Two Tier Cluster based Routing Protocol (TTCRP) for Wireless Sensor Networks

Author

Asif Usman Khattak
MS-07 (Software Engineering)

Supervisor

Dr.Ghalib Asadullah Shah
Assistant Professor

DEPARTMENT OF COMPUTER ENGINEERING
COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SCIENCES AND ECHNOLOGY
ISLAMABAD.

July, 2010

Two Tier Cluster based Routing Protocol (TTCRP) for Wireless Sensor Networks

Author

Asif Usman Khattak

2007-NUST-MS PhD-CSE (E)-01

MS-07 (Computer Software Engineering)

A thesis submitted in partial fulfillment of the requirements for the degree of

MS (Computer Software Engineering)

Thesis Supervisor:

Dr. Ghalib Asadullah Shah

Assistant Professor,

Department of Computer Engineering.

Thesis Supervisor Signature: ____

DEPARTMENT OF COMPUTER ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

AUGUST, 2010

ABSTRACT

Wireless Sensor Networks (WSNs) consist of thousands of small nodes having the capability of sensing, computation, and wireless communication. Many routing, power management and data dissemination protocols have been particularly designed for WSNs where energy utilization is an essential design issue. Since wireless sensor network protocols are application specific, therefore the focus has been given to the routing protocols that might be different depending on the application and network architecture.

Over the recent years, one of the most important problems in wireless sensor networks (WSNs) is to develop an energy-efficient and reliable routing protocol and provide network robustness. To achieve these objectives, we propose a new data dissemination protocol called Two Tier Cluster Based Routing Protocol (TTCRP) for reliable data delivery in WSNs. TTCRP uses resource rich cluster heads with dual channels to form the clusters for efficient data delivery. While it provides power control algorithm to connect the isolated low power sensor nodes with minimum required power that achieves the network robustness. Simulation results reveal that TTCRP is efficient and provides reliability in both uniform as well as non-uniform sensor nodes deployment
Keywords: Network Robustness, Clustering, Power Control Algorithm (PCA), and Wireless Sensor Networks (WSN).

UNDERTAKING

I certify that research work titled **"Two Tier Cluster Based Routing Protocol (TTCRP) for Wireless Sensor Networks"** is my own work. The work has not been presented elsewhere for assessment. Where material has been used from other sources it has been properly acknowledged / referred.

Asif Usman Khattak,

REG NO: 2007-NUST-MS PhD-CSE (E)-01

ACKNOWLEDGEMENTS

Thanks to Allah who enabled me to present this thesis. When I started preparing my thesis I was perplexed in choosing thesis topic. It is an important stage where everyone needs assistance and guidance. For this purpose, I went to Dr.Ghalib and discussed some topic related to the field of computer networks. All this credit goes to him who took keen interest in my work by properly guiding and giving me assistance and read my manuscript with patience and blessed me with a lot of suggestions.. I am most grateful to him. Without his help, encouragement and guidance I could do nothing. I enjoyed my thesis due to his friendly and cooperative nature.

After my theoretical work, when I was doing simulation of my work. I got stuck in some technical problems in NS-2(NETWORK SIMULATOR). I was in search of someone who helps me in some technical problems. I discussed some technical problems with one of my friend who helped me in it thoroughly. I would to like to say thanks to Mr. Muhammad Ahsan for helping me in technical problems. I would like to say thanks to Mr.Riaz Ahamd Khan, Mr.Adeel Shahzad and Mr.Fiaz Ullah khan for their encouragement and support in my thesis work. Finally I am grateful to all those who helped me during my thesis work.

DEDICATION

This thesis is dedicated to my parents, who taught me that the best kind of knowledge to have is that which is learned for its own sake. It is also dedicated to my respectable teacher ***Dr.Ghalib Asadullah Shah****, who taught me that even the largest task can be accomplished if it is done one step at a time.*

"The task of the excellent teacher is to stimulate 'apparently ordinary' people to unusual effort. The tough problem is not in identifying winners: it is in making winners out of ordinary people."

K. Patricia Cross

TABLE OF CONTENTS

LIST OF FIGURES

Number *Page*

LIST OF ACRONYMS

ACQUIRE	Active Query forwarding In sensor network
ADRP	Adaptive Decentralized Reclustering Protocol
APS	Ad-hoc positioning system
APTEEN	Adaptive Periodic Threshold sensitive Energy Efficient sensor Network
ATD	Analogue to digital
CC	Chessboard clustering
CPU	Central Processing Unit
DD	Directed Diffusion
DMSS	Disaster Management and surveillance System
EAR	Energy Aware Routing
GAP	Geographic adaptive fidelity
GOAFR	Greedy other adaptive face routing
GEAR	Geographic and energy aware routing
GROUP	Grid-clustering Routing Protocol
HCR	Hierarchical Clustering Routing
HEED	Hybrid Energy Efficient Distributed Clustering
HPAR	Hierarchical Power-Active Routing
LEACH	Low Energy Adaptive Clustering Hierarchy
MMSPEED	Multi path and Multi SPEED
MECN	Minimum energy communication network
MCFA	Minimum Cost Forwarding Algorithm (MCFA)
MANETs	Mobile Ad hoc Networks

PCA	Power Control Algorithm
PEGASIS	Power Efficient Gathering in Sensor Information Systems
QoS	Quality of service
SAR	Sequential Assignment Routing (SAR)
SPIN	Sensor Protocols for Information via Negotiation (SPIN)
SRPSN	Secure Routing Protocol for Sensor Networks
TEEN	Threshold sensitive energy efficient sensor network protocol
TTCRP	Two-Tier Cluster-Based Routing Protocol
UART	Universal Asynchronous Receive and Transmit
WSN	Wireless Sensor Network

CHAPTER 1

Introduction

1.1 Introduction to Wireless Sensor Networks

Wireless sensor network (WSN) consists of a group of sensor nodes which are homogenous and self-organized. Each sensor node has the capability of sensing the environment, processing on the sensed data and communication of data using radio frequency channel. The most fundamental job of wireless sensor networks is to sense the events of interest, collect particular data and then forward it to the destination. Various features and characteristics of sensor networks make them dissimilar from the conventional wired and wireless distributed systems. Conventional wired or wireless networks have adequate amount of resources like unlimited power, memory, fixed network topologies, enough communication range and computational capabilities. These various features make the conventional networks able to meet the communication demand [25, 26].

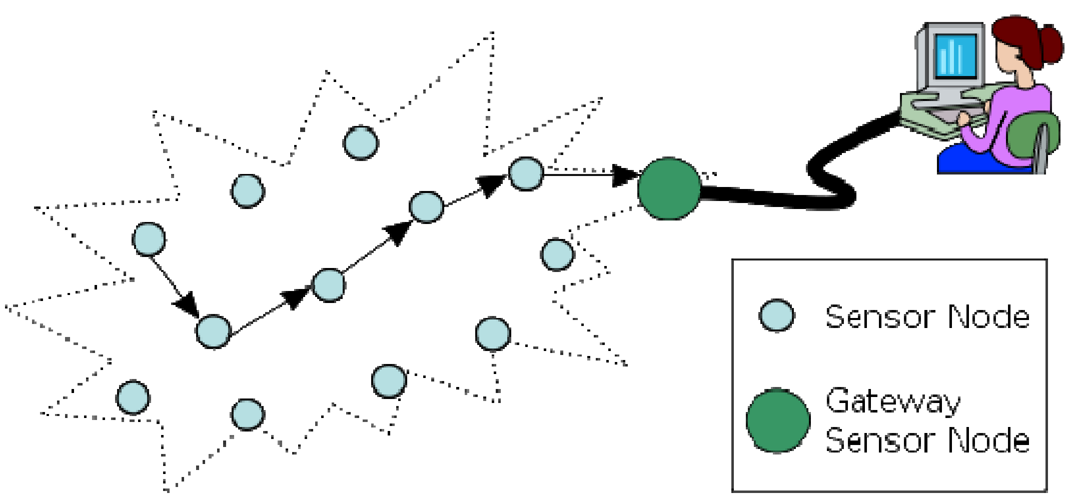

Figure 1: Typical WSN architecture [53]

WSNs are distributed systems which are limited in resources such as low energy, low bandwidth and short communication range. The essential features that make WSNs different from the conventional networks are; self-organizing capabilities, multi-hop routing, sparse deployment, short range communication, dense deployment, limited energy and memory, and also frequently altering topology due to losses and failures [32, 26]. The constrained resource nature and unpredictable network structure (sensor nodes which are spread out densely in a particular environment of interest) causes various design and communication challenges for WSNs. According to [25] "The challenges in the hierarchy of: detecting the relevant quantities, monitoring and gathering the data, assessing and evaluating the information, formulating significant user displays, and performing decision-making and generating alarms are massive." In general, the wireless sensor network operation is divided into two types; data acquisition and data reporting. Additionally, there is also a central sink which is responsible for WSN management, monitoring and control as shown in Figure 2 below.

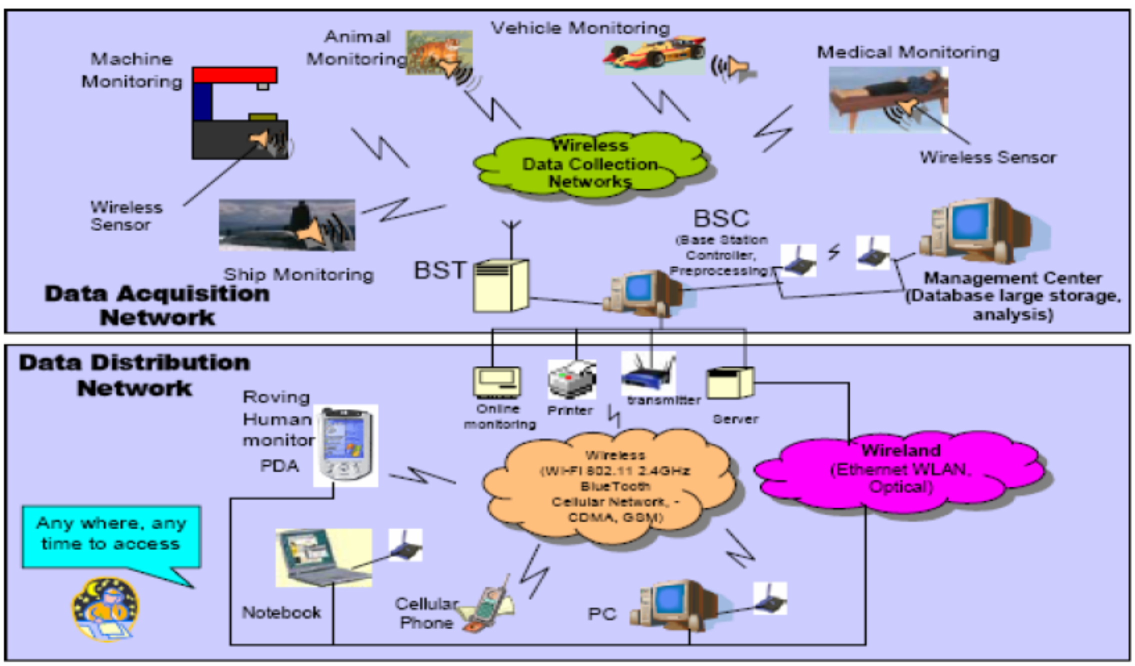

Figure 2: Wireless sensor network [25]

WSNs applications are based on the incorporation of contemporary technologies like CPU, sensor and Radio which performs Processing, sensing and communication. Thus it entails better understanding of contemporary network technologies as well as of WSNs hardware units for an effective WSN. Regardless of all these main challenges, the importance and significance of WSN cannot be ignored because of its diverse type of application domain [25].

1.2 Network Components of WSN

General WSN consists of main components which are the sensor nodes, the base station (sink) and the events of interest being observed.

1.2.1 Sensor Node and its Functional Units

In WSN, sensor nodes have the capabilities of sensing, processing and communicating data to the requested destination. The fundamental units in sensor nodes consists of sensing unit, its power unit, processing unit, communication unit and memory unit to carry out the operations shown in Figure 3 below.

i) Sensing Unit

Sensor nodes play a key role in sensor networks by forming a connection among the physical world and computation world. Sensor is a hardware device which is used to measure the change in physical state of a particular region of interest and then produces response to that change in state. Sensors sense the location of interest or the environment, collect raw data and convert that data to original data before forwarding it for additional processing. It takes the analogue data which are sensed data from an environment and converts it to digital data and sends it to the microcontroller for performing additional processing. Sensors are divided into different categories which are easily obtainable. It can be applied and used which depends on the nature of the intended operation. A typical wireless sensor node is a micro-electronic node that has less than 0.5 Ah and 1.2 V power source. The size and energy consumptions of sensor nodes are the key factors to be considered in selection of sensor nodes [32, 33, 34].

ii) Memory Unit

Sensor node uses this unit to store and accumulate both the data and program code. A sensor node normally uses Random Only Memory (ROM) in order to store data packets which are received from their neighboring (other) sensor nodes. And uses flash memory or Electrically Erasable Programmable Read Only Memory (EEPRM) to store the program code [32, 33, 34].

iii) Power Unit

Each sensor node needs power (energy) which is used for data computation and its transmission. A sensor node that consists of a power unit is responsible to distribute power to all its units associated with the node. The vital power utilization at node is due to performing computation and transmission of data. The date transmission is the most costly and expensive activity at sensor node in terms of power consumption. Typically, each sensor node is battery operated but it can also forage energy from the environment through solar cells [32, 33, 34].

iv) Processing Unit

Each sensor node has a microcontroller and each microcontroller consist of a processing unit, Universal Asynchronous Receive and Transmit (UART) interfaces, memory and converters (analogue to digital, ATD) timer to perform the processing responsibilities. The processing unit is actually responsible for data acquirement, processing of incoming and outgoing information, implementing, adjusting and regulating routing information which considers the performance conditions of the data transmission [32, 33, 34].

v) Communication Unit

Communication unit is an important unit in any type of sensor node. Nodes use different radio frequencies and optical communication for achieving the networking objective. This job is controlled by radio units that use electromagnetic spectrum to communicate the information to their respective destinations. Generally each sensor node transmits data to other node or sinks (base station) directly or using multi-hop routing (flat routing) [32, 33, 34].

