Energy consumed for beam forming (The beam forming approach reduces energy consumption.

The constants used in the radio model are given in Table 4.1.

Table 4.1: Sample parameter values of the radio communication model used in the quantitative analysis (Chandrakasan, Heinzelman & Balakrishnan, 2000)

Description	Symbol	Value
Energy consumed by the amplifier to transmit at a shorter distance	E_S	10 pJ/bit/m^2
Energy consumed by the amplifier to transmit at a longer distance	E_L	0.0013 pJ/bit/m^4
Energy consumed in the electronics circuit to transmit or receive the signal	E_e	50 nJ/bit
Energy consumed for beam forming	E_{BF}	5 nJ/bit

4.4.1 Election Phase

For a sensor network of n nodes, the optimal number of clusters is given as c. All nodes are assumed to be at the same energy level at the beginning. The amount of consumed energy is same for all the clusters. At the start of the election phase, the base station (sink) randomly selects a given number of cluster heads.

- First, the cluster heads broadcast messages to all the sensors in their neighborhood.
- Second, the sensors receive messages from one or more cluster heads and choose their cluster head using the received signal strength.
- Third, the sensors transmit their decision to their corresponding cluster heads.
- Fourth, the cluster heads receive messages from their sensor nodes and remember their corresponding nodes.

For each cluster, the corresponding cluster head chooses a set of s (sensor data) associates, based on signal analysis. For uniformly distributed clusters, each cluster contains $\frac{n}{c}$ nodes where n is number of nodes and c is number of clusters.

Using Equation 4.2 and Equation 4.3, the energy consumed by a cluster head is estimated as follows:

$$E_{CH\text{-}elec} = \left\{ E_e + E_S\, d^2 \right\} + \left\{ \left(\frac{n}{c} - 1 \right) (E_e + E_{BF}) \right\} \qquad (4.4)$$

Where

$E_{CH\text{-}elec}$= Energy consumed by a cluster head

n= number of nodes

c= number of clusters

The first part of Equation 4.4 represents the energy consumed to transmit the advertisement message; this energy consumption is based on a shorter distance energy dissipation model. The second part of Equation 4.4 represents the energy consumed to receive $\left(\frac{n}{c} - 1\right)$ messages from the sensor nodes of the same cluster. Equation 4.4 can be simplified as follows:

$$E_{CH\text{-}elec} = E_e \frac{n}{c} + E_{BF}\left(\frac{n}{c} - 1\right) + E_S d^2 \qquad (4.5)$$

Using Equation 4.2 and Equation 4.3, the energy consumed by non-cluster head sensor nodes is estimated as follows:

$$E_{non\text{-}CH\text{-}elec} = \{cE_e + cE_{BF}\} + \{(E_e + E_S d^2)\} \qquad (4.6)$$

Where

$E_{non\text{-}CH\text{-}elec}$ = Energy consumed by a non-cluster head

c= number of clusters

The first part of Equation 4.6 shows the energy consumed to receive messages from c cluster heads; it is assumed that a sensor node receives messages from all the cluster heads. The second part of Equation 4.6 shows the energy consumed to transmit the decision to the corresponding cluster head. Equation 4.6 can be simplified as follows:

$$E_{non\text{-}CH\text{-}elec} = E_e (1 + c) + cE_{BF} + E_S d^2 \qquad (4.7)$$

4.4.2 Data Transfer Phase

During data transfer phase, the nodes transmit messages to their cluster head and cluster heads transmit the aggregated messages to a distant base station through

their respective distributed gateways. The energy consumed by a cluster head is as follows:

$$E_{CH / frame} = \{E_e + E_L d^4\} \{(\frac{n}{c} - s)(E_e + E_{BF})\} + 1 \quad (4.8)$$

Where

$E_{CH / frame}$ = energy consumed by a cluster head in transmitting messages or frames

S= Sensor data or message

The first part of Equation 4.8 shows the energy consumed to receive messages from the remaining $(\frac{n}{c} - S)$ nodes which is not a part of the cluster head-set. The second part of Equation 4.8 shows the energy consumed to transmit a message to the distant sink. Equation 4.8 can be simplified as follows:

$$E_{CH / frame} = E_L d^4 + (\frac{n}{c} - s + 1) E_e + (\frac{n}{c} - s) E_{BF} \quad (4.9)$$

The energy, $E_{non\text{-}CH / frame}$, consumed by a non-cluster head node to transmit the sensor data to the gateway is given below:

$$E_{non\text{-}CH/frame} = E_e + E_S d^2 \quad (4.10)$$

Where

$E_{non\text{-}CH/frame}$ = energy consumed by a non-cluster head node to transmit the sensor data

For circular clusters with a uniform distribution of sensor nodes and a network diameter of M, the average value of d^2 is, given, as: $E[d^2] = \frac{M^2}{2\pi c}$. Equation 4.10 can be simplified as follows:

$$E_{\text{non-CH/frame}} = E_e + E_S\left(\frac{M^2}{2\pi c}\right) \qquad (4.11)$$

Where

M = Network diameter

In one iteration, N_f (where N_f is number of data frames) data frames are transmitted.

The N_f/c (where N_f/c is the frames transmitted by each cluster) frames are uniformly divided among n/c nodes of the cluster. Each cluster head frame transmission needs $\frac{n}{c} - s$ non-cluster head frames. For simplification of equations, the fractions f_1 and f_2 are given as below:

$$f_1 = \left(\frac{1}{\frac{n}{c} - s + 1} \right) \frac{1}{c} \qquad (4.12)$$

$$f_2 = \left(\frac{\frac{n}{c} - s}{\frac{n}{c} - s + 1} \right) \frac{1}{c} \qquad (4.13)$$

Where

f_1 = cluster frames and f_2 = non-cluster frames

The energy consumptions in a data transfer stage of each cluster are as follows:

$$E_{\text{CH-data}} = f_1 N_f E_{\text{CH/frame}} \qquad (4.14)$$

Where

$E_{\text{CH-data}}$ = energy consumption in a data transfer stage of each cluster

f_1 = cluster frames

N_f = number of data frames

$$E_{non\text{-}CH\text{-}data} = f_2 \, N_f \, E_{non\text{-}CH/frame} \qquad (4.15)$$

Where

$E_{non\text{-}CH\text{-}data}$ = energy consumption in a data transfer stage of each non-cluster

f_2 = non-cluster frames

4.4.3 Start Energy for One Round

There are c clusters and n nodes. In each iteration, s nodes are elected for each cluster. Thus, in each iteration cs nodes are elected as members of head-sets. The number of iterations required for all n nodes to be elected is $(\frac{n}{cs})$, which is the number of iterations required in one round. Moreover, an iteration consists of an election phase and a data transfer stage. The energy consumed in one iteration of cluster is as follows:

$$E_{CH/iter/cluster} = E_{CH\text{-}elec} + E_{CH\text{-}data} \qquad (4.16)$$

Where

$E_{CH/iter/cluster}$ = energy consumed in one iteration of cluster head

$E_{CH\text{-}elec}$ = energy consumed by cluster head sensor node

$E_{CH\text{-}data}$ = energy consumed in a data transfer of each cluster head

$$E_{non\text{-}CH/iter/cluster} = E_{non\text{-}CH\text{-}elec} + E_{non\text{-}CH\text{-}data} \qquad (4.17)$$