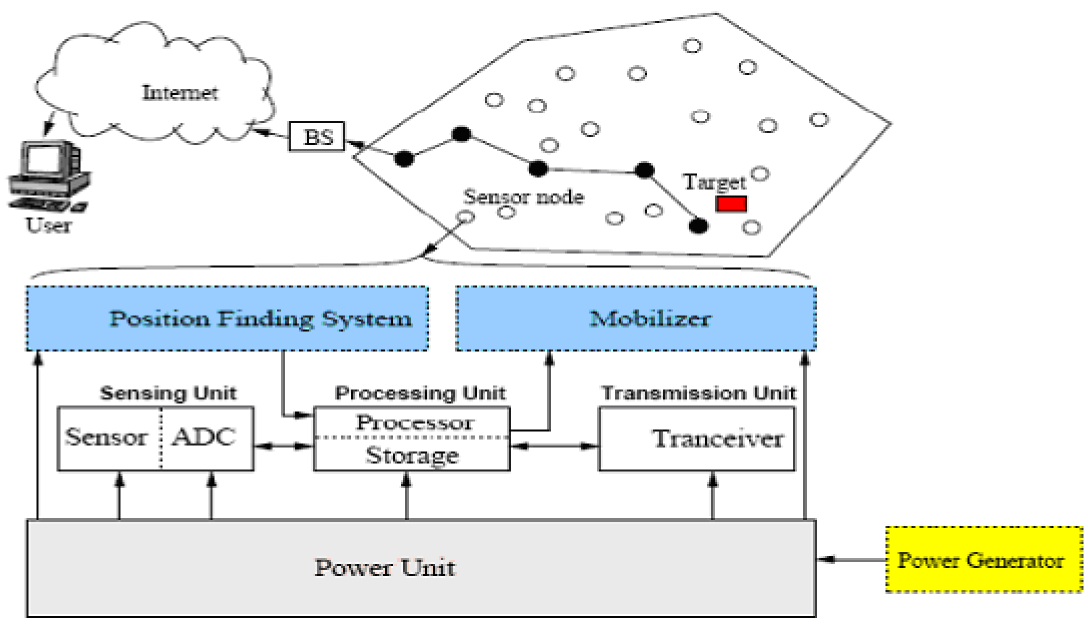

Figure 3: Components of sensor node [34]

1.2.2 Base Station (Sink)

The sink node (some time base station or cluster head) which acts like an interface between the computational world and external world. In other words sink node is an interface between the sensor network and management center. It is usually a resource rich node which have unconstrained computational capabilities and high energy supply. A network consists of single or multiple base stations. Practically, in order to decreases network delay and achieve better performance using robust data gathering, the use of multiple base stations are required. Base station can be stationary or dynamic depending on the network architecture. The dynamic base stations can affect the performance of routing protocols significantly because of its varying position from one location to another which will be not apparent to all the sensor nodes in a network. Despites the mobility of base stations there are some other characteristics of sink like its coverage, presence and the number of nodes cause routing challenges for routing protocols of WSN which are briefly explained in chapter 2[34, 35].

1.3 WSN Operation

Commonly, WSN operation entails communication between sensor node and sink. The sensor node senses environment of interest, performing some computation (if essential) and then reports gathered information to the sink. If sink is attached with some actuator which generates the alarm for human interference in case of an event of interest in a critical region [34].

1.3.1 Communication Model

Though sensor nodes are the same devices but their characteristics fluctuate with the network structures and architecture. Sensor node coverage, deployment, transmission power, computational task, reporting of data, addressing and communication samples significantly affected the routing protocol operation both at sensor nodes and at sink. Routing protocols that are used for WSN communication supports different kind of communication like unicast (one-to-one), multicast (one-to-many) and reverse-multicast (many-to-one) in the following ways [34].

i) Node-to-Node

In a multihop or flat communication data passes from sensor nodes using intermediate nodes towards the destination. In node to node communications data is passed from one node to another until the destination is reached. In general, this type of communication is not essential or needed in WSN communication.

ii) Node-to-Base Station

Node to base station communication is used whenever sensors node need to send responses back to the sink. It is a reverse-multi path communication in which more than one node can communicate to the sink node directly or indirectly. This type of communication can also be unicast if multiple base stations are involved in the network or there is a particular node (group leader or super cluster head), who is responsible to collect sensed information and transmit it to the sink node or base station [34].

iii) Base Station-to-Node

When base station needs to request data from sensor nodes this pattern of communication is needed. In general, the mode of communication is anycast (one-to-many) in which any sensor node that have the requested data can act in response to the base station. This type of communication can also be unicast or multicast if the identification of nodes is distinctive by their IDs or their locations etc [34].

1.4 Classification of Sensor

Sensor node can be categorized based on different aspects such as technological aspects, their output signals and sensor materials, detection means and field of application. Though diverse classification is required while looking on its application side but can be classified in to following categories [36].

1.4.1 Active Sensors

To do the measurements active sensors stimulate the environment. Active sensors are for example laser scanners, seismic sensors, infrared sensors, sonar's etc [37].

1.4.2 Passive, Directional Sensors

Without disturbing the environment, these types of sensors can monitor the environment. For Examples thermometers, humidity sensors, light sensors and pressure sensors and so on [37].

1.4.3 Narrow Beam Sensors (Passive)

To measure the environment (medium), these types of passive sensors require a clear direction. For Examples camera and ultrasonic sensors etc [37].

1.5 Classification of Sensor Network Applications

Wireless sensor networks can be deployed for numerous applications which depend on different requirements according to [37]. Such as data delivery requirements, application type and its objectives. The demands of applications are different from each other according to application nature. Some applications are intended of only data collection but not in robust data delivery while in some applications delay cannot be tolerated because of critical interest about the event. There are various different application classes with various different transmission demands. Hence it is essential to categorize applications of WSNs in classes for understanding of their nature and requirements. In general, WSN applications can be categorized into following four classes.

1.5.1 Event Detection and Reporting

In these types of applications sensor nodes are used rarely. These sensor nodes are active (come to life) only when a particular event of interest occurs otherwise these nodes remain inactive most of the time. When an event of interest occurs, each sensor node detected the event and then sends event report to the base station which contains location information about the event as well as information about the nature of the event. The most important in these applications are the reliability and delay which is the major concern about the event. The most important challenge in this sort of network at application level is to reduce false reporting about the event as well as to decrease delay. Also routing of date about the event to the base station is a major design issue from networking point of view. Examples of such type of applications are [38, 39, 26 and 40]. As one of such application scenario about earthquake or seismic activity in an active volcano is depicted by the below figure 4.

- Intruder detection in military surveillance application
- Observation and Quality check at product line/ anomalous behavior
- Detection of forest fire/ Floods
- Detection of eruption/seismic activity or earthquake
- Detection and observation of ocean environment

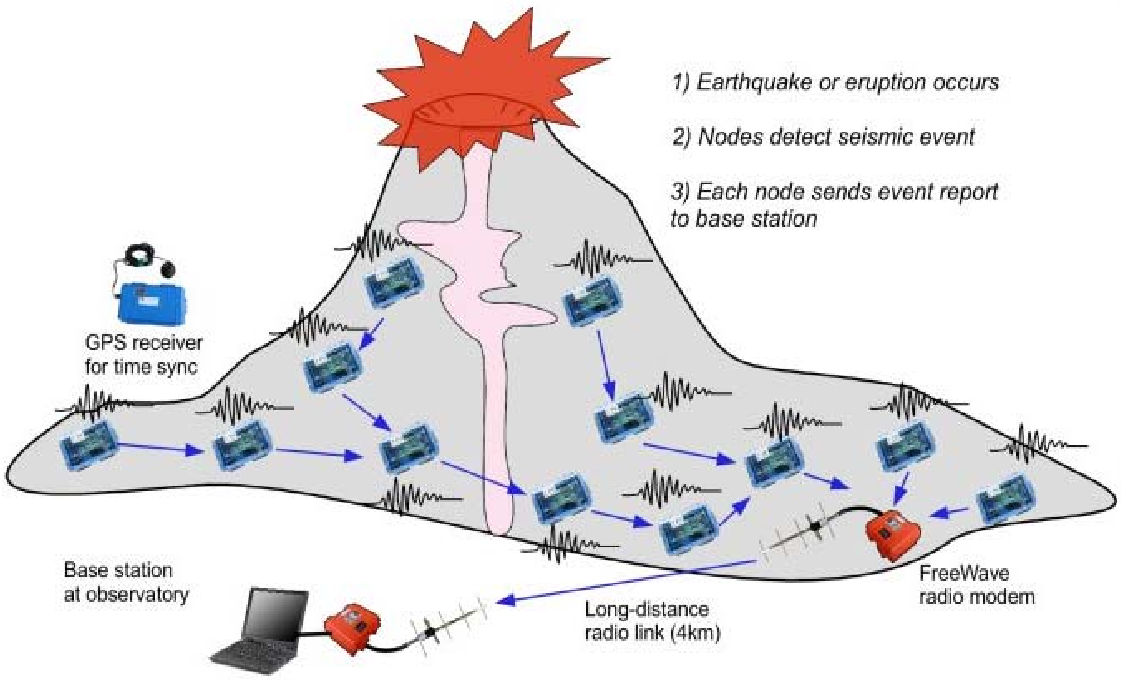

Figure: 4 Application scenario about seismic event [21]

1.5.2 Data Gathering and Periodic Reporting

In this class of applications the functional behavior of sensor nodes is of continuous and constant nature. In these types of applications some activity is recorded by continuously monitoring it and then reports to the base station separately just like point-to- point communication. But when the network is very large, base station involves in distributed computation on collected data instead of individual node reading in order to avoid traffic load at base station. On some occasion when these sensor nodes are attached with actuators. The base station might require storing the geographical information of the sensor nodes in the particular area of interest. For example an application likes monitoring of humidity in a glass house. Critical requirement of these types of applications is well-organized and efficient consumption of energy.

Examples of these types of applications are; [38, 39, 26 and 41]

- Monitoring humidity, temperature and light etc
- Monitoring of environmental conditions
- Monitoring of Home/office/factory smart environments
- Monitoring of Health like applications etc

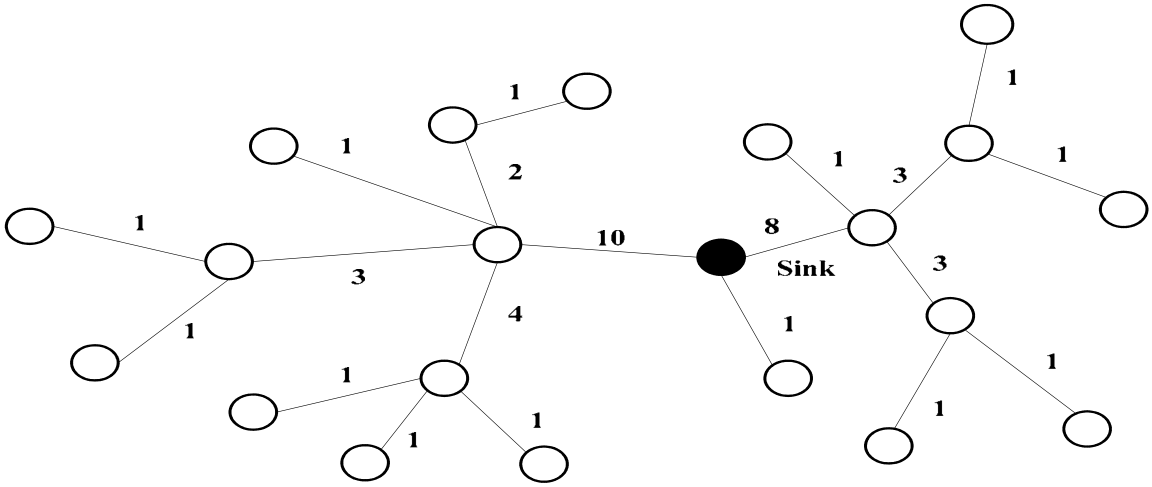

Figure 5: WSN node topology example [41]

1.5.3 Sink-Initiated Querying

This class of WSN applications has the extra feature of base station querying in addition to monitoring. In this scenario, instead of periodic reporting of the individual node, base station has the capability of sending a query to a set of sensor nodes for their reading about the environment of interest. This permits the base station to collect information of diverse locations and also assists in verifying the measurements to take a decision such as triggering an actuator or raising an alarm.

Some of the examples of these type applications are; [38, 26, 42].

- Structural health of building or environmental control in buildings
- Monitoring and controlling of soil condition
- Detection of Biological attacks
- Forecasting and monitoring of weather
- Fire alarming like applications

1.5.4 Tracking Based Application

The WSN application of this class has some of the characteristics and features of the earlier three classes. Tracking applications entail both the detection and location information. Whenever a sensor node detected a target at any location, it has to report the base station quickly where accuracy of target is the major concern. Then the base station may require sending queries to the particular group of nodes for getting the location information of the particular target. Sink also assist to verify the measurements taken by individual node about the target detection and recognition. The decision of prompting actuator or raising an alarm for human interruption is based on the readings or signals received by this group of sensor nodes.

Examples of these types of applications are; [32, 34, 44].

- Targeting in intelligent ammunition
- Tracing and Tracking of doctors and patients or rescue team in hospital
- Tracking and tracing of inhabitant in a building
- Tracing and tracking of animals in forest
- Tracking, monitoring and controlling the people in park and building

1.6 Motivation

As stated earlier, the sensor nodes due to their small form factor have limited power. In order to prolong the life of the wireless sensor networks, the routing protocols apart from being robust and scalable, needs to be highly energy efficient.

The size and cost of the sensor nodes are the fundamental design goals that limit the node resources such as processing power, memory, energy, transmission range and bandwidth. However, the energy constraint is the most critical issue that determines the life time of the node. The life of node relates to the usable energy reserve, where a node is supposed to die if its energy goes down to the energy required for the transmission of information or data. Therefore, the design of communication protocols, particularly routing and MAC, are mainly influenced by the energy conservation of nodes in order to prolong the life of the network [1], [2].

Apart from the energy conservation, it is also necessary to balance the traffic load among all the nodes. By distributing traffic load, eventually the lifetime of the network is maximized. As sensor nodes have energy constraints. Therefore most of the nodes die soon during the observation of a particular region due to a large amount of energy consumption. Reliability in WSN is defined in terms of number of packets received at the destination [4] in certain time period.

According to communication energy model [14], hierarchical routing protocols have better adaptability and energy conservation than plane routing protocols. Cluster-based optimization algorithm and the transmission path of cluster head are designed for hierarchical routing protocols, which must consider the important issues and the key parameters about affecting energy dissipation of nodes and improving the network lifetime in WSN.

Network robustness is an important issue in WSNs that is a major challenge in application like DMSS. Since the sensing coverage of sensor nodes are generally limited [4], large scale region sensor nodes require large scale deployment. As the nodes are randomly distributed in the critical region, in which many nodes can not reach or access the base station or other nodes in the area of interest causing disconnectivity in the network. In such cases the out of range nodes are not functioning due to fixed transmission range. In addition existing clustering techniques use only single channel for communication which may result into delay and congestion.

In a cluster based routing protocol, sensor nodes are grouped in the form of clusters. In each cluster, a node is elected as a cluster-head, which collects data from its member nodes and route it to the destination sink. In some applications, cluster-heads also perform data aggregation to save energy and then sends the aggregated data to the destination or sink. The cluster-based approach is highly adaptive to network dynamics unlike the traditional tree-based routing protocols [2]. A number of clustering techniques have been proposed in the literature so far [3], [7], [5], [8], [11], [12]. However, the existing protocols can perform well in uniform nodes deployment, but they are unable to provide efficient and robust routing in non-uniform nodes deployment. A detailed review of the existing clustering techniques is provided in [2].

The work reported in this thesis aims at designing a two-tier cluster based routing protocol(TTCRP) which configures the nodes in the form of clusters at two levels. At first level sensor nodes join pre-designated resource rich cluster heads. These cluster heads form a second level of clusters to deliver data to the base station. We assume that the cluster heads are equipped with dual channels in which different channels are used for communication at both tiers. Thus cluster-heads receive data from heir members at one channel and use second channel to send it to the base station through other cluster heads. Moreover, the proposed scheme implements a power control algorithm to allow the isolated sensor nodes as well as cluster-heads to dynamically change their transmission power for connecting sensor nodes with unreachable clusters and hence provides network robustness.