Where

$E_{non\text{-}CH/iter/cluster}$ = energy consumed in one iteration of non-cluster head

$E_{non\text{-}CH\text{-}elec}$ = energy consumed by non-cluster head sensor node

$E_{non\text{-}CH\text{-}data}$ = energy consumed in a data transfer of each non-cluster head

Since there are s nodes in a cluster head-set, the $E_{CH/iter/cluster}$ is uniformly divided among the cluster head-set members, as given below:

$$E_{CH/node} = \frac{E_{CH/iter/cluster}}{s} \quad (4.18)$$

Similarly, there are $\frac{n}{c} - s$ non-cluster head nodes in a cluster.

The $E_{non\text{-}CH/iter/cluster}$ is uniformly distributed among all the non-cluster head members as follows:

$$E_{non\text{-}CH/node} = \frac{E_{non-CH/iter/cluster}}{\frac{n}{c} - s} \quad (4.19)$$

The start energy, E_{start}, is energy of a sensor node at the initial start time. This energy should be sufficient for at least one round. In one round, a node becomes a member of head-set for one time and a non-cluster head for $\frac{n}{cs} - 1$ times. An estimation of E_{start} is given below:

$$E_{start} = E_{CH/node} + \left(\frac{n}{cs} - 1\right) E_{non\text{-}CH\text{-}node} \quad (4.20)$$

Using Equation 4.18, Equation 4.19 and Equation 4.20, E_{start} can be described as below:

$$E_{start} = \frac{1}{s}(E_{CH/iter/cluster} + E_{non\text{-}CH/iter/cluster}) \quad (4.21)$$

Using Equation 4.21, Equation 4.16, Equation 4.17, Equation 4.14, and Equation 4.15, E_{start} can be given as follows:

$$E_{start} = \frac{E_{CH-elec} + E_{non-CH-elec}}{s} + \frac{N_f}{s}\left(f_1 E_{CH/frame} + f_2 E_{non-CH/frame}\right) \quad (4.22)$$

Using Equation 4.22, N_f can be given as below:

$$N_f = \frac{sE_{start} - E_{CH-elec} + E_{non-CH-elec}}{f_1 E_{CH/frame} + f_2 E_{non-CH/frame}} \quad (4.23)$$

4.4.4 Optimum Number of Clusters

In a cluster, the energy consumed to transmit an aggregated reading to the base station is as follows:

$$E_c = E_{CH/frame} + \left(\frac{n}{c} - s\right)E_{non\text{-}CH/frame} \quad (4.24)$$

The first part of Equation 4.24 is due to the energy consumption as an active member of the cluster head-set. The second part of Equation 4.24 is due to $\left(\frac{n}{c} - s\right)$ non-cluster head nodes. The total energy consumed by c clusters is as follows:

$$E_{total/frame} = cE_c \quad (4.25)$$

Using Equation 4.25, Equation 4.24, Equation 4.11, and Equation 4.9, the total energy consumed by c clusters is given below:

$$\mathrm{E}_{\text{total/frame}} = \left\{E_L d^4 + \left(\frac{n}{c} - s + 1\right)E_e + \left(\frac{n}{c} - s\right)E_{BF}\right\} + c\left\{\left(\frac{n}{c} - s\right)\left(E_e + E_s \frac{M^2}{2\pi c}\right)\right\}$$

(4.26)

The above equation can be simplified as follows:

$$\mathrm{E}_{\text{total/frame}} =$$

$$\left\{cE_L d^4 + (n - cs + c)E_e + (n - cs)E_{BF}\right\} + \left\{(n - cs)E_e + (n - cs)E_s \frac{M^2}{2\pi c}\right\} \qquad (4.27)$$

$$\mathrm{E}_{\text{total/frame}} =$$

$$cE_L d^4 + (2n - 2cs + c)E_e + (n - cs)E_{BF} + nE_s \frac{M^2}{2\pi c} - E_s \frac{M^2}{2\pi c} \qquad (4.28)$$

The optimum number of c for minimum consumed energy can be determined as follows:

$$\frac{dE_{total}}{dk} = 0 \qquad (4.29)$$

Using Equation 4.28 and Equation 4.27, the following result was obtained:

$$E_l d^4 - (2s - 1)E_e - sE_{BF} - nE_s \frac{M^2}{2\pi k^2} = 0 \qquad (4.30)$$

Using Equation 4.30, the optimum value of c for minimum dissipation of frame energy is as follows:

$$c = \sqrt{\frac{n}{2\pi}} \sqrt{\frac{E_s}{E_l d^4 - (2s-1)E_e - sE_{BF}} M} \qquad (4.31)$$

4.4.5 Time to Complete One Round

Sensor nodes transmit messages according to a specified schedule, which is based on TDMA (Time Division Multiple Access). The frame time, t_{frame} is the addition of transmission times of the messages transmitted by all the nodes of a cluster. The time to transfer a message, t_{mgs}, is as follows:

$$t_{mgs} = \frac{l}{R_b} \qquad (4.32)$$

Where

R_b bits/second = data transfer rate

l bits = message length

In one frame, messages are transmitted by all the non-cluster head nodes and the active member of the cluster head-set. Since at one time only one member of head-set is active, the inactive head-set members, which are $s - 1$, do not transmit during the frame transmission. The time for one frame is given as follows:

$$t_{frame} = \left\{ \sum_{i=1}^{\frac{n}{c}-s} t_{msg_i} \right\} + \left\{ t_{msg_{cluster\_head}} \right\} \qquad (4.33)$$

The first part of Equation 4.33 is due to $\frac{n}{c} - s$ messages from non-cluster head nodes. The second part of Equation 4.33 is due to the transmission of the active member of

the cluster head-set. If it is assumed that message transfer time is same for all the nodes, Equation 4.33 can be simplified as follows:

$$t_{frame} = \left(\frac{n}{c} - s + 1 \right) t_{msg} \quad (4.34)$$

As N_f frames are transmitted in one iteration, time for one iteration, $t_{iteration}$ is as follows:

$$t_{iteration} = t_{frame} \times N_f \quad (4.35)$$

Using Equation 4.35, Equation 4.34, and Equation 4.32, the iteration time $t_{iteration}$ can be given as below:

$$t_{iteration} = \frac{l}{R_b} \left(\frac{n}{c} - s + 1 \right) N_f \quad (4.36)$$

Since there are $\frac{n}{cs}$ iterations in one round, the time for one round, t_{round}, is as follows:

$$\begin{aligned} t_{round} &= t_{iteration} \times \frac{n}{cs} \\ &= \frac{l}{R_b} \frac{n}{c} \left(\frac{n}{c} - s + 1 \right) \frac{N_f}{s} \end{aligned} \quad (4.37)$$

4.5 Results and Discussions

In this section the results are analyzed for the proposed routing protocol (SECDRP) using the radio communication model for the quantitative analysis.

4.5.1 Optimum Number of Clusters

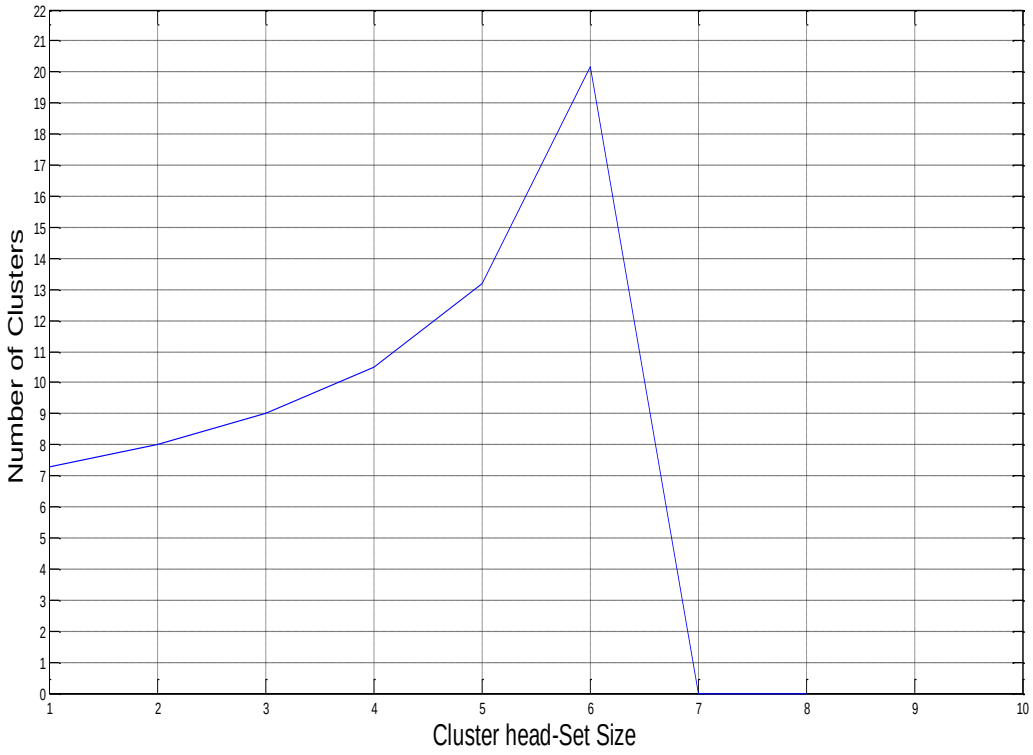


Figure 4.1: Optimum Number of Clusters with Respect to the Cluster head-set size.

Figure 4.1 shows the optimum number of clusters with respect to the cluster head-set size. From the graph, the maximum number of clusters is approximately 20 and its corresponding cluster head-set size is 6. As the graph shows, the number of clusters decreases as the number of cluster head-set size exceeds 6. The minimum number of clusters is approximately 7 and its corresponding cluster head-set size is 1. Therefore as the number of clusters increases the cluster head-set size also increases. As a result, bigger number of clusters can manage bigger cluster head set size whiles smaller number of clusters can manage smaller cluster head-set size. The reason for this observation is that, in the wireless sensor network, if a cluster head-set size is not carefully chosen for its respective number of clusters, during data transmission much burden is put on the cluster head-set nodes. This causes fast depletion of the batteries of the cluster head set nodes since the nodes depend on their respective batteries for

their power source. For this reason power conservation and power management should be seriously considered in Sensor Networks.

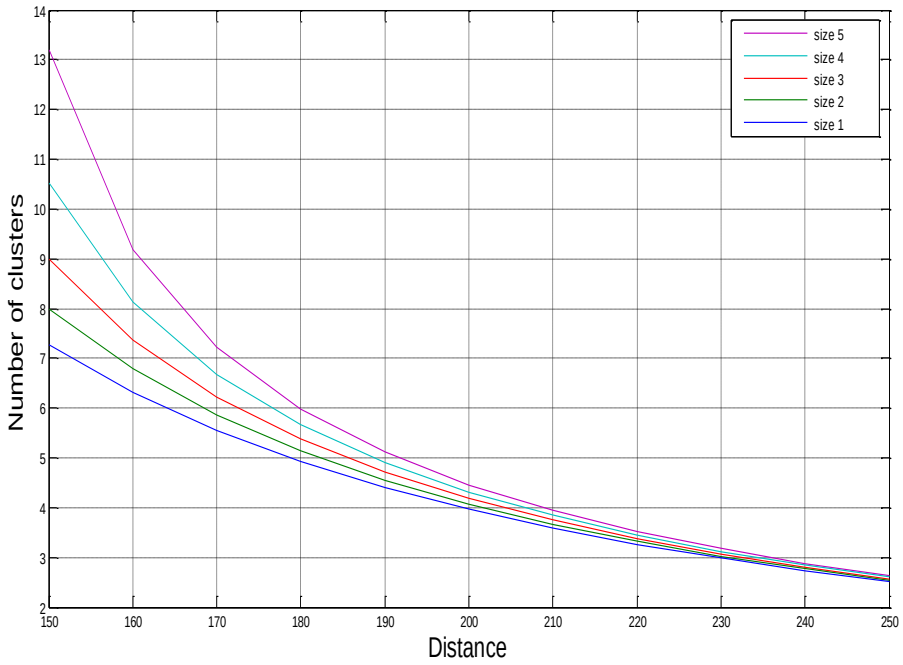


Figure 4.2: Number of Clusters with Respect to Distance from the Base Station for various Number of Cluster Head-set sizes.

Figure 4.2 shows the variation in number of clusters with respect to distance from the base station for various number of cluster head-set sizes. In the graph the number of clusters decreases as the distance from the base station increases. This variation is the same for the various cluster head set sizes (that is from cluster head set size 1 to cluster head set size 5). This implies that the cluster head set size does not vary with respect to distance but rather the number of clusters. As a result, when the number of clusters increases, the distance from the base station decreases hence less energy would be required during the transmission of frames. This is because, the sensor node is a tiny device and because of its tiny nature it has challenges with its power control unit and communication unit. Due to these limitations, energy consumption

should be as efficient as possible to extend the network lifetime. Also if the transmission range from the base station is short it reduces the possibility of data being eavesdropped. On the other hand, as the distance from the base station increases, the number of clusters reduces, hence more energy would be required during the transmission of frames. Apparently, when the transmission range from the base station is long, nodes would require high transmission power to reach the base station and also increases the chance of eavesdropping.