1.7 Statement of the Problem

Advances in semiconductor technology have resulted in the various Sensor Aid networks for disaster monitoring and surveillance. In this thesis, Cluster based routing protocol is proposed that is used by Disaster management and surveillance system (DMSS) using wireless sensor networks to handle various issues related to the disaster. The DMSS will continuously monitor the vital signs of victims and deliver the critical information to first responders as well as handle, tracing and tracking rescue teams, and victims. It will also monitor the causes of disaster such as fire, earthquake, and gas leakage in civil structures like buildings and bridges.

Generally, such telemetry service is restricted to the close proximity of disastrous scenes or within the hospitals due to the short range of wireless sensors. These constraints can be overwhelmed if the wireless sensors are provided with connectivity anywhere anytime. This project aims to exploit the ubiquitous networking environment to provide access to buildings, hospitals, disaster management offices and other relevant departments. In essence, DMSS will provide a spontaneous infrastructure to monitor and manage: structural health, earthquakes, fire detections as well as to trace the victims and rescue team members and proper deployment and utilization of those members.

Routing is the most challenging job in WSNs because of their distinctive characteristics and unique features which makes it unique and special from other wired and wireless networks such as cellular or mobile ad hoc network (MANETs). There is no such routing protocol which is used particularly for all the applications such as civilian and military applications. The DMSS that handles all the issues related to the disaster .At the disaster scene the most important is the flow of critical information between the rescue teams and fire fighters. As well as before the disaster is going to be happen , it is extremely necessary that the communication between the rescue teams are so fast so that as soon as possible to overcome the loss and damage of natural and human resources. Therefore, it is required that routing protocols should have the capabilities to handle these characteristic for reliable and efficient communication. Different routing methods have been proposed to address the routing problems in WSNs taking into account WSNs network architecture and application demands. The main emphasis of this work is to design the efficient and reliable routing protocol which is best suited for the DMSS.

1.8 Objectives

The basic objectives of our proposed protocol are the following.

- To provide an energy efficient cluster based routing protocol to achieve reliability.
- To provide power control algorithm that is used by routing protocol to achieve network robustness.
- To provide reliability in uniform as well as non- uniform sensor nodes deployment.
- To develop and design the cluster based routing protocol which could be used for any DMSS like applications to achieve the reliability in sparse deployment.

1.9 Thesis Organization

This thesis is organized as follow:

- Chapter I: discusses problem statement, brief introduction to wireless sensor networks, motivation, solution to cluster based protocol issues and then objectives that needs to be achieved.
- Chapter II: introduces to cluster based routing protocols, and gives the detail about the past work in the area of cluster based routing protocol for WSN's.
- Chapter III: this chapter is about proposed approach, introduction to TTCRP, and the details of power control algorithm (PCA).
- Chapter IV: discusses the experimental results of the proposed approach and comparison with other cluster based routing protocols, then conclusion and future work in this area.

CHAPTER 2

Related Work

2.1 Introduction to Routing Protocol for WSNs

As discussed in chapter 1, the design of routing protocols for WSN is a hot issue of intense research as the applications centered on WSN requires information of both quality and quantity in nature delivered to the end-users. There is an observation about different protocols that they work better in different environments/applications. The issue of the effective utilization of energy resource has also been addressed extensively in the literature. This chapter mainly deals with related work and the underlying concepts which form the foundation of energy aware routing protocols for WSNs.

2.2 Classification of Routing Protocols in WSN

In order to overcome the shortcomings of the recourse constraint nature of the WSNs, different routing protocols have been designed. According to the network structure or intended operations, the deployed WSN can be differentiated. Due to which, routing protocols for WSN needs to be classified based on the nature of WSN operation and its network architecture. Routing protocols could be subdivided into two broad types; network architecture based routing protocols and operation based routing protocols [33, 35].

2.2.1 Architecture Based Routing Protocols

Routing protocols are categorized based on the structure of network which is very critical for the mandatory operations. According to this category, the protocols are further subdivided into three subcategories based on the functionalities. These routing protocols are [33, 35].

- Flat routing
- Hierarchical routing
- Location-based routing

i) Flat Routing

Whenever a large number of sensor nodes in the network are needed then flat routing is used. The reason is that every sensor node plays the same role. As the number of sensor nodes are very large so it is not possible to assign a specific Id to each and every node in the network. It gives rise to data centric routing approach, where Base station sends query to a set of particular nodes in a particular region of interest and then waits for its response. Few of the examples of Flat routing protocols are; [33, 44, 35].

- Directed Diffusion (DD)
- Energy Aware Routing (EAR)
- Sequential Assignment Routing (SAR)
- Active Query forwarding In sensor network (ACQUIRE)
- Minimum Cost Forwarding Algorithm (MCFA)
- Sensor Protocols for Information via Negotiation (SPIN)

ii) Hierarchical Routing

Hierarchical routing is the best option whenever network scalability, efficient and reliable communication is required. It is also known as cluster based routing. Hierarchical routing is an energy efficient method. In hierarchical routing, high energy nodes are elected randomly for processing and sending of data. As the basic purpose of low energy nodes is to sense the event and then send information or data to the cluster heads. Hierarchical routing pays a great contribution to the network scalability, lifetime and minimum energy due to this property. Few examples of hierarchical routing protocols are; [33, 44, 35].

- Power efficient gathering in sensor information systems
- Hierarchical Power-Active Routing (HPAR)
- Minimum energy communication network (MECN)
- Threshold sensitive energy efficient sensor network protocol (TEEN)

iii) Location-Based Routing

Location based routing is a network architecture where sensor nodes are scattered in a random manner regarding the relevant area of interest and mostly identified by the geographical position of the sensor nodes where they are distributed for deployment. Sensor nodes are located mostly using GPS. The received signal strength from these nodes and coordinates are computed by exchanging the information among their neighboring nodes due to which the distance among the sensor nodes is predicted. Examples of Location-based routing networks are; [33, 35]

- Ad-hoc positioning system (APS)
- Geographic adaptive fidelity (GAP)
- Greedy other adaptive face routing (GOAFR)
- Sequential assignment routing (SAR)
- Geographic distance routing (GEDIR)
- Geographic and energy aware routing (GEAR)

2.2.2 Operation Based Routing Protocols

Based on the functionality, WSNs applications are classified. Therefore Routing protocols are categorized regarding their operations in order to meet these functionalities. The purpose of this categorization is to achieve optimal performance and to save the insufficient resources of the network. Operation based routing Protocols are classified as following:

- Multipath routing protocols
- Query based routing
- Negotiation based routing
- QoS based routing
- Coherent routing

i) Multipath Routing Protocols

As it is obvious from its name, there is an option for multiple path selection for a message to reach the destination which decreases delay and increases the overall network performance. Due to increased overheads, the network reliability is attained. While network routes are kept active due to transmission of periodic messages and therefore consume greater amount of energy. Examples of Multipath routing protocols are: [33]

- Sensor Protocols for Information via Negotiation (SPIN)
- Multi path and Multi SPEED (MMSPEED)

ii) Query Based Routing Protocols

Query based routing protocols supports sending and receiving queries for the data. The destination node sends relevant query from a node using the network. The node with this relevant query matches it and then sends back to the target node which has initially sent the query. In general, the query based on high level languages. Examples of Query based routing protocols are as following: [33]

- Directed Diffusion (DD)
- COUGAR
- Sensor Protocols for Information via Negotiation (SPIN)

iii) Negotiation Based Routing Protocols

Negotiation based routing protocols is the class of protocols which uses high level data descriptors to remove redundant data communication through negotiation. This class of routing protocols makes intelligent decisions either actions based on facts or for communication for example how much resources are offered. Examples of negotiation based routing protocols are: [33, 46]

- Directed Diffusion (DD)
- Sequential assignment routing (SAR)
- Sensor Protocols for Information via Negotiation (SPAN)

iv) QoS Based Routing Protocols

In QoS based routing protocols, network requires to have a balance strategy for the QoS of applications. In this scenario the application can be delay sensitive. Consequently to achieve QoS metric network have to look both for its energy consumption and delay. Therefore to achieve QoS, the cost function for the preferred QoS also requires to be considered. Examples of such routing protocols are: [46, 33]

- Multi path and Multi SPEED (MMSPEED)
- SPEED
- Sequential assignment routing (SAR)

v) Coherent Data Processing Routing Protocol

Whenever energy-efficient routing is needed, Coherent data processing routing is used. In this case, sensor nodes do least amount of processing (in general, time-stamping and suppression etc) on the data which is received locally before sending to other nodes.

Then the date is sent to other nodes which are called aggregator for further processing known as aggregation [33, 47]. Data processing in non-coherent processing involves three stages. In the first stage target is detected, its data is collected and preprocessing of the sensed data takes place. In stage 2 where sensor nodes show its intention to their neighboring nodes. Lastly, in step 3 a core (center) node is elected for additional refined processing. Hence central node must be rich in resources such as having enough energy resources and computation abilities [33].

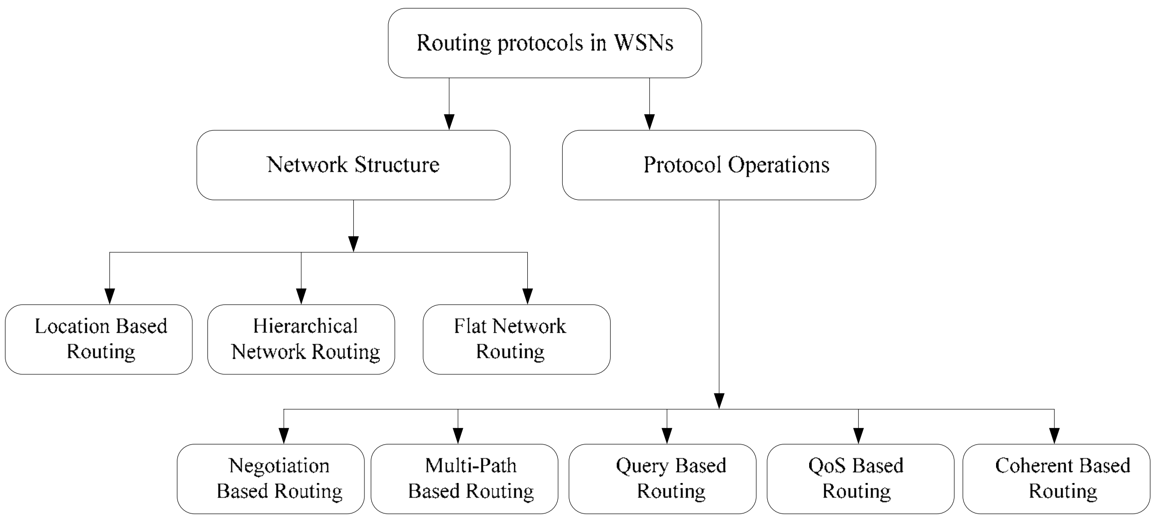

Figure 6: Routing Protocol in WSN: A Taxonomy [41]

2.3 Low Energy Adaptive Clustering Hierarchy (LEACH)

LEACH [16] is the first cluster based routing protocol in WSN which provides the basis for the researchers to work on clustering schemes. In LEACH, cluster heads are elected randomly and the clusters are reformed in certain rounds to evenly distribute the energy load among the sensor nodes in the network. Each cluster head accumulates data from its member nodes, aggregates it and then directly sends the aggregated data to the sink. This protocol involves the following steps:

STEP 1: Base station collects the status of all the nodes.

STEP 2: Base station or sink selects the cluster heads (CH).

STEP 3: Sensor node chooses the best CH among the cluster heads

STEP 4: Date distributes and disseminates to the cluster heads.

STEP 5: Cluster heads disseminate the aggregated data to the base station.

The main drawback of the LEACH scheme is that it does not suit for the non-periodical event delivery but it works well in periodical event delivery like applications. In addition, clusters can also be created with in the range of each other causing uneven load distribution among the sensor nodes in the network.

2.4 Power Efficient Gathering in Sensor Information Systems

Lindsay and Raghavan [48] proposed Power Efficient Gathering in Sensor Information Systems (PEGASIS) that attained about 100-300% enhancement over a LEACH protocol over a range of percentages of sensor nodes dying out in different sizes of network . In this protocol, when sensor nodes transmit and receive the data from different number of sensor nodes then they form a chain to transmit and receive data. Each sensor node transmits or receives data from a neighbor node and a single node is selected as the cluster head or leader from the chain which is responsible for forwarding the data towards the sink. Each sensor node when receives the data, then aggregates it including its own data, after that then transmits it further and so on. The aggregated data is finally sent to the sink. The chain is formed in a greedy manner towards the sink.

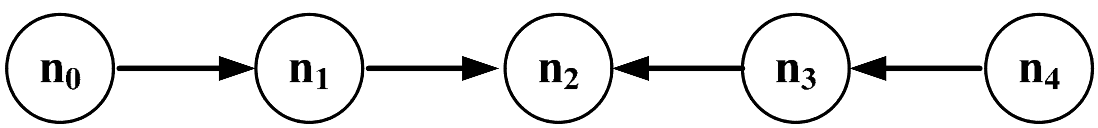

Figure 7: PEGASIS

Figure 7 demonstrates that node-0 sends out data to node-1. Node-1 collects data from node-0 and then aggregates it and includes its own data, then sends out the aggregated data to the node-2. In the same way, node-4 transmits its sensed data to node-3. And node-3 then aggregates data to its own data and then sends out the aggregated data to the node-2. The node-2 which lastly aggregates all the received data from different nodes including with its own data and transmits it to the base station.

2.5 Threshold sensitive Energy Efficient sensor Network

Threshold sensitive Energy Efficient sensor Network (TEEN) is proposed by Manjeshwar and Agrawal [49] is in fact an updated and modified version of LEACH scheme. The modification and change which was proposed in this scheme was to transmit two attributes from the cluster heads towards the sensor nodes which are namely, (i) Hard Threshold and (ii) Soft Threshold. Hard threshold was actually the absolute value of the sensed parameter beyond which the value must be sent out to the cluster head (CH). While, Soft threshold was the value of the sensed parameter beyond which the sensor node must becomes active through activating its transmitter. When it was expected that the data to be transmitted from the source nodes to the sink less frequently than the data being sensed, therefore this protocol is much more energy efficient and more effective then LEACH. But this protocol has some shortcomings which cannot be ignored. First, if the required threshold is not achieved then the data is not communicated from the sensor nodes to the sink. Second, there were no such messages which are used to inform the sink or base station that whether the node has gone dead due to which data is not received at the sink or the data is not significant and critical enough to be reported to the sink. Therefore, the reliability of important data reportage becomes an important problem that needs to be addressed.

2.6 APTEEN

Adaptive Periodic Threshold sensitive Energy Efficient sensor Network APTEEN [50] was actually designed to remove the shortcomings of TEEN. APTEEN adds an additional attribute to the packet which is sent by the cluster heads to their member nodes in the cluster. This protocol not only contains the thresholds but also it contains the maximum interval between the two packets. This amendment and modification which is made to APTEEN, makes this protocol usable and functional even by time driven networks. In APTEEN, the data is sent periodically which makes it easy for the sink or base station to know that the sensor node is dead or alive and it has failed to send the data. The major issue concerned with this protocol is that the data is not communicated between the sensor node and the sink until the required threshold is not acquired.