4.5.2 Energy Consumption

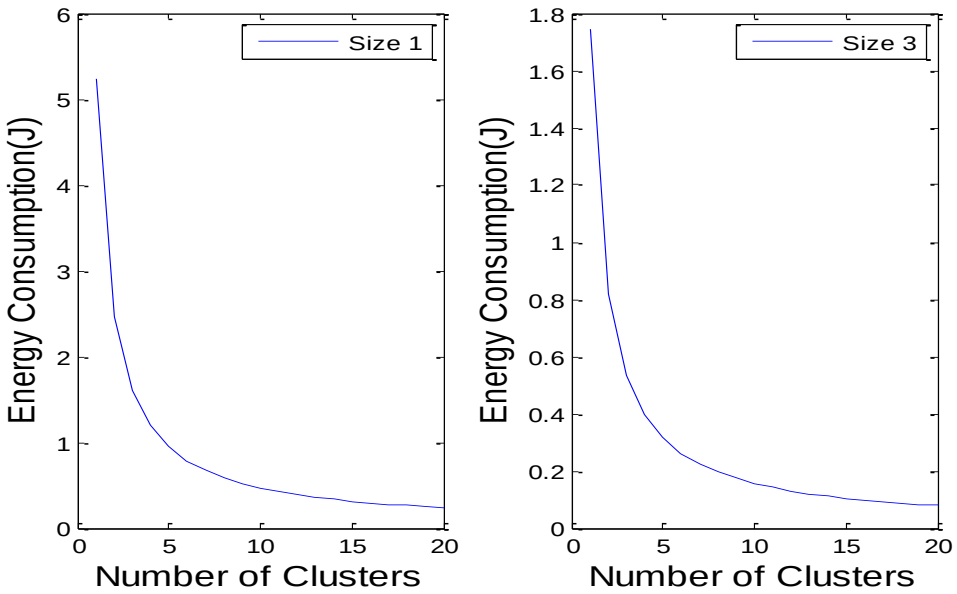


Figure 4.3a: Optimum Number of Clusters Figure 4.3b: Optimum Number of Clusters

Figure 4.3a and 4.3b illustrates the energy consumption with respect to the number of clusters for various cluster head-set sizes.

From Figure 4.3a the energy consumption reduces as the number of clusters increases. The optimum variation in the energy consumption ranges between 0 (Joules) and 6 (Joules) when the cluster head-set size is 1.

From Figure 4.3b the energy consumption reduces as the number of clusters increases. The optimum variation in the energy consumption ranges between 0 (Joules) and 1.8 (Joules) when the cluster head-set size is 3.

Therefore comparing the two graphs, the energy consumption in Figure 4.3b is comparatively lower when cluster head-set size is 3 as compared to Figure 4.3a when cluster head-set size is 1. The energy consumed in Figure 4.3b is approximately three times less when cluster headset size is 3 as compared to Figure 4.3a (LEACH), when cluster head-set size is 1. As a result, the bigger the cluster head-set size the lower the energy consumption during transmission and vice versa. The reason for this is that, if the energy consumption is reduced in the network, the lifetime of the network would be prolonged and more transmissions can take place in the network.

4.5.3 Iteration Time and Frames

The average time to complete one iteration such that every node becomes a member of cluster head-set is estimated. In other words, an average time for one iteration in each round is estimated. Moreover, frames transmitted in each iteration are also evaluated.

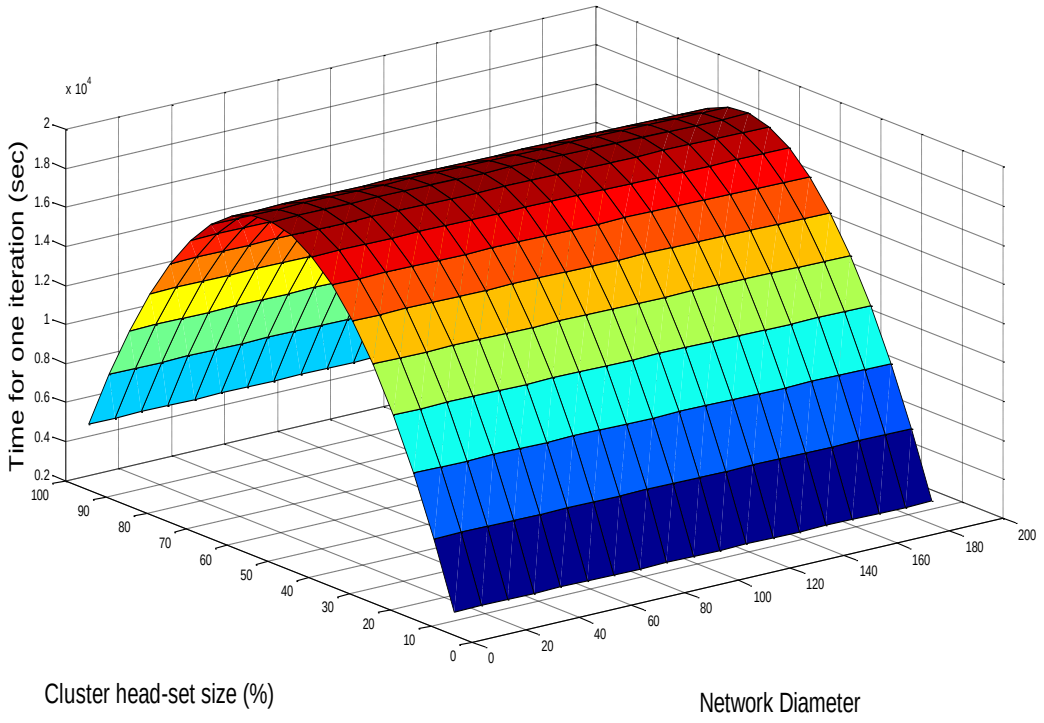


Figure 4.4: Time for Iteration with Respect to the Cluster Head-set size and Network Diameter.

The Figure 4.4 illustrates the variation in time to complete one iteration with respect to cluster head-set size and network diameter. The x-axis, y-axis, and z-axis represent the network diameter, cluster head-set size, and time to complete one iteration, respectively. The cluster head-set size is given as a percentage of cluster size (number of clusters). The start energy, E_{start} is fixed for all the cases. From this graph, when the cluster headset size is less than 50% of the cluster size, there are fewer transmissions in each iteration because distance from the base station increases as illustrated in Figure 4.2 and therefore would require more iterations to complete the round. However, when the cluster head-set size is greater than 50% of the cluster size, there are more transmissions in each iteration because distance from the base station decreases, hence, less iterations to complete the round.

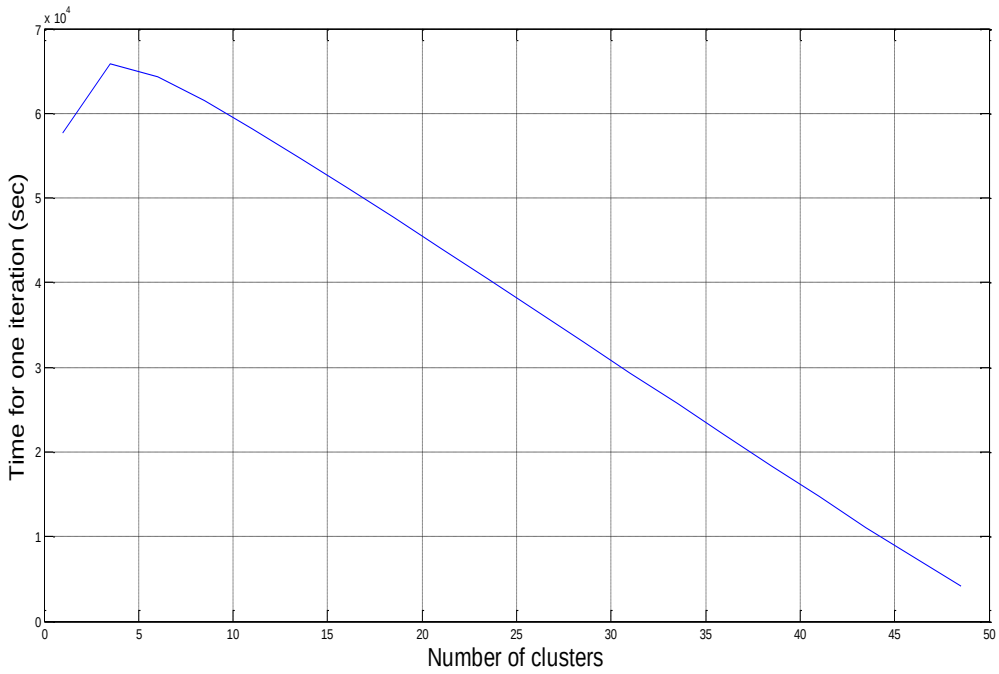


Figure 4.5: Time for Iteration with Respect to the Number of Clusters.

Figure 4.5 shows the variation in time for one iteration with respect to the number of clusters. As illustrated in the graph, the time for one iteration decreases as the number of clusters increases. Therefore, for larger number of clusters, the time for one iteration reduces. The reason for this is that, when more nodes are introduced into the network it reduces the time for transmission hence, more nodes can be transmitted within a short time. This helps the nodes to conserve more energy to extend the network lifetime.

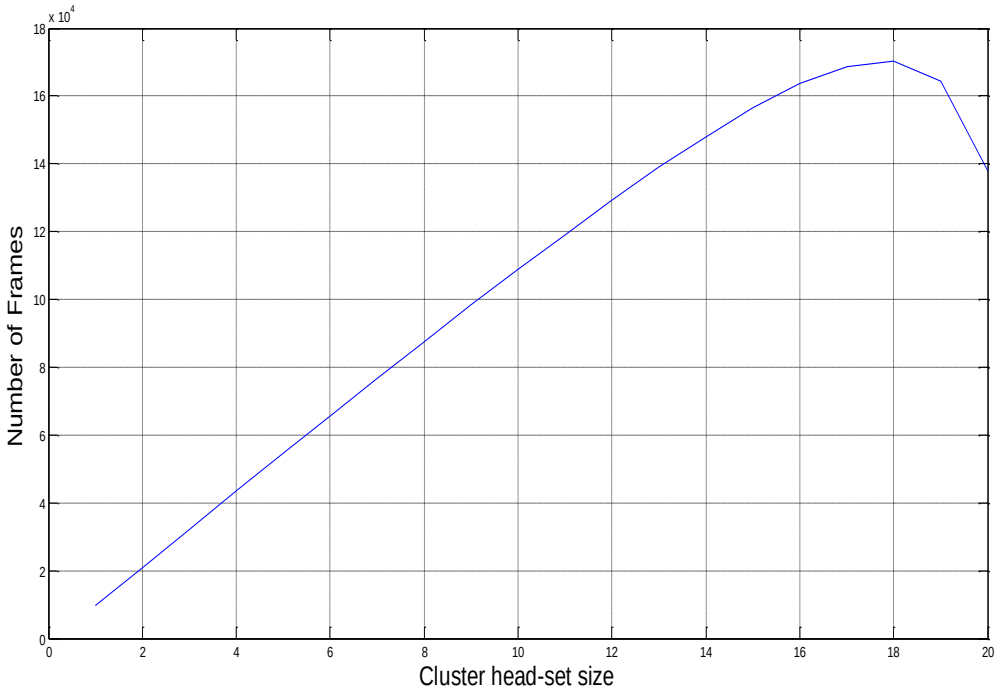


Figure 4.6: Number of Frames Transmission per Iteration with Respect to the Cluster Head-set size.

Figure 4.6 illustrates the number of frames transmitted with respect to the cluster head-set size. From the graph, the number of frames transmitted increases as cluster head-set size also increases. As expected, for increased value of cluster head-set size, more frames can be transmitted since iteration will be few. As a result, the iteration can last for a longer time, which is also consistent with the results shown in Figure 4.5. This is because, when the cluster head-set size is increased, there would be more control and management of sensor nodes. Consequently, the data collecting nodes can be used for a longer period of time.

4.6 Conclusion

In summary when optimum number of clusters with respect to the cluster head-set size was compared, it was observed that bigger cluster size can manage

bigger cluster head-set size whiles smaller cluster size can manage smaller cluster head-set size. Comparing the cluster size with respect to distance from base station, the maximum number of clusters decreases when the distance from the base station increases. As a result, when the distance from the base station is increased, more energy is spent for a distant transmission.

When energy consumed for specific number of frames and different cluster head-set sizes were compared, it was observed that, when the number of clusters is greater than the optimum range, there will be more transmissions to the distant base station since less energy will be required. Also using a cluster head-set of sensor nodes is more desirable than a single cluster head.

Time for iteration with respect to cluster head-set size was compared and it was observed that the start energy can be used for the longest period of time when the head-set size is 50% of the cluster size (number of clusters). When the cluster headset size is less than 50% of the cluster size, there are fewer transmissions in each iteration but there are more iterations to complete the round. However, when the cluster head-set size is greater than 50% of the cluster size, there are more transmissions in each iteration, although there are less iterations to complete the round. As a result, the cluster head-set size and the number of clusters should be carefully chosen to extend the network life time. Also, for increased value of cluster head-set sizes, more frames can be transmitted. In other words, when the cluster head-set size is increased, there would be more control and management of sensor nodes.

4.7 Findings

- Bigger cluster size can manage bigger cluster head-set size whiles smaller cluster size can manage smaller cluster head-set size.