2.7 Hybrid Energy Efficient Distributed Clustering (HEED)

HEED [13] is a distributed cluster based routing protocol which uses a metric of hybrid of available energy and communication cost to elect the cluster head nodes. HEED does not select cluster heads (CH) randomly unlike LEACH and the probability that the two sensor nodes which are within the transmission range of each other's becoming CHs are very small. Furthermore, it is not assumed that the energy consumption will be uniform for all the sensor nodes in the whole network and therefore it is adaptive to the unpredicted traffic pattern. The HEED has a number of drawbacks. The major issue concerned with HEED is the unfair distribution of traffic load among the sensor nodes in which cluster heads which is closer to the sink not only forwards the data traffic from its own members but also forwards from the farther clusters toward the sink. In addition the routing of data is based on greedy approach (greedy forwarding) due to which energy holes are created near the sink. The fair distribution of traffic load is dealt in [10] that form clusters which have unequal size of clusters such that the clusters which is closer to the sink are smaller in size because they also forwards the traffic of farther clusters to the base station. Bandyopadhyay et. al. [18] proposes an energy efficient hierarchical cluster based routing protocol. The purpose of this protocol is to maximize the network lifetime. And k-hops nodes are permitted to join a cluster in this multihop clustering scheme. Likewise, each cluster-head that has data to send towards the sink within h-hops. Thus, the optimal values of k and h are decided to maximize the lifetime of the network.

2.8 Adaptive Decentralized Reclustering Protocol (ADRP)

Chih-Yu et al [3], propose a randomized, decentralized algorithm for re-clustering of the sensor nodes in an ad hoc network which uses the criteria of random waiting timer and a neighbor-based for the formation of clusters automatically. In adaptive decentralized reclustering protocol (ADRP), sensor network operations are separated into two phases; initial phase and data cycle phase. In the initial phase of ADRP, all the sensor nodes in the network send their information to the base station or sink for the formation of clusters. Sink forms the cluster using the received information from the sensor nodes and then sends the cluster-head IDs back for every node. The node having the node IDs are selected as next cluster-heads (CH) in cluster reformation process. During the data cycle phase, all the cluster-heads send TDMA schedule to their member nodes for the data delivery purpose. This protocol has a number of drawbacks. First, it incurs high overheads due to the exchange of information between the sensor nodes and the sink in forming clusters. Second drawback is that without knowing the future state of the nodes, the future cluster-heads for reformation of clusters are determined as well as the traffic pattern causes in inefficient and ineffective clustering.

2.9 Hierarchical Clustering Routing (HCR)

Sajid et al [51] propose a hierarchical Clustering Routing (HCR) which is an extension of low energy adaptive clustering hierarchy (LEACH). In this protocol, a set of associates is used for the management of each cluster. The clusters which are energy efficient are maintained and used for a longer time. This protocol uses heuristics based approach for the identification of energy efficient clusters in the network. In a variation of this protocol, cluster is formed by the base station or sink node. For the generation energy-efficient hierarchical clusters, it uses Genetic Algorithm (GA). The sink node broadcasts the GA-based clusters configuration. When it is received by the sensor nodes then the whole network is configured in a similar way. For the purpose of continuous and constant monitoring like applications, the simulated results of HCR show that it is more energy efficient and reliable than the conventional and existing cluster based routing schemes. The major shortcoming of this protocol is that the base station is involved in cluster formation process which incurs high overhead due to exchange of information between the sensor nodes and the sink.

2.10 Secure Routing Protocol for Sensor Networks (SRPSN)

Tubaishat et al. [52] propose an energy efficient level based hierarchical routing protocol. In addition, it designs a Secure Routing Protocol for Sensor Networks (SRPSN) that makes available security in opposition to different sorts of attacks and gives assurances that packets arrive at the sink from the source nodes even in the existence of intermediate adversaries. In this protocol, sensor nodes are grouped into hierarchy to divide the network into different levels. The categorization of sensor network into different levels in the hierarchy is based on the number of sensor node's neighbors. A node which has the large number neighbor nodes will be given higher level in the hierarchy. Each sensor node broadcasts a hello message to determine the identity (node ID) and number of neighbor nodes (NBR). To form the different groups or clusters in the network, the sensor nodes exchange the node IDs and NBRs for a certain round. The sensor nodes that have the maximum number of neighbor nodes become cluster heads. The responsibility of cluster heads is to aggregate the data and then forwards the aggregated data. When the cluster heads receives the data, it filters and aggregates the data and then sends the summary of the data to the root nodes. Each sensor nods uses a secret key with the base station for the purpose of security and protection of data. This protocol is divided into two phases: first secure route is discovered and second is forwards the secured data (secure route discovery and secure data forwarding). The major issue concerned with this scheme is that non-uniform clusters are formed due to which sensor nodes have the higher number of neighbors soon becomes out of energy and burn out fastly due to higher burden of traffic on that nodes.

2.11 Grid-clustering Routing Protocol (GROUP)

In GROUP [24], the whole sensor network is divided into various clusters dynamically. In each cluster, a sensor node is selected to act as the cluster head (CH). A virtual cluster grid is constructed by all the cluster heads. Sink transmits all the data queries to the sensor nodes through the selected cluster heads. The sensor node which receives the query, will forward the data back using the same route towards the sink through cluster heads. According to the application, specific data aggregation strategy can be used by the cluster heads.

Initially the sensor network is formed by deploying the sensor nodes in the field. The sink nodes which are present in the network will select one sink node to act as the primary sink (PS). The primary sink is responsible to start the process of cluster grid construction which is based on their location information. To have a minimum duration of the grid construction process, the sink which is closer to the center of the network is elected as the primary sink. Periodically PS broadcasts a GS-election command for the cluster grid construction using its radio coverage range. The basic objective of GS-election command is the selection of a Gird Seed (GS) from the sensor nodes which are neighbors of PS. When the one sensor node is selected as a GS then the new GS broadcasts a CH-election packet which contains the location information of GS and the size of cell to their neighbors for the election of four cluster heads. The sensor nodes which receive the CH-election packet, will also store the most recent received CH-election packet in order to get recovery from failure. Each cluster head forward the CH-new packet to the GS and stores in its cache information about the GS as its upstream cluster head. After that like the GS does, the new cluster head broadcasts a CH-election packet for the election of its downstream cluster heads. The selection of cluster head is repeated until all the cluster heads in the network are elected. And the duplicate CH-election packets which is identified by its sequence number, will be dropped .The major drawbacks concerned with the GROUP is that first it incurs high overhead because sensor nods also take participation in the selection of cluster heads and second drawback is that the data dissemination nodes are also involved in communication with each other directly as well as it's a multi hop routing protocol which is inefficient in terms of energy and delay.

2.12 LRS AND CC

LRS [31] takes the name of protocol from its author's names (Lindsey, Raghavendra and Sivalingam). LRS is a data gathering routing protocol which uses the energy*delay metric. It is a chain-based 3–level hierarchical protocol in which sensor nodes are primarily grouped into clusters which are based on their distances from the base station. Initially at the lowest level of the hierarchy a chain is formed among the sensor nodes due to which clusters are formed. In the cluster, data is gathered and then it moves from one node to another node, and get combined the collected data. Lastly the aggregated data arrives at the cluster head in the chain i.e., a designated leader. The cluster heads or designated leaders from the prior level are grouped into clusters to form one or more chains at the next level of the hierarchy. The data is then collected and aggregated and then forwards to the designated leader in each chain in a similar fashion. In this protocol a cluster head is selected periodically among the sensors to resolve the problem of uneven energy consumption in cluster heads. Although, this method has drawback of overheads due to periodic re-clustering. In addition, this scheme does not solve the problem of uneven energy consumption (which is due to many-to-one traffic pattern) by rotating the cluster-head in the chain. The sensors which are near to the base station have heavier load than other sensor nodes due to which such sensors have limited lifetime.

In Chessboard clustering (CC) [23] method, the sensor nodes is installed with a chessboard. The whole network is partitioned into a number of tiny cells which are equal in size, and all neighboring cells are highlighted with different colours which are white or green colors. In CC sensor network consists of two types of nodes which are the powerful high-end sensors (H-sensors) and the low-end sensors (L-sensors). CC is based on the assumption that both H-sensors and L-sensors are deployed uniformly and randomly in the field of interest. Each sensor has location information due to which each H-sensor can find out whether it is in a white cell or it is in a green cell. In the initialization phase, the H-sensors are active only in white cells while the H-sensors which are in green cells turn themselves off. At the same time all of the low-end sensors are active. The H-sensors which are located in white cells become cluster heads and the clusters are constructed around these sensors. Whenever these sensors run out of energy, the H-sensors which are in green cells become active and then these sensors form a different number of clusters in the network again. Due to the formation of a different number of clusters, sensor nodes in white cells become non-critical sensors, and sensor nodes in green cells sensors become critical sensors. Since significant and critical sensor nodes consume large amount of energy than other non critical sensor nodes, this change from critical sensors to non critical sensors balances the energy utilization between sensors nodes, and considerably extends the network lifetime. The major issue concerned with CC is that sensor nodes which are closer to the cluster heads have a heavier burden and traffic load than nodes which are far away from the cluster heads, causing uneven energy consumption among sensor nodes.

Recently, a cluster-based approach is proposed to pre-serve the coverage [5]. The authors have analyzed different coverage-aware cost metrics, which can be utilized in the periodic election of cluster head nodes in homogeneous networks, ensuring that sensors that are more important to the network coverage task are less likely to be selected as cluster head nodes. Similarly, the cost metrics can also be used to find the set of active sensor nodes that provide full network coverage, as well as the set of routers that forward the cluster head nodes' data load to the sink. Although this hybrid approach can perform well for efficient and robust data delivery in dense uniform nodes deployment, it can't achieve these goals for sparse and non-uniform deployment scenarios. The work in this thesis is motivated by all the properties which are desirable considering the sensor network applications and are found lacking in existing protocols. Hence, various energy efficient clustering protocols are proposed to prolong the lifetime of the network. These solutions, mainly, deal with the uniform nodes deployment and assume that the network is alive only if all the nodes are alive otherwise the life is expired. Moreover clustering in a sparse deployment is still an open issue that must need to be addressed.

CHAPTER 3

Proposed Approach

3.1 Introduction to Proposed Approach

Two-Tier Cluster-Based Routing Protocol (TTCRP) is a new cluster based protocol which solves all those problems that were faced by researcher before this work. TTCRP uses power control algorithm (PCA) to achieve network robustness and efficiency and reliability in delivering the data from the source nodes to the base station or sink node. TTCRP uses resource rich cluster heads which minimizes the energy consumption. The details are given in the following sections.

3.2 Network Model

Consider a heterogeneous wireless sensor networks that comprises of n number of low-energy sensor nodes and m number of high-energy nodes, where $\mathbf{m \ll n}$. The required number of high-energy nodes can be computed using an optimal clustering algorithm as given in [16], [7]. The network is configured in the form of clusters in which only the high-energy nodes function as cluster-heads (CHs), while the low-energy nodes become the members of the clusters. We model the network as a connected directed graph $\mathbf{G = (V, E)}$. The notation **V** denotes the set of all the sensor nodes, i.e., $\mathbf{|V| = n + m}$. Each node $\mathbf{i \in V}$ has transmission range $\mathbf{r \in [r_{min}, r_{max}]}$, where $\mathbf{r_{min}}$ and $\mathbf{r_{max}}$ represents the minimum and maximum transmission range, respectively. **E** represents the set of edges between any two nodes in **V**. Due to variable transmission power, there may exist asymmetric links between two nodes. However, an edge $\mathbf{e(i, j) \in E}$ can exist if and only if $\mathbf{r(i) \geq d(I, j) \wedge r(j) \geq d(I, j)}$, where $\mathbf{(I, j) \in V}$ and $\mathbf{d(i, j)}$ is the distance between the two nodes **i** and **j** .

We make the following assumptions to facilitate the design of clustering protocol;

Assumption 1: Both the sensor nodes as well as the sink are stationary. High-energy nodes are uniformly deployed, while low-energy nodes can be deployed according to application requirements.

Assumption 2: High-energy nodes use dual channels with single interface, such as, IEEE 802.11, or they are equipped with two interfaces, such as, IEEE 802.11 and Zigbee. They use one channel (**channel-1**) for communication with the low energy nodes and the other **(channel-2)** for communication with the high-energy nodes.

Assumption 3: All the sensor nodes are capable of increasing and decreasing their transmission power level on **channel-1**.

3.3 Two-Tier Cluster-Based Routing Protocol

In this section, we describe the proposed Two Tier Cluster based Routing Protocol (**TTCRP**). **TTCRP** implements two algorithms; two-tier clustering algorithm and power control algorithm (**PCA**). In the clustering algorithm, nodes initially form clusters using the minimum transmission range $\mathbf{r_{min}}$. However, if the nodes are left to become members of any cluster due to non-uniform deployment, it triggers the PCA to get them join the clusters. Thus, it ensures network robustness to achieve the reliability.

3.3.1 Cluster Formation

The cluster formation is based on two-tier hierarchy. At the lower-tier, the whole network is organized in the form of clusters, where each high-energy node becomes a cluster-head and invites its neighboring low-energy nodes to join the cluster. Whereas, the high-energy cluster-heads also form clusters at the higher-tier for data routing to the sink. Lower-tier uses channel-1 for localized transmission within the cluster. While, the higher-tier uses channel-2 for transmission among the cluster-heads. Based on the assumption of uniform deployment of high-energy nodes, uniform clusters are possibly formed. However, the size of each cluster may vary greatly since there is no assumption about the deployment of low-energy nodes which may be sparse and non-uniform.

Each high-energy node **h** sets its transmission range **r** to $\mathbf{r_{min}}$ and broadcasts an advertisement message (**CH_ADV**) on channel-1 to invite the nodes in its cluster. The **CH_ADV** message contains the head ID (**head_ID**), its transmission power $\mathbf{P_{tx}}$ used for sending the message and the number of nodes in its cluster ($\mathbf{N_c}$). Initially, the value of $\mathbf{P_{tx}}$ is set to the power level to achieve the transmission range $\mathbf{r_{min}}$. Similarly, the cluster size is zero and therefore, the field is empty. However, the values may change as the clustering algorithm progresses. A sensor node may receive a number of **CH_ADV** messages from its nearby cluster-heads and measure the received signal strength (**RSS**) for each message. It chooses to join a cluster-head whose **RSS** value of the received message is higher than the others. The node also sets its transmission power to the value of $\mathbf{P_{tx}}$ field and sends a **CH_JOIN** message to the selected head. If a node receives the advertisement with same **RSS** value from more than one CHs then it choose the one that has the lesser value of $\mathbf{N_c}$**.** Each cluster-head maintains a members table that contains the **ID** of the nodes sending **CH_JOIN** message to it and the transmission power $\mathbf{P_{tx}}$ used for communication with the nodes. In the initial phase, the $\mathbf{P_{tx}}$ value for each node is identical. Hence, the lower-tier single hop clusters are formed in a simple way.