- Distance from the base station reduces when there are more clusters, hence less energy is spent during data transmission
- Energy consumption reduces as the number of clusters increases
- The start energy can be used for the longest period of time when the cluster head-set size is 50% of the cluster size (number of clusters).
- When the cluster headset size is less than 50% of the cluster size, there would be fewer transmissions in each iteration but there would be more iterations to complete the round. However, when the cluster head-set size is greater than 50% of the cluster size, there would be more transmissions in each iteration and therefore less iterations to complete the round.
- More frames can be transmitted if cluster head-set size increases.
- For these reasons, the number of clusters and cluster head-set size should be carefully chosen to extend the network life time and also control and manage the sensor nodes efficiently within a network.

CHAPTER FIVE

CONCLUSIONS, RECOMMENDATIONS AND FUTURE WORK

5.1 Conclusions

At the end of the study it was observed that Wireless Sensor Networks have drawn a lot of attention due to their broad applications. Sensor nodes in the network are characterized by severely constrained energy resources and communicational capabilities. Sensor nodes call for energy efficient and secure communication protocols because of their limited capabilities that include storage, power and processing. Consequently, if security in data communication in WSNs is not considered, it will result in mistrust and subsequently restrict people from taking advantage of the full benefit of the technology.

There are several conventional protocols for data routing which use different approaches in providing security and energy efficiency in wireless sensor networks that are limited. In the cluster-based approach, nodes send their data to cluster-head and cluster-head then aggregate and forward the data towards the sink. This approach was explored by the researcher and a new protocol called Secure Energy Efficient Cluster-based Data Routing Protocol (SECDRP) was proposed.

SECDRP uses positive features of symmetric key cryptography and cluster-based methods. In SECDRP the wireless sensor network is divided into several clusters. Each cluster has a head set and a gateway node. Again each cluster has a cluster head-set which contains several associate cluster heads. Sensor nodes in a cluster select its cluster head from the cluster head set to perform the data aggregation before sending it to the sink. The cluster head set was introduced to reduce energy consumption instead of using one static cluster head as in the LEACH protocol to perform all data aggregation. A gateway node was also introduced to allow multihop inter-cluster routing where the cluster head forwards aggregated data

through its gateway to the sink. Other conventional protocols use single hop inter-cluster routing where cluster head forwards aggregated data straight to the sink. This burdens the cluster head in addition to the responsibility of in-network operations such as data aggregation and fusion. Even if cluster heads are equipped with more powerful and durable batteries, this heavy burden could result in fast battery depletion of the cluster heads and thus gives it a shorter lifetime compared to other sensor nodes.

In this study the performance of the new proposed protocol (SECDRP) for wireless sensor network was analyzed using the Radio Communication Model. The results of the analysis have been presented in the form of graphs. The results of the analysis suggest that the proposed protocol performs better than the conventional ones.

5.2 Recommendations

In terms of security, asymmetric cryptography is more secured and reliable than the symmetric cryptography in data encryption. The reason is that asymmetric cryptography uses different keys to encrypt (Public key) and decrypt (Private key) whiles the symmetric cryptography uses the same key to encrypt and decrypt and as such an adversary that subverts even a single node can obtain access to the secret key. On the other hand symmetric cryptography is well known to be computationally less expensive and therefore more suited for resource constrained sensor networks than asymmetric cryptography which comes with its associated computational cost and therefore not suitable for resource-constrained sensor networks.

Even though the asymmetric cryptography is computationally expensive, it is more secured than the symmetric cryptography and therefore should be explored to

reduce the computational cost so that it can be used in resource constrained sensor networks to ensure reliability and robustness of the network.

5.3 Future work

Future work will focus on investigation of the variation in the cluster head-set size for different network conditions. This work will be extended to incorporate non-uniform cluster distributions. A simulation model will be developed to validate and verify the quantitative analysis. After the simulation model, focus will be on the implementation of SECDRP in NS-2 as a separate module so that it could be tested more accurately.

REFERENCES

Abdul-Elminaam, D. S., Abdul-Kader, H. M., & Hadhoud, M. M. (2008). Performance evaluation of symmetric encryption algorithms. IJCSNS

International Journal of Computer Science and Network Security, 8(12), 280-286.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless Sensor Networks: A Survey, Computer Networks, Journal, 38(4): 393-422.

Blaze, M. (1996). Cryptography policy and the information economy. AT and Tlabs Research. mab@research.att.com.

Blaβ, E. & Zitterbart, M. (2006). An efficient key establishment scheme for secure aggregating sensor networks. ASIACCS'06, pp. 233–241. ACM 1-59593-272-0/06/0003.

Burman, S. (2007). Cryptography and security - Future challenges and issues. 15th International Conference on Advanced Computing and Communications, India: Guwahati, pp.547-551.

Cam, H., Ozdemir, S., Nair, P., Muthuavinashiapan, D., & Sanli, H. O. (2005). Energy-Efficient Secure Pattern Based Data Aggregation for Wireless Sensor Networks. Computer Communication, 29(1), Elsevier.

Carman, D. W., Krus, P. S., & Matt, B. J. (2000). Constraints and approaches for distributed sensor network security. Technical Report 00-010, NAI Labs, Network Associates, Inc., Glenwood, MD.

Chan, H. & Perrig, A. (2005). Pike: Peer intermediaries for key establishment in sensor networks. In IEEE Infocom.

Chan, H., Perrig, A., & Song, D. (2003). Random key predistribution schemes for sensor networks. Proceedings of the 2003 IEEE Symposium on Security and Privacy. IEEE Computer Society.

Chandrakasan, A., Heinzelman, W. R., & Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor network. IEEE

Proceedings of the 33rd Hawaii International Conference on System Sciences, Hawaii, USA, pp. 3000-3014.

Choi, W., & Das, S. K. (2004). A framework for energy-saving data gathering using two-phase clustering in wireless sensor networks. In Proceedings of Mobi Quitous Networking Conference.

Culler, D. E., & Hong, W. (2004). Wireless Sensor Networks, Communication of the ACM, 47(6): 30-33.

Dasgupta, K., Kalpakis, K., & Namjoshi, P. (2003). Improving the lifetime of sensor networks via intelligent selection of data aggregation trees. In Proc. of the CNDS'03, Orlando: Florida.

Devasenapathy, M., Suat, Z., & Nair, P. (2003). Energy efficient security protocol for wireless sensor networks. IEEE, pp. 0–7803–7954.

Ding, W., Iyengar, S. S., Kannan, R., & Rummler, W. B. (2004). Energy equivalence routing in wireless sensor networks. Special issue on Wireless Sensor Networks in Journal of Microcomputers and Applications, 28(8): 467-475.

El-Fishawy, N. (2007). Quality of encryption measurement of bitmap images with rc6, mrc6, and rijndael block cipher algorithms. International Journal of Network Security, pp. 241–251.

Fasolo, E., Rossi, M., Widmer, J. & Zorzi, M. (2007). In-Network Aggregation Techniques for Wireless Sensor Networks: A Survey. IEEE Wireless communication.

Frey, H., Ruehrup, S. & Stojmenovic, I. (2009). *Routing in wireless sensor networks*, Springer-Verlag, (4): 81–111.

Garg, V., Meitei, M. S., Raman, S., Kumar, A., Tewari, N., & Ghosh, R. K. (2006). "Ad hoc networks", 168-185.

Hanna, L. & Hailes, S. (2011). Privacy and Wireless Sensor Networks. University College, London.

Heinzelman, W. R. et al., (2000). "Energy-Scalable algorithms and protocols for Wireless Sensor Networks", in the Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00), Istanbul, Turkey.

Hussain, Sajid & Matin, Abdul W. (2005). Energy Efficient Hierarchical Cluster-Based Routing for Wireless Sensor Networks. Jodrey School of Computer Science, Acadia University, Wolfville, Nova Scotia, Canada, TR-2005-011.

Intanagonwiwat, C., Govindan, R., & Estrin, D. (2000). Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom ’00), Boston, USA.

Jolly, G., Kuşçu, Mustafa C., Kokate, P., & Younis, M. (2003). A low-energy key management protocol for wireless sensor networks. Proceedings of the Eighth IEEE International Symposium on Computers and Communication (ISCC’03), 410: 335-340.

Kasimoglui, Ismail H., & Akyildiz, Ian F. (2004). Wireless sensor and actor research challenges. (Elsevier) Journal, 2(38):351–367.

Kargl, F., Lawrence, E., Fischer, M. & Lim, Y. (2008). "Security, Privacy and Legal Issues in Persuasive eHealth Monitoring Systems". In Proceedings of Seventh International Conference on Mobile business, pp. 296-304.

Karlof, C., & Wagner, D. (2003). Secure routing in wireless sensor networks: Attacks and countermeasures. Elsevier’s AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols, 1(2-3):293-315.

Khan, M., Bhargava, B., Agarwal, S. Lilien, L., & Pankaj (2007). Self-configuring node clusters, data aggregation, and security in microsensor networks. Department of Management Information Systems Krannert Graduate School of Management Purdue University, West Lafayette, (IN47907), pankaj@mgmt.purdue.edu.

Kumar, S. P., & Chee-Yee, Chong (2003). Sensor networks: Evolution, opportunities, and challenges. Proc IEEE.

Lee, Weinan M., & Wong, Vincent W. S. (2005). An energy-aware spanning tree algorithm for data aggregation in wireless sensor networks. IEEE Pac Rrim.

Liu, D., & Ning, P. (2004). Multilevel μTesla: Broadcast authentication for distributed sensor networks. ACM Transactions on Embedded Computing Systems (TECS), New York, NY, USA 3(4): 800-836.

Madden, S., Franklin, M., Hellerstein, J., & Hong, W. (2002). 'Tag: A tiny aggregation service for adhoc sensor networks', OSDI 2002, December, Boston, MA.

Newsome, J., Shi, E., Song, D., & Perrig, A. (2004). "Sybil Attacks in Sensor Networks: Attacks and Analysis." In Proceedings of Conference on Information Processing in Sensor Networks (IPSN), pp. 259-268, ACM Press.

Parno, B., Perrig, A., & Gligor, V. (2005). Distributed detection of node replication attacks in sensor networks. In Proceedings of IEEE Symposium on Security and Privacy.

Pathan, A-S. K., Islam, H. K., Sayeed, S. A., Ahmed, F., & Hong, C. S., (2006). A Framework for Providing E-Services to the Rural Areas using Wireless Ad Hoc and Sensor Networks. In IEEE ICNEWS.

Perig, A., Szewczyk, R., Wen, V., Culler, D. & Tygar, J. D. (2002). "SPIN: Security Protocols for Sensor Networks". *Wireless Networks*, 8(5): 521-34.

Price, Samir A., & Kosaka, K. (2004). A secure key management scheme for sensor networks. Proceedings of the Tenth Americas Conference on Information Systems, New York.

Sahoo, P. K., Chen, J. J., & Sun, P. (2005). Efficient Security Mechanisms for the Distributed Wireless Sensor Networks. ICITA (2): 541-546.

Schneier, B. (1993). The blowfish encryption algorithm. Retrieved from http://www.schneier.com/blowfish.html, October 2011.

Sharma, P., Tyagi, D., & Bhadana, P. (2010). "A Study on Prolong the Lifetime of Wireless Sensor Network by Congestion Avoidance Techniques", International Journal of Engineering and Technology, 2(9): 4844-4849.

Sharma, Shriram (2009). *Energy-Efficient Secure Routing in Wireless Sensor Networks.* MTech thesis.

Sinha, A., & Chandrakasan, A. (2001). Dynamic power management in wireless sensor networks, IEEE Design and Test of Computers, vol. 18(2): 62-74.

Song, D., Przydatek, B., & Perrig, A. (2003). SIA: Secure information aggregation in sensor networks. Proceedings of the IEEE Conference on Measuring and Modeling of Computer Systems, pp. 255-265, ACM, New York.

Tamimi, A. A. (2008). "Performance Analysis of Data Encryption Algorithms". http://www.cs.wustl.edu/~jain/cse56706/ftp/encryption_perf/index.html, Accessed 20/10/2011.

Toh, C. K. (2002). Ad-hoc Mobile Wireless Networks Protocols and Systems", Prentice Hall, Inc.

Vaidyanathan, K., Sur, S., Narravula, S., & Sinha, P. (2004). Data aggregation techniques in sensor networks. *Technical Report, OSU-CISRC 11/04-TR60*.

Wood, A. D., & Stankovic, J. A. (2002). "Denial of Service in Sensor Networks," *Computer*, 35(10): 54–62.