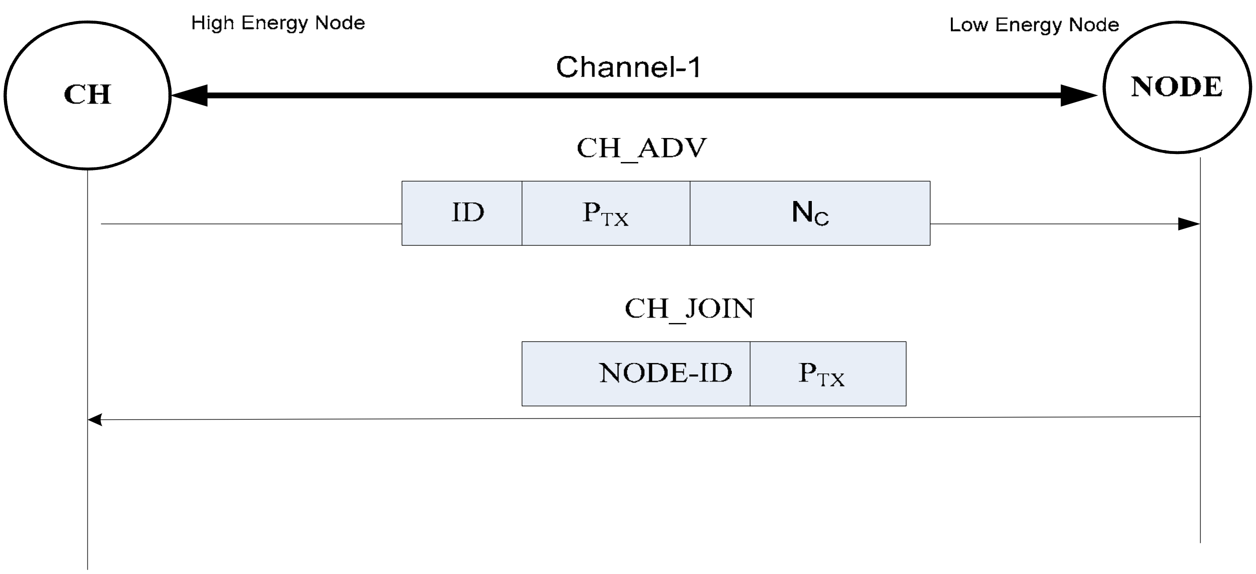

Figure 8: Cluster Formation at lower tier

Due to the limited transmission range and non-uniform deployment of low-energy nodes, there may exist nodes, which have not received **CH_ADV** message from any of the designated cluster-heads during the first round of clustering. In such scenario, TTCRP triggers the transmission power control algorithm described in Section IV-B to get these nodes join any of the cluster-head. It can be noted that such situation may also arise due to the failure of any cluster-head or bad propagation conditions in any part of the field. Hence, every node will join a cluster after certain rounds, where the convergence depends on the adaptation parameter in **PCA**. Moreover, cluster-heads broadcast **CH_ADV** message periodically in order to keep the members table updated. It will also give member nodes a choice to change their cluster-heads if they experience bad signal quality or its current cluster-head is serving larger number of member nodes.

Figure 9: demonstrates that the CH1, CH2 and CH3 nodes are uniformly placed between the sensor nodes 1-12 and nodes 1,2,3,4,5,6,8,9 and 12 join to their respective cluster heads to form clusters. Note that each higher energy cluster head node and every lower energy node uses **channel-1** to form the clusters. Some nodes such as node 7,10 and 11, which are out of transmission range of channel-1 could not become the member nodes of any cluster head and become the out of range nodes or disconnected nodes from the entire network.

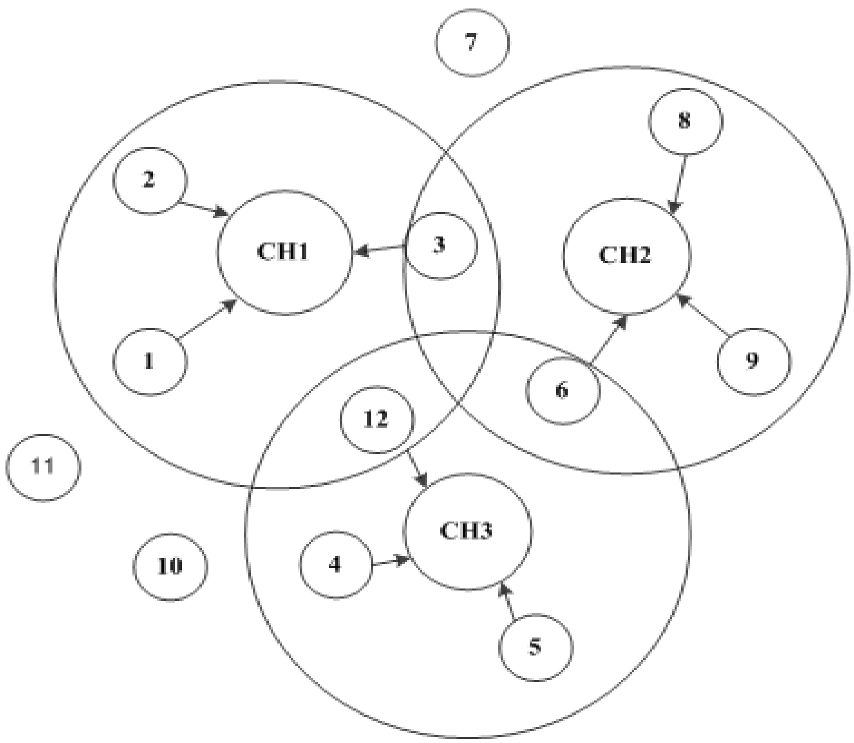

Figure 9: TTCRP Cluster Formation

3.3.2 Intra-Cluster Communication

For data delivery, cluster- heads are the nodes which are responsible of forwarding data from their member nodes to the sink. Thus, whenever, a node samples an event, it sends it to the cluster-head that may perform aggregation and sends the data to its higher-tier cluster-head as discussed in the following section. It is worthwhile to note that low-energy nodes neither maintain any neighboring list nor participate in data routing. The only message that low-energy nodes transmit for cluster formation is **CH_JOIN.** Thus, a great amount of energy can be saved that is a formidable requirement in WSNs.

3.3.3 Inter -Cluster Communication

At the higher tier, cluster-heads also form clusters of themselves by selecting super cluster-heads. This clustering is made to route data from clusters to the sink. Unlike the low-tier clusters, super cluster-head selection is based on the closeness of the cluster to the sink. Each cluster head broadcasts hello message on channel2 which is received only by the other cluster heads. The message contains location of the node and the number of low energy members (M) associated with it. After exchanging these messages, each cluster-head i computes the weigh Wj of its neighboring head j as follows

$$W_j = \left(1 - \frac{d(j,s)}{d(i,s)}\right) + \frac{M_i}{M_i + M_j} \qquad (1)$$

Where $d\ (i,\ s)$ is the distance computed between the head i and the sink s, and M_i represents the number of members associated with head i.

Thus, head i computes the weight for all of its neighboring heads and selects the super cluster-head that has the highest weight. That super cluster-head might also be a member of another super cluster-head closer to the sink. This enables to establish a multihop path from each cluster to the sink. It is important to note that data delivery will be made on $channel_2$ and thus it does not involve the low energy nodes in this process..

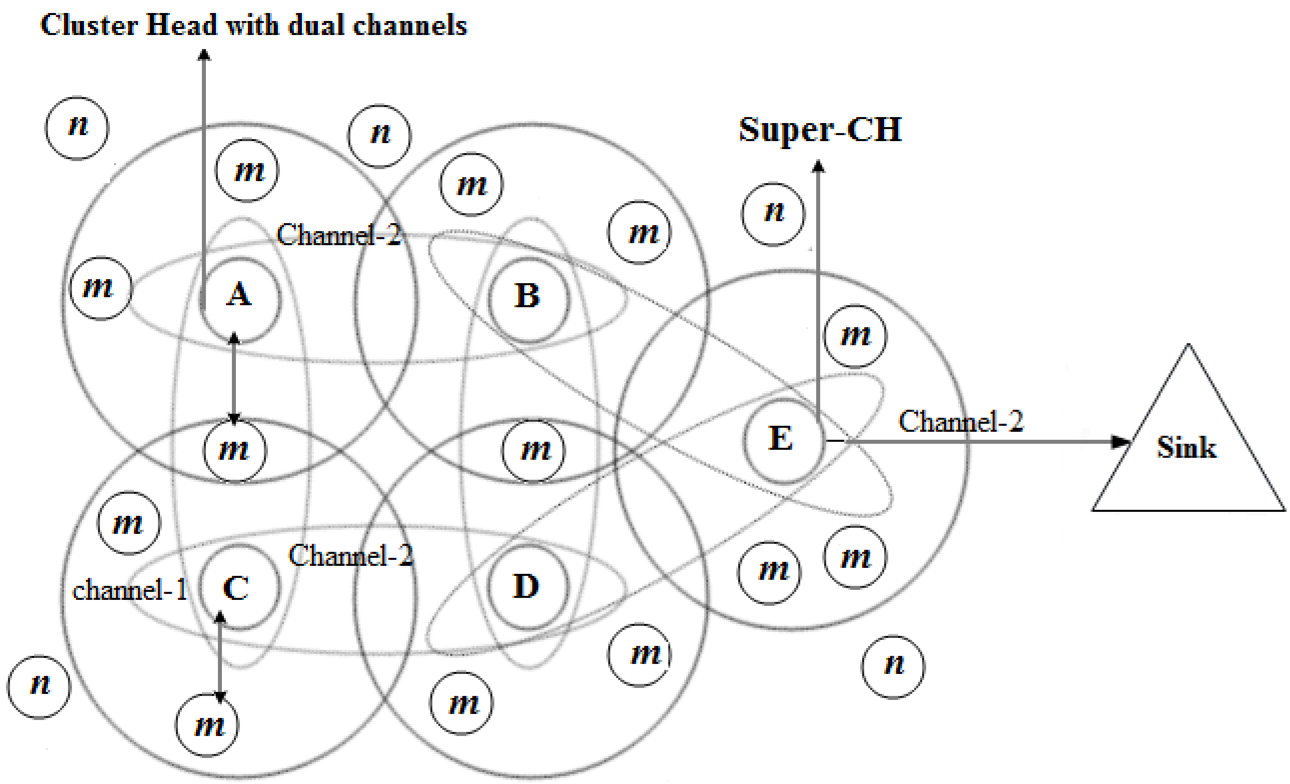

Figure 10: TTCRP Communications between Clusters

The example scenario is depicted in figure 10. Here m nodes represent the low energy member nodes of CHs and each member node communicate with the cluster head node using $channel_1$. *n* represents the non-member nodes which are out of range of cluster-heads or also called isolated nodes. It can be seen that CHs *A*, and *C* could not transmit data directly to the sink. *A* chooses *C* as its super CH and *B* chooses *D* after computing their weight. Whereas *C* and *D* choose CH *E* as their super cluster-heads. Hence clusters are organized hierarchically setting path from every CH to the sink without involving low energy nodes.

3.4 Power Control Algorithm (PCA)

The purpose of power control algorithm is to provide the network robustness and overcome the disconnectivity of the nodes which are out of the transmission range of CHs. As the isolated nodes could be introduced, TTCRP uses power control algorithm (PCA) in which both CHs and isolated nodes dynamically change their transmission range. When the isolated node i does not receive CH_ADV message from any CH, it gradually increases its transmission power (P_i) α unit in each step and pings the CHs. Where α is the increase in P_i in each attempt. Large value of α would rapidly converge to the power level required to reach the CH. However, this may increase the P_i unnecessarily higher than the required threshold and thus cost in more energy consumption. On the other hand small value of α will slowly converge to the required power level but would keep the P_i closer to the threshold. Hence, the tradeoff between delay and energy consumption gives an application choice to select appropriate value. After sending ping message with increased power level, node i waits for the response message CH_ADV from any nearby cluster-head. If it does not arrive, it increases its power level further by α unit and pings again. When the Cluster Head CH_i receives the ping message from node *i* after some attempts, it sets its power level equal to the node's power level P_i and sends the CH_ADV message. When node *i* receives the CH_ADV message, it sends the CH_JOIN message to that cluster head CH_i. In response the Cluster Head CH_i sends back the ACK.

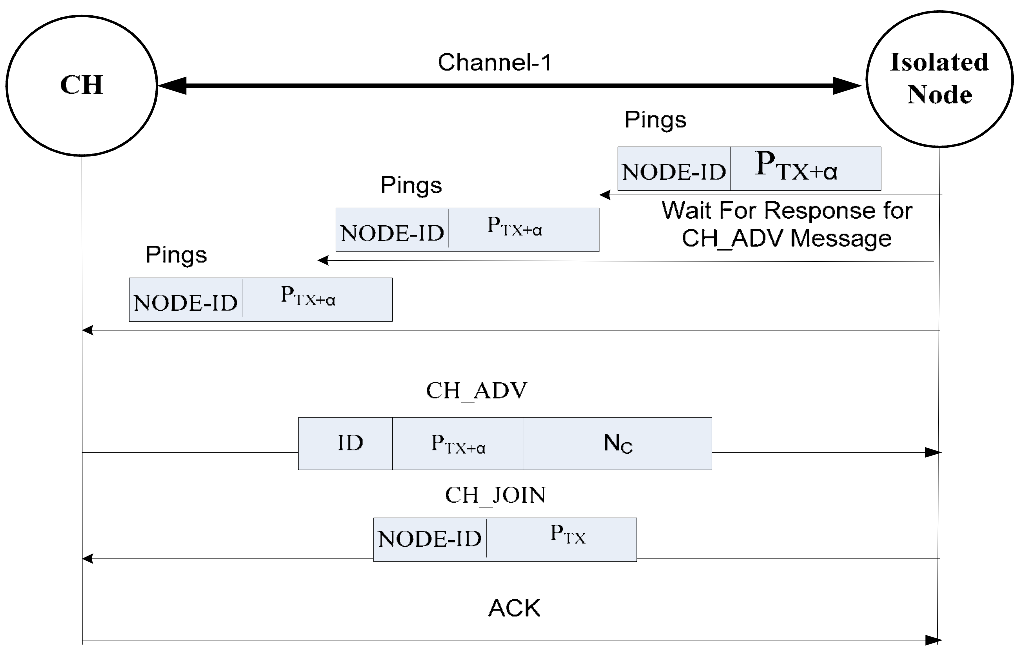

Figure 11: PCA Process in TTCRP

Apart from node *i*, a previously isolated node *j* , that has already joined some cluster-head CH_j using PCA algorithm, may also receive the CH_ADV message from CH_i. If *j* receives message with higher signal strength than its own cluster head, then it will also join cluster-head CH_i by sending CH_JOIN message to it and releases its previous membership with CH_j . Notably, CH_j will also reset its power level to the value used before associating node *j*. The pseudo-code of the operations of PCA is reported in Algorithm 1.

Algorithm 1 TTCRP Power Control

```
The pseudo code executed for transmission power control
i denote Lower energy nodes and CH denotes resource rich cluster heads
p denotes minimum transmission range r_min of node i
α is the increase in sensor node transmission range
initialization:
if r(i) < d(i, CH ) then
Setpower (i, p)
 if CH_ADV (i).count = 0 then
  repeat
    Setpower(i, p + α)
    CHping (i)
  until i receiveCH_ADV(CH)
  if CH receiveCHping(i) then
    Setpower (CH, p + α)
    SendCH_ADV (CH)
  end if
  while CH_ADV (i).count > 1 do
    i CompareRSS ( )
    if CH HigherRSS ( ) then
      sendCH_JOIN (i)
      Setpower (i, p_new)
    else
      if same RSS then
        i CompareCH_N_C( )
        if CH Lowest_N_C( ) then
          Setpower(i, p_new)
          sendCH_JOIN(i)
        end if
      end if
    end if
  end while
 else
  CH receiveCH_JOIN (i)
  CH sendACK( )
 end if
end if
```

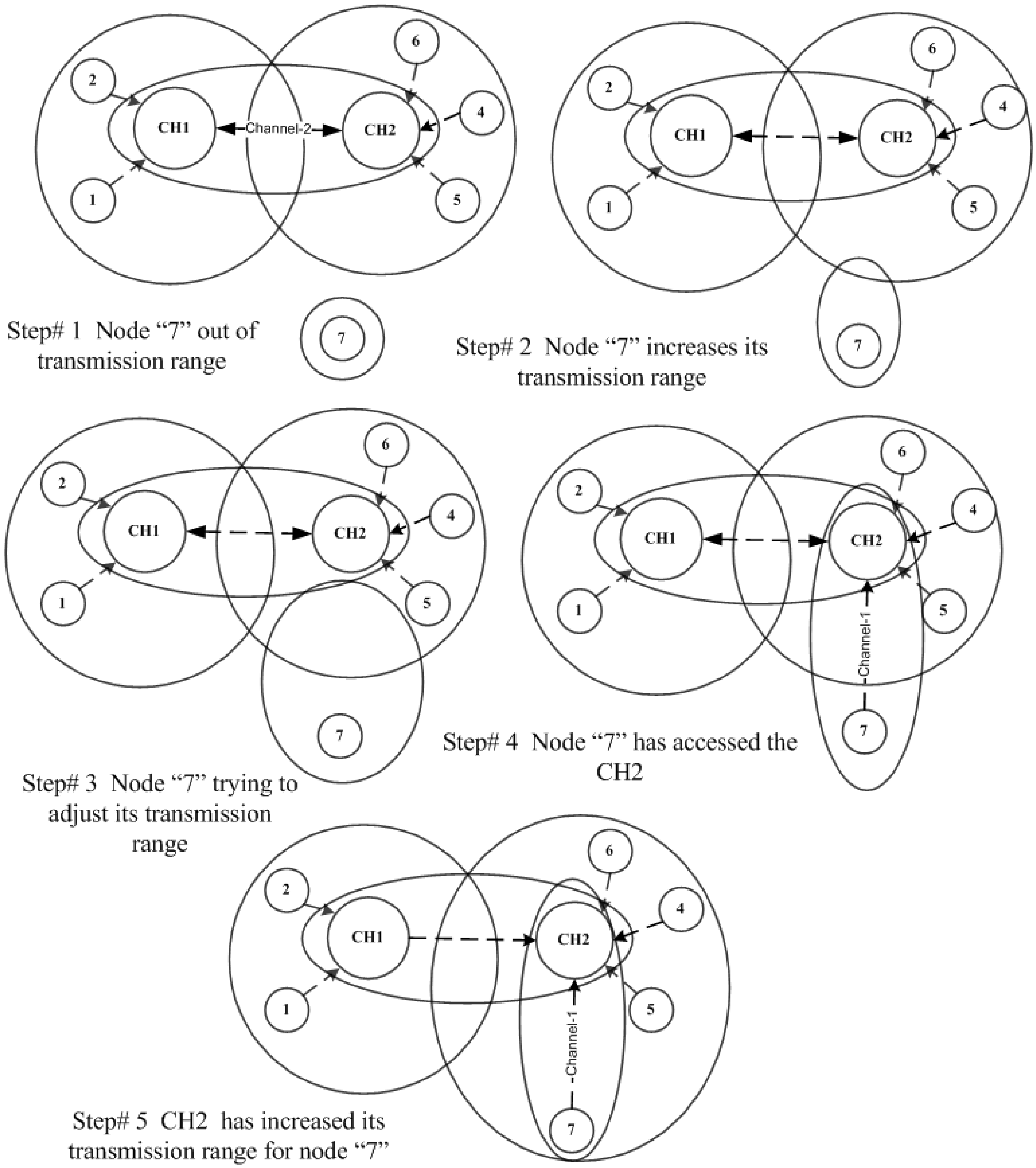

Figure 12: Power Control Algorithm in TTCRP

Figure 12: demonstrate the example scenario in which PCA applied to associate isolated nodes with cluster heads. Node *7* is out of the transmission range. Node *7* has not yet received CH_ADV message from any cluster head and is an isolated node. It increases its transmission range P_{tx} per α to access the cluster head and pings the CH_2. After certain attempts made by node 7, cluster head CH_2 receives the ping message and sets its transmission power equal to the value of P_{tx} field present in the ping message of node 7. CH_2 then sends the CH_ADV message to node *7*. On successful reception, node *7* sends the CH_JOIN message to CH_2. In this way the isolated lower energy node *7* joins the network and is able to send event data.

3.5 Data Dissemination

The date disseminates using TTCRP protocol in wireless sensor network as the source nodes sends their sensed information to the CH using Channel-1. CHs in tern sends it to their neighbor CH node using channel-2 and which in tern send it to the super cluster head. Super cluster head forwards the data to the sink node via channel-2. CH node receives data from member nodes on one channel and sends it to CH on another channel, which decreases delay.

CHAPTER 4

Experimental Results

4.1 Simulation Parameters and Results

In this section, we evaluate the performance of TTCRP by comparing the results with the existing protocols CC [23], LRS [31] and GROUP [24]. The protocol is implemented in ns2 [22] which is a scalable discrete event simulator. Simulation scenario consists of a sink node, varying number of high energy nodes with variable number of low energy nodes in the range from 100 to 300. The sink is placed at the center of the field. Table I lists the simulation parameters used in the setup.

Parameter	**Value**
Simulator	NS-2
MAC layer	IEEE802.11
Number of total nodes	1200
Number of Source nodes	10-100
Number of Cluster heads(CH)	1-40
Node placement	Random
Field Area(X-Y Plane)	1000-2000 m
Initially Transmission range (pt)	0.2
Application traffic	TTCRP
Simulation time	100-500 sec
Propagation model	Two Ray Ground
Packet size	512
Channel for node	Single Wireless Channel
Channel for Cluster Head	Two Wireless Channels

Table 1 Simulation Parameters

In the first scenario, we measure the energy consumption of TTCRP by varying cluster-heads from 5 to 15 with 100 sensing nodes. Results are plotted in figure 13 hat shows the energy consumption is lesser as we increase the cluster heads. It is due to the fact that the isolated nodes are lesser with higher cluster heads that would not cause them to increase the range and thus saving the energy of nodes. By increasing the number of cluster heads from 5 to 15, the energy consumption is reduced 90%. Similarly the throughput achieved with introducing different number of isolated nodes in non-uniform deployment is plotted in figure 14. It can be seen that the throughput decreases slightly by increasing the disconnected nodes from 5 to 10.

This is achieved by using PCA algorithm because it associates the farther nodes with the cluster heads soon after the clusters are formed.

The scalability of TTCRP is also evaluated by varying the number of nodes from 100 to 300 with 50 sources. Figure 15 shows the average throughput with different number of cluster heads. We achieve 96% throughput when the cluster heads are 15 and the value drops to 86% by increasing the number of sensor nodes 3 times. Thus the reduction is only 10%. Similarly the throughput is 92% when the cluster heads are 5. Thus, we obtain even higher throughput at lower cluster heads by use of PCA algorithm.

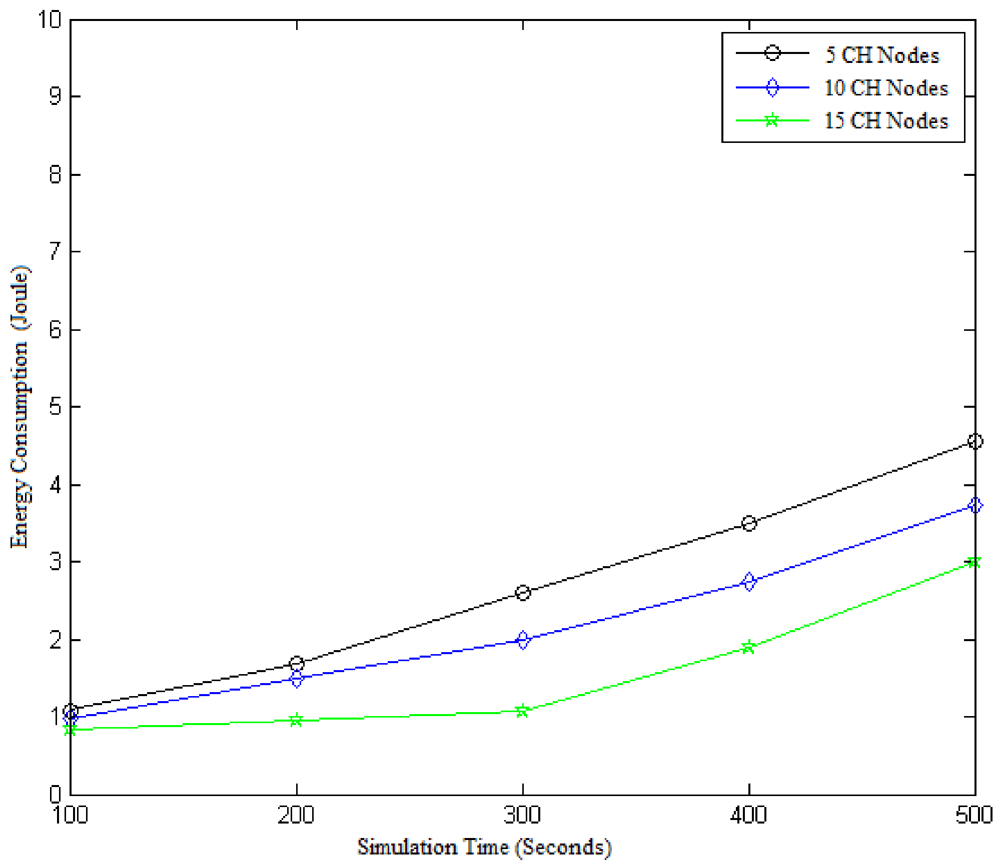

Figure 13: Energy consumption using PCA in TTCRP

It can also be seen in figure 16 that TTCRP exhibits lower average delay than the GROUP protocol. As GROUP is a multi-hop clustering routing protocol, while TTCRP is a single-hop routing protocol at lower tier and multi-hop at higher tier. Moreover, the transmission at higher tier is done on different channel or interface that does not interfere with the transmission of member nodes and thus incurs lesser delay. Moreover, TTCRP demonstrates significantly lower average delay with large number of sensor nodes as compared to GROUP.

Finally, we provide comparison of total energy consumption with the existing protocols CC and LRS in a network of 1200 nodes. Experiments are run for 500 seconds simulation time. The results are shown in Figure 17. As we can see the total energy consumption in LRS and CC is much larger than total energy consumption in TTCRP. This is due to the fact that member nodes also perform routing in LRS that result in higher energy consumption even though it balances the energy consumption.