APPENDICES

MATLAB Source Codes the several graphs

MATLAB source code for the graph of Figure 4.1

```
el=0.0013*10^-12; es=10*10^-12; Ee=50*10^-9; Ebf=5*10^-9;
n=2000; l=4000; Nf=10000; M=100;
[s]=[1:1:8];
d=150;
```

```
f1=el*(d^4)-Ebf.*s-(2*s-1)*Ee;
c=((n/(2*3.14))^(0.5))*((es./f1).^(0.5))*M;
plot(s,c);
xlabel('Cluster head-Set Size', 'FontSize',18)
ylabel('Number of Clusters', 'FontSize',18)
```

MATLAB source code for the graph of Figure 4.2

```
el=0.0013*10^-12; es=10*10^-12; Ee=50*10^-9; Ebf=5*10^-9;
n=2000; d=110; l=4000; M=100;
[s,d]=meshgrid([1:1:5],[150:10:250]);
f1=-Ebf.*s+el.*(d.^4)-(2*s-1)*Ee;
c=((n/(2*3.14))^(0.5))*((es./f1).^(0.5))*M;
plot(d,c);
xlabel('Distance', 'FontSize',18)
ylabel('Number of clusters', 'FontSize',18)
```

MATLAB source code for the graph of Figure 4.3a and 4.3b

```
el=0.0013*10^-12; es=10*10^-12; Ee=50*10^-9; Ebf=5*10^-9;
n=2000; M=100; d=100; l=4000; Nf=10000;
[c]=[1:1:20];
s=1;
Ech_elect=l.*(Ee.*((n./c)-s+1)+es*d^2);
Enon_ch_elect=l.*(Ee.*(1+c)+es*d^2);
Ech_frame=l.*(((n./c)-s+1)*Ebf+el*d^4+((n./c)-s+1)*Ee);
```

```
Enon_ch_frame=l.*(Ee+es*((M^2)./((2*3.14159).*c)));
Ech_data=(1./((n./c)-s+1)).*(Nf./c).*Ech_frame;
Enon_ch_data=(((n./c)-s)./((n./c)-s+1)).*(Nf./c).*Enon_ch_frame;
Ech_iter=Ech_elect+Ech_data;
Enon_ch_iter=Enon_ch_elect+Enon_ch_data;
Eround=(Ech_iter/s)+(((n./(c*s))-1).*Enon_ch_iter)./((n./c)-s);
subplot(1,2,1);plot(c,Eround);
xlabel('Number of Clusters', 'FontSize',14)
ylabel('Energy Consumption(J)', 'FontSize',14)
>> el=0.0013*10^-12; es=10*10^-12; Ee=50*10^-9; Ebf=5*10^-9;
n=2000; M=100; d=100; l=4000; Nf=10000;
[c]=[1:1:20];
s=3;
Ech_elect=l.*(Ee.*((n./c)-s+1)+es*d^2);
Enon_ch_elect=l.*(Ee.*(1+c)+es*d^2);
Ech_frame=l.*(((n./c)-s+1)*Ebf+el*d^4+((n./c)-s+1)*Ee);
Enon_ch_frame=l.*(Ee+es*((M^2)./((2*3.14159).*c)));
Ech_data=(1./((n./c)-s+1)).*(Nf./c).*Ech_frame;
Enon_ch_data=(((n./c)-s)./((n./c)-s+1)).*(Nf./c).*Enon_ch_frame;
Ech_iter=Ech_elect+Ech_data;
Enon_ch_iter=Enon_ch_elect+Enon_ch_data;
Eround=(Ech_iter/s)+(((n./(c*s))-1).*Enon_ch_iter)./((n./c)-s);
Eround=(Ech_iter/s)+(((n./(c*s))-1).*Enon_ch_iter)./((n./c)-s);
subplot(1,2,2);plot(c,Eround);
xlabel('Number of Clusters', 'FontSize',14)
ylabel('Energy Consumption(J)', 'FontSize',14)
```

MATLAB source code for the graph of Figure 4.4

```
>> el=0.0013*10^-12; es=10*10^-12; Ee=50*10^-9; Ebf=5*10^-9;
n=1000; k=50; d=100; l=4000; r=250*10^3; Estart=0.1;
s=3;
[M,m]=meshgrid([1:10:190],[1:1:19]);
Ech_elect=l.*(Ee*(n/k)+Ebf*((n/k)-1)+es*d^2);
Enon_ch_elect=l.*(Ee*(1+k)+(k*Ebf)+es.*((M.*M)/(2*3.14159*k)));
Ech_frame=l.*(((n/k)-m)*Ebf+(el*d^4)+((n/k)-m+1)*Ee);
Enon_ch_frame=l*(Ee+es.*((M.^2)/(2*3.14159*k)));
f1=1./(((n/k)-m+1)*k);
f2=((n/k)-m)./(((n/k)-m+1)*k);
p1=((m.*Estart)-Ech_elect-Enon_ch_elect);
p2=(f1.*Ech_frame)+(f2.*Enon_ch_frame);
Nf=p1./p2;
ti=Nf.*(n/k-(m-1))*(l/r);
surf(M,(m*k*100)/n,ti);
xlabel('Network Diameter', 'FontSize',18)
ylabel('head-Set Size (%)', 'FontSize',18)
zlabel('Time for one iteration (sec)','FontSize',18)
```

MATLAB source code for the graph of Figure 4.5

```
el=0.0013*10^-12; es=10*10^-12; Ee=50*10^-9; Ebf=5*10^-9;
n=1000; M=100; d=100; l=4000; r=250*10^3;
Estart=0.1;
s=20
c=[1:2.5:50];
Ech_elect=l.*(Ee.*(n./c)+Ebf.*((n./c)-1)+es*d^2);
Enon_ch_elect=l.*(Ee.*(1+c)+(c.*Ebf)+es.*((M*M)./(2*3.14159*c)));
```

```
Ech_frame=l.*(((n./c)-s)*Ebf+(el*d^4)+((n./c)-s+1)*Ee);

Enon_ch_frame=l*(Ee+es.*((M^2)./(2*3.14159*c)));

f1=1./(((n./c)-s+1).*c);

f2=((n./c)-s)./(((n./c)-s+1).*c);

p1=((s.*Estart)-Ech_elect-Enon_ch_elect);

p2=(f1.*Ech_frame)+(f2.*Enon_ch_frame);

Nf=p1./p2;

ti=Nf.*(n./c-(s-1))*(l/r);

plot(c,ti);

xlabel('Number of clusters', 'FontSize',18)

ylabel('Time for one iteration (sec)', 'FontSize',18)
```

MATLAB source code for the graph of Figure 4.6

```
el=0.0013*10^-12; es=10*10^-12; Ee=50*10^-9; Ebf=5*10^-9;

n=1000; c=50; d=100; l=4000; r=250*10^3; Estart=0.1;

M=200;

s=[1:1:20];

Ech_elect=l.*(Ee*(n/c)+Ebf*((n/c)-1)+es*d^2);

Enon_ch_elect=l.*(Ee*(1+c)+(c*Ebf)+es.*((M.*M)/(2*3.14159*c)));

Ech_frame=l.*(((n/c)-s)*Ebf+(el*d^4)+((n/c)-s+1)*Ee);

Enon_ch_frame=l*(Ee+es.*((M.^2)/(2*3.14159*c)));

f1=1./(((n/c)-s+1)*c); f2=((n/c)-s)./(((n/c)-s+1)*c);

p1=((s.*Estart)-Ech_elect-Enon_ch_elect);
```

```
p2=(f1.*Ech_frame)+(f2.*Enon_ch_frame);

Nf=p1./p2;

plot(s,Nf);

xlabel('Cluster head-set size', 'FontSize',18)

ylabel('Number of Frames','FontSize',18)
```