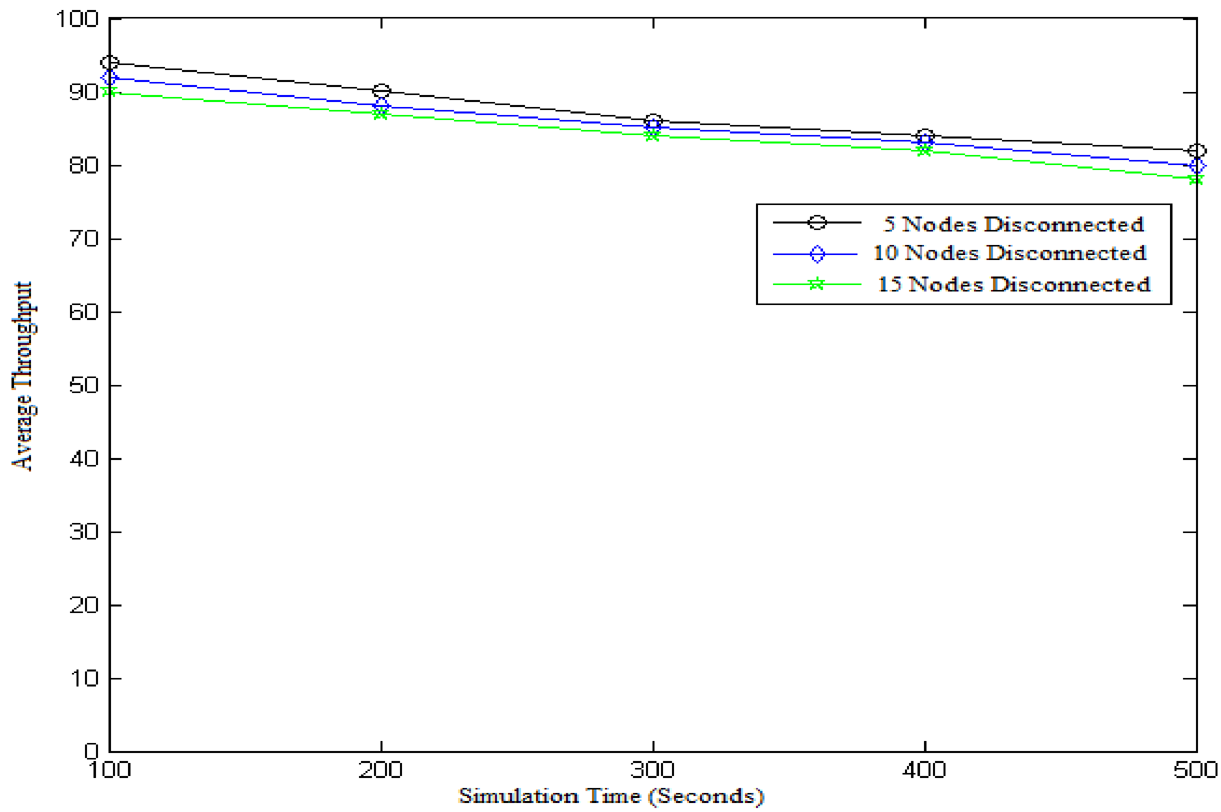

Figure 14: Average Throughput vs. Simulation time

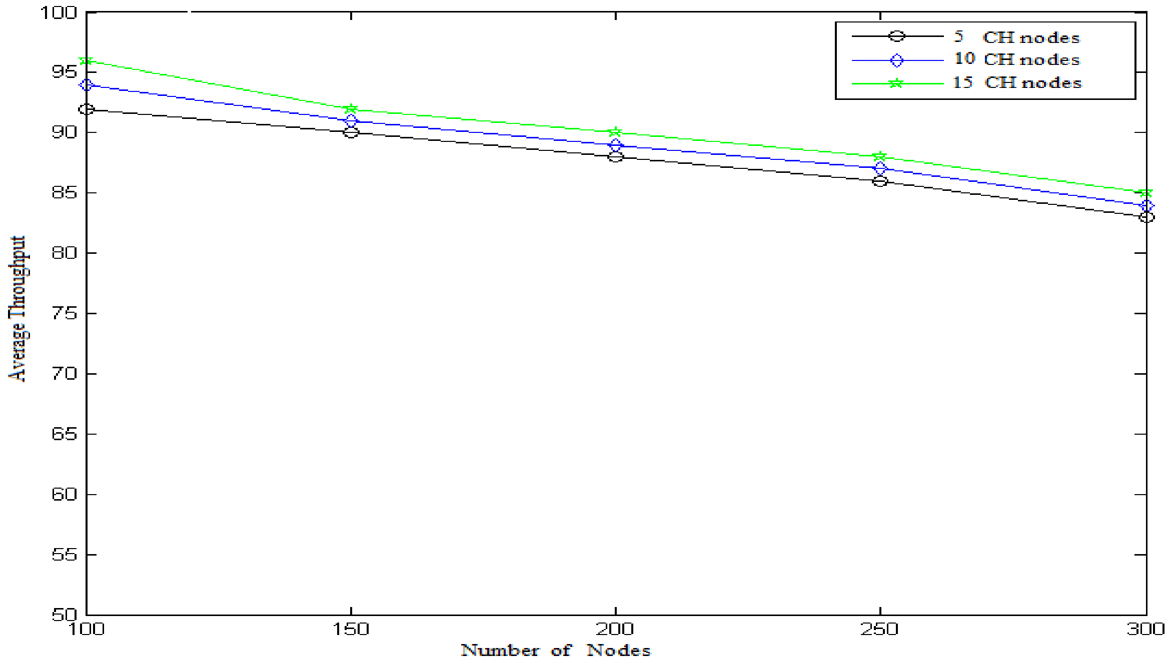

Figure 15: Average Throughput vs. Number of nodes

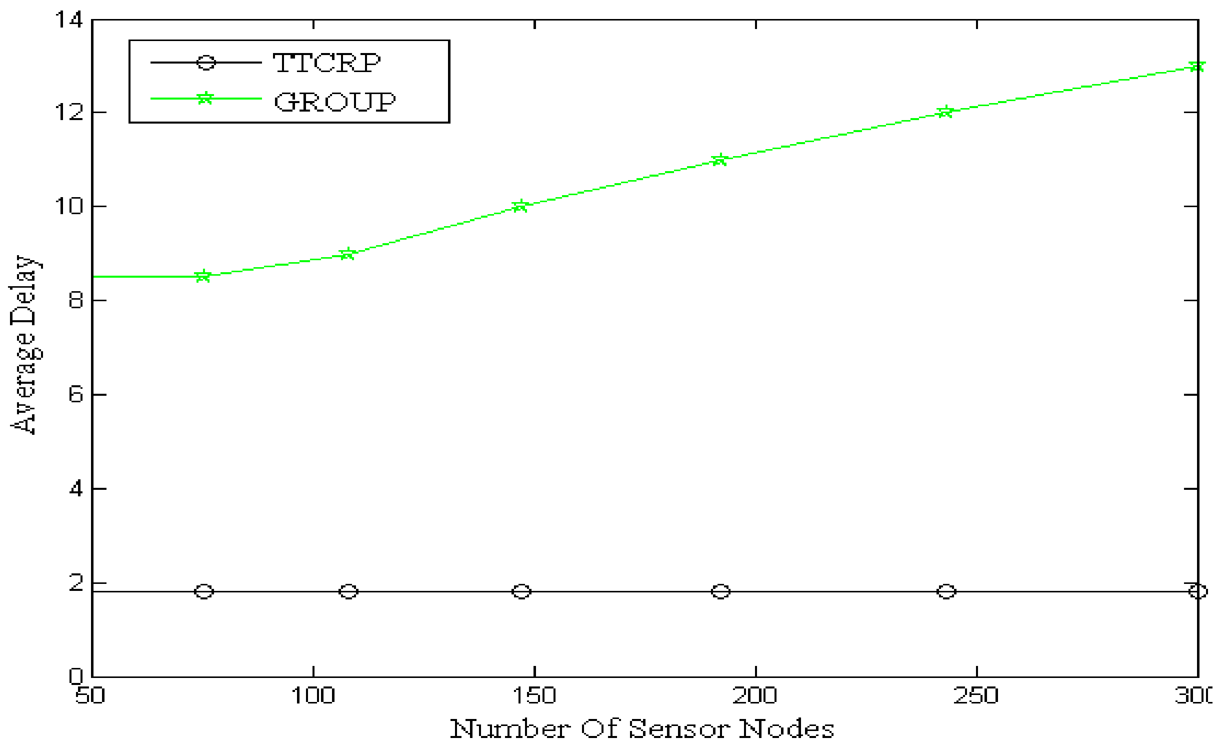

Figure 16: Average Delay vs. Number of Nodes

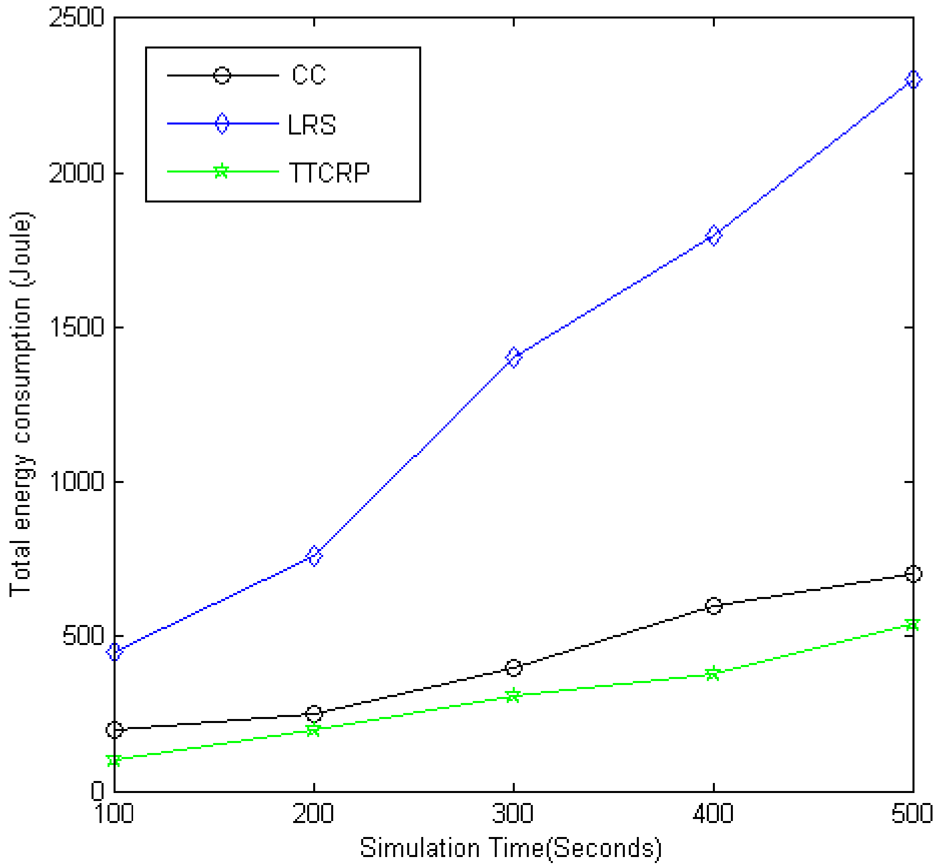

Figure 17: Total Energy consumption vs. Simulation time

In figure 18 and 19 shows the simulation of NS-2 in NAM. In which the red color node (17) is a sink node and the blue color node are high-energy nodes (cluster heads) which takes data from source nodes or its member nodes in cluster and forwards it to the sink node. Such as in this case node 0, 1, 2, 3, 4 and 5 are cluster heads and all other nodes in green color are sensor nodes in the network. Black node (40) is a phenom node or event node which generates events. In Appendix A, TTCRP TCL script scenario is given which generates the NAM as shown in Figure 18 and 19.

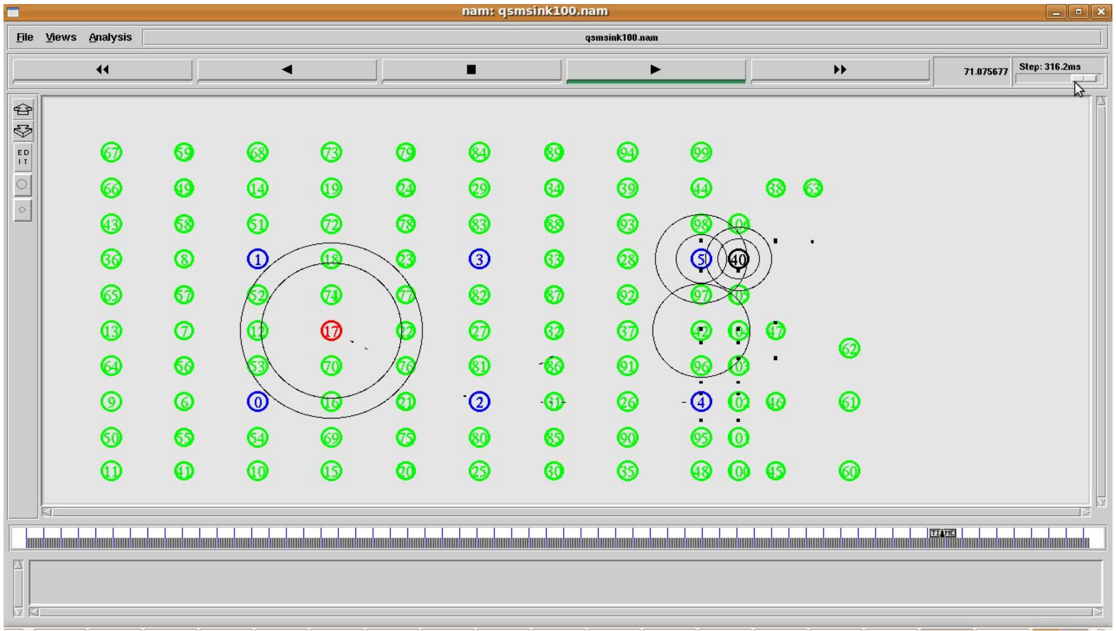

Figure 18: TTCRP Protocol: Simulation in NS-2 NAM

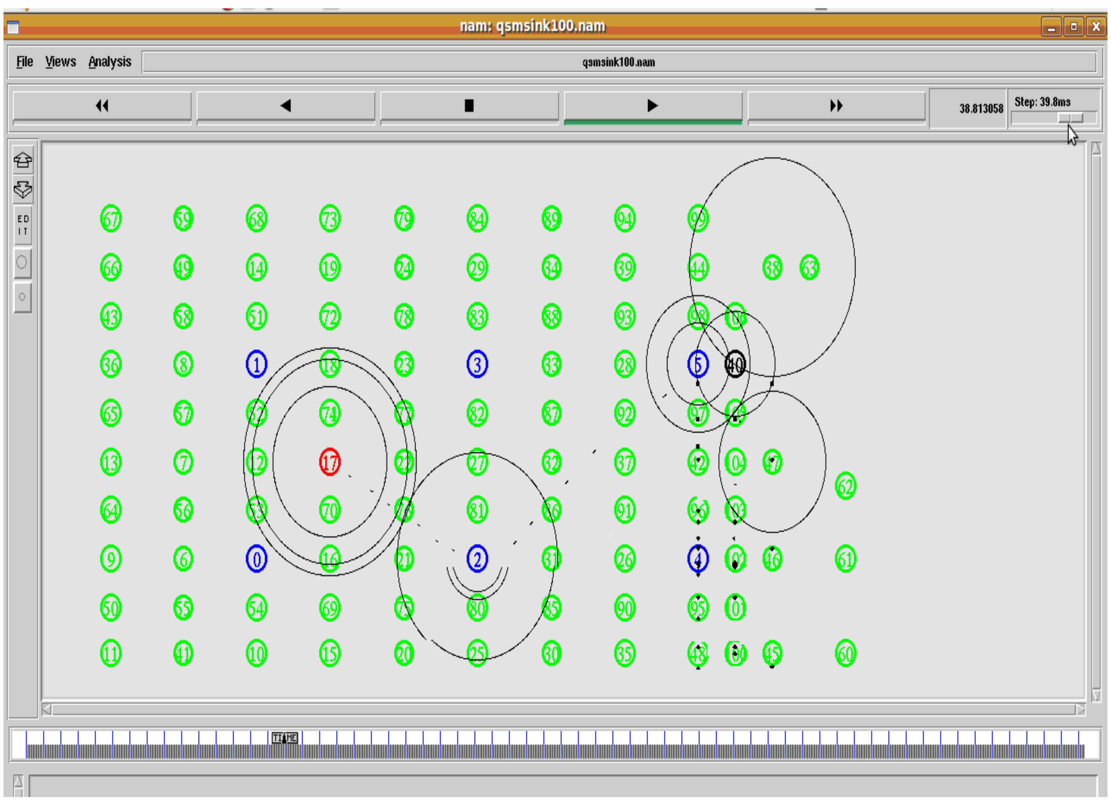

Figure 19: TTCRP Protocol: Simulation in NS-2 NAM

4.2 Conclusion & Future Work

In this paper, a novel clustering protocol is proposed for the large scale wireless sensor networks. The aim of designing such an algorithm is to run in a disaster management and surveillance system in which the deployment is highly dense and non-uniform. Thus the protocol must needs to be very efficient and reliable. The goal is achieved by designing the protocol in two tiers and limiting the routing function only on the classified high energy nodes. Simulation results have revealed that TTCRP is an efficient and reliable protocol which has very small end to end delay and achieve high throughput because of using the designated cluster heads with dual channels. As well as the use of dual channels, the CHs have significantly overcome the delay and consume less energy as compared to LRS and CC.

We conclude that TTCRP is well suited for disaster aid network which is time critical application and is also quite efficient and reliable in terms of energy consumption and response time.

During this study, our focus was only on the designing of the two tier cluster based routing protocol. TTCRP is based on the assumption that low energy nodes as well as high energy nodes are static. The main limitation of our protocol is that it uses static sensor nodes. Therefore, the future direction might be designing, implementing and extend the TTCRP protocol for mobile sensor nodes. However, there are many other issues of WSNs that also needs to be explored. At design level of WSN, various issues which need to be investigated further are node deployment, heterogeneity and localization. As well as different challenges need to be implemented on different protocols including TTCRP in real scenarios to identify protocols efficiencies and remove deficiencies. The study of the packet loss percentage may also lead to some important conclusions such as effect of interferences and noise brought about by range adjustment which is caused by power control algorithm. Also extending the proposed protocol for the real life event driven networks might be productive. The protocol is expected to achieve better performance for event driven networks with some minor modifications.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," Computer Networks, vol. 38, Pp.393-422, 2002 .

[2] A. A. Abbasi, and M. Younis, "A survey on clustering algorithms for wireless sensor networks," Computer Communication vol. 30, no. 14-15, pp. 2826- 2841, Oct. 2007.

[3] F. Bajaber, and I. Awan, "Adaptive decentralized re-clustering protocol for wireless . sensor networks," Journal of Computer and System Sciences, 2010,

[4] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., Cayirci, E., Wireless sensor networks: a survey, Computer Networks, Elsevier: 2002; 38(4) pp. 393-422.

[5] S. Soro, W. B. Heinzelman,"Cluster head election techniques for coverage preservation in wireless sensor networks," Ad Hoc Networks, vol. 7 no. 5, pp. 955-972, Jul. 2009.

[6] C. Song, M. Liu, J. Cao, Y. Zheng, H. Gong, G. Chen, "Maximizing network lifetime based on transmission range adjustment in wireless sensor networks," Computer Communications, vol. 32, no. 11, pp. 1316-1325, Jul. 2009.

[7] G. A. Shah, O. B. Akan and F. B. Hussain "Cluster-based coordination and routing framework for wireless sensor and actor networks", Wiley Journal of Wireless Communication and Mobile Computing, 2009.

[8] S. Lee, J. Lee, H. Sin, et. al. "An Energy-Efficient Distributed Unequal Clustering Protocol for Wireless Sensor Networks," In Proc. of World Academy of Science, Engineering and Technology, pp. 443-447, 2008.

[9] S. Elvakennedy, S. Sinnappan, and Y. Shang, "A biologically-inspired clustering protocol for wireless sensor networks," Computer Communications vol. 30, no. 14-15, pp. 2786-2801, Oct. 2007.

[10] G. Chen, C. Li, M. Ye, J. Wu, "An unequal cluster-based routing protocol in wireless sensor networks", Springer, Wireless Networks Journal, vol. 15, no. 2, pp. 193-207, 2007.

[11] M. Yu, K. K. Leung and A. Malvankar, "A dynamic clustering and energy efficient routing technique for sensor networks," IEEE transactions on wireless communications, vol. 6 no. 8, pp. 3069-3079, 2007.

[12] J.S. Liu, C. R. Lin, "Energy-efficiency clustering protocol in wireless sensor networks," Ad Hoc Networks, vol. 3, no. 3, pp. 371-388, May 2005.

[13] O. Younis and S. Fahmy, "Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach," in Proc. of IEEE INFOCOM, vol. 1, pp. 629-640, Mar. 2004.

[14] V. Mhatre, C. Rosenberg, "Design Guideline for Wireless Sensor Networks: Communication, Clustering and Aggregation," Ad-Hoc Networks Journal ,Elseviser Science, vol. 2, no. 1, pp45-63, Jan 2004.

[15] J. Ibriq and I. Mahgoub, "Cluster-Based Routing in Wireless Sensor Networks: Issues and Challenges", In Proc. of the Symposium on Performance Evaluation of Computer Telecommunication Systems, pp.759-766, Jul. 2004.

[16] W.B. Heinzelman, A.P. Chandrakasan, H. Balakrishnan, "Application specific protocol architecture for wireless microsensor networks," IEEE Transactions on Wireless Networking, 2002.

[17] A. Manjeshwar and D. P. Agarwal, "APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks", In Proc. of International Symposium on Parallel and Distributed Processing (IPDPS'02), pp. 195-202, 2002.

[18] S. Bandyopadhyay and E. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," In Proc of the 22nd INFOCOM 2003.

[19] R.S. Chang and C.J. Kuo, "An Energy Efficient Routing Mechanism for Wireless Sensor Network", In Proc. of 20th International Conference on Advanced Information Networking and Applications, vol. 2, pp. 18-22, 2006.

[20] B. Son, Y. Her, and J. Kim "A Design and Implementation of Forest Fires Surveillance System based on Wireless Sensor Networks for South Korea Mountains", School of Computer and Communication Engineering, Daegu University, Daegu, pp. 712-714 Korea ,2006.

[21] G. W. Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh "Monitoring Volcanic Eruptions with a Wireless Sensor Network", 2005.

[22] UCB/LBNL/VINT Network Simulator, Available: http://www.isi.edu/nsnam/ns/.

[23] Du, X. and Xiao, Y. (2006) Energy efficient Chessboard Clustering routing in heterogeneous sensor networks, Int. J. Wireless and Mobile Computing, Vol. 1, No. 2, pp.121130.

[24] Liyang Yu, Neng Wang and Wei Zhang " A Grid-clustering Routing Protocol for Wireless Sensor Networks" 2006 IEEE.

[25] F.L. Lewis, "wireless sensor network," *Technologies Protocols and Applications*, New York, 2004

[26] M. Lyas and I. Magoub. *Compact wireless and wired sensor system.* CRC Press, 2004.

[27] S. Hedetniemi and A. Liestman. A survey of gossiping and broadcasting in Communication networks. Networks, 18(4):319–349, 1988.

[28] Zygmunt Haas, Joseph Y. Halpern, and Li L. Gossip-based ad hoc [29] Routing. *IEEE/ACM Trans. Netw.*, 14(3):479–491, 2006.

[29] W. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks, 1999.

[30] Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In MobiCom '00:Proceedings of the 6th annual international conference on Mobile computing and networking, pages 56–67, New York, NY, USA, 2000. ACM Press.

[31] Lindsey, S., Raghavendra, C.S. and Sivalingam, K. (2001) Data gathering in sensor networks using the energy delay metric, Proc. 15th International,Parallel and Distributed Processing Symposium, San Francisco,CA, April, pp.2001-2008.

[32] l Stojmenovic. *The state of the art of sensor network*. John wali and sensor.2005

[33] A.Khetrapal, "Routing techniques for Mobile Ad Hoc Networks Classification and Qualitative/ Quantitative Analysis," Department of Computer Engineering, Delhi College of Engineering University.

[34] L.Cui, F. wang and H. Luo. "Network and Parallel Computing," Springer Berlin / Heidelberg. Ltd.14 Oct 2004.

[35] G. Acs and L. Buttyabv. "A taxonomy of routing protocols for wireless sensor networks," *BUTE Telecommunication department,* Jan. 2007.

[36] T. He, et.al, "Achieving Real-Time Target Tracking Using Wireless Sensor Networks,"in *Proceedings of the 12th IEEE* Vol.4, Issue 7, pp.37-48, April. 2006.

[37] J. Fraden. *A hand book of modern sensor: Physic, design, and application*. Birkauser, 2004.

[38] I.Akyildiz, W. Su, Y. Sankarasubramaniam,"A survey on sensor networks," IEEE Communications Vol: 40 Issue: 8, pp.102-114, August 2002.

[39] G. Gelet, "Performance Evaluation of Wireless Sensor Network Routing Protocols for Critical Condition Monitoring Application" M.A. thesis. Addis Ababa University, Oct 2007.

[40] Wenning, B.L. Pesch, D.Timm-Giel, A. Görg. "Environmental monitoring aware routing in wireless sensor networks*," in Proceedings of the IFIP joint conference on Mobile and Wireless Communications Networks (MWCN 2008)* and Personal Wireless Communications, 2008, pp. 5-16.

[41] A.al-yasiri and A.sunley. “Data aggregation in wireless sensor networks using the SOAP protocol,” Journal of Physics Conference Series 76**,** 2007.

[42] K. Mitta, A. Veda, B.K. Meena, “Data Aggregation, Query Processing and Routing in Sensor Networks,” MTech IT, Powai, Mumbai.

[43] Y. Wang, G. Attebury and B. Ramamurthy. “A survey of security issues in wireless sensor networks,” *IEEE communication surveys*, Vol.8, No.2, 2006.

[44] Jamal N.Al-Karaki, A.E. Kamal, “Routing techniques in wireless sensor networks a survey,” *Wireless Communications*, IEEE Publication Vol.11, Issue. 6, pp.6-28, Dec. 2004.

[45] M. Frikha, J.B. Slimane, “Conception and Simulation of Energy-Efficient AODV protocol Ad Hoc Networks,” Tunisian Communication’s, Tunis.

[46] S. Sharma, D. Kumar and R. Kumar, “QOS-Based Routing Protocol in WSN,” Advances in Wireless and Mobile Communications.ISSN 0973-6972 Vol. 1, No. 1-3, pp.51-57, 2008.

[47] X. Hong, K. Xu and M. Gerla."Scalable Routing Protocols for Mobile Ad Hoc Networks," IEEE Network, University of California at Los Angeles, Aug. 2002.

[48] Stephanie Lindsey and Cauligi S. Raghavendra. Pegasis: Power-efficient gathering in sensor information systems, 2002.

[49] Arati Manjeshwar and Dharma P. Agrawal. Teen: A routing protocol for enhanced efficiency in wireless sensor networks. 15th International Parallel and Distributed Processing Symposium (IPDPS’01), 03, 2001.

[50] Arati Manjeshwar and Dharma P. Agrawal. Apteen: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In IPDPS ’02: Proceedings of the 16th International Parallel and Distributed Processing Symposium, page 48,2002.

[51] Sajid Hussain, Abdul W. Martin, Hierarchical Cluster based Routing in Wireless Sensor Networks, Journal of networks,Academy Publisher, Vol 2, Number 5, Pages 87-97, 2007.

[52] Malik Tubaishat, Jain Yin, Biswajit Panja and Sanjay Madria, A secure hierarchical model for sensor network, ACM SIGMOD Record, 33, 7-13 2004

[53] www.wikipedia.org

Appendix A

TTCRP TCL SCRIPT SCENARIO

This TCL Script describes the simulation of TTCRP protocol according to the specific scenario. In this tcl script different parameters are given which is used in the simulation scenario such as set val(chan) for Channel/WirelessChannel and for TTCTP routing Protocol we set val(adhocRouting) to ttcrp .Each Line is clearly explained by comments. The following Tcl Script produces the NAM and Trace files. Figure 20 clearly demonstrates the NAM of TTCRP Protocol.

```
#=======================================================================
# Define options
#=======================================================================


set val(prop)           Propagation/TwoRayGround
set val(netif)          Phy/WirelessPhy
set val(mac)            Mac/802_11
set val(ifq)            Queue/DropTail/PriQueue
set val(ll)             LL
set val(ant)            Antenna/OmniAntenna
set val(x)              2500        ;# X dimension of the topography
set val(y)              1000        ;# Y dimension of the topography
set val(ifqlen)         54          ;# max packet in ifq
set val(seed)           0.0
set val(adhocRouting)   ttcrp
set val(nn)             6           ;# how many nodes are simulated
set val(na)             114         ;# number of Actor nodes
set val(rxPower)        0.00175     ;# receptión energy W
set val(txPower)        0.00175     ;# transmissión energy W
set val(energymodel)    EnergyModel ;
set val(initialenergy)  1000        ;# Initial energy in Joules
set val(txPower)        0.0016      ;# transmitting power in mW
set val(rxPower)        0.00004     ;# recving power in mW
set val(sensePower)     0.00000175  ;# sensing power in mw
set val(idlePower)      0.0         ;# sensor nodes idle power
#======================================================================

set val(stop)           500.0       ;# simulation time

Phy/WirelessPhy set CPThresh_ 10.0
Phy/WirelessPhy set CSThresh_ 1.559e-11
Phy/WirelessPhy set RXThresh_ 1.58e-10 ;# Sensibilidad: -98 dBm.
Phy/WirelessPhy set Rb_ 2*1e6
Phy/WirelessPhy set Pt_ 0.20
Phy/WirelessPhy set freq_ 916e+6
Phy/WirelessPhy set L_ 1.0
Phy/WirelessPhy set Pt_consume_ 0.002 ;# power consumption for
transmission (W)
```

```
Phy/WirelessPhy set Pr_consume_ 0.002 ;# power consumption for
reception (W)
Phy/WirelessPhy set P_idle_  1.0      ;#idle power consumption (W)
Phy/WirelessPhy set P_sleep_ 0.00005  ;# sleep power consumption (W)
#======================================================================
# Main Program
#======================================================================
# create simulator instance

set ns_           [new Simulator]

# setup topography object

set topo    [new Topography]

#=====================================================================
set tracefd [open trace/TTCRP120.tr]   ;#change 10
set namtrace    [open TTCRP120.nam w]

$ns_ trace-all $tracefd
$ns_ namtrace-all-wireless $namtrace $val(x) $val(y)

# define topology
$topo load_flatgrid $val(x) $val(y)

# Create God
set god_ [create-god [expr $val(nn) + $val(na)]]

set chan_0_ [new $val(chan)]
set chan_1_ [new $val(chan)]
set chan_2_ [new $val(chan)]

#global node setting

$ns_ node-config -adhocRouting $val(adhocRouting) \
                 -llType $val(ll) \
                 -macType $val(mac) \
                 -ifqType $val(ifq) \
                 -ifqLen $val(ifqlen) \
                 -antType $val(ant) \
                 -propType $val(prop) \
                 -phyType $val(netif) \
                 -topoInstance $topo \
                 -agentTrace ON \
                 -routerTrace ON \
                 -macTrace ON \
                 -energyModel $val(energymodel) \
                 -idlePower 0.0 \
                 -rxPower 0.00004 \
                 -txPower 0.0016 \
                 -initialEnergy 1 \
                 -transitionTime 0.005

 $ns_ node-config -channel $chan_1_\
       -channel2 $chan_2_\
```

```
#Create the specified number of nodes [$val(nn)] and "attach" them to
#the channel.
#Create Normal Nodes
for {set i 0} {$i < $val(nn) } {incr i} {
            set node_($i) [$ns_ node]
            $node_($i) random-motion 0 ;# disable random motion
            $ns_ initial_node_pos $node_($i) 50
            $god_ new_node $node_($i)
}
Phy/WirelessPhy set CPThresh_ 10.0
Phy/WirelessPhy set CSThresh_ 1.559e-11
Phy/WirelessPhy set RXThresh_ 1.58e-10 ;#Sensibilidad: -98 dBm.
Phy/WirelessPhy set Rb_ 2*1e6
Phy/WirelessPhy set Pt_ 0.20
Phy/WirelessPhy set freq_ 916e+6
Phy/WirelessPhy set L_ 1.0
Phy/WirelessPhy set Pt_consume_ 0.002 ;# // power consumption for
transmission (W)
Phy/WirelessPhy set Pr_consume_ 0.002 ;# // power consumption for
reception (W
Phy/WirelessPhy set P_idle_  1.0;      # // idle power consumption (W)
Phy/WirelessPhy set P_sleep_ 0.00005; #// sleep power consumption (W)
$ns_ node-config -channel $chan_1_\
       -channel2 $chan_2_\


      for {set i $val(nn)} {$i < [expr $val(nn)+$val(na)] } {incr i} {
            set node_($i) [$ns_ node]
            $node_($i) random-motion 0 ;# disable random motion
            $ns_ initial_node_pos $node_($i) 50

      }
[$node_(40) set ragent_] event_rate 0.1
[$node_(40) set ragent_] event_type 1 ;#EV_LIGHT
[$node_(40) set ragent_] phen_ 1 ;#EV_LIGHT

[$node_(58) set ragent_] event_rate 0.1
[$node_(58) set ragent_] event_type 1 ;#EV_LIGHT
[$node_(58) set ragent_] phen_ 1 ;#EV_LIGHT

[$node_(40) set ragent_] event_rate 0.1
[$node_(40) set ragent_] event_type 1 ;#EV_LIGHT
[$node_(40) set ragent_] phen_ 1 ;#EV_LIGHT

[$node_(99) set ragent_] event_rate 0.1
[$node_(99) set ragent_] event_type 1 ;#EV_LIGHT
[$node_(99) set ragent_] phen_ 1 ;#EV_LIGHT

[$node_(68) set ragent_] event_rate 0.1
[$node_(68) set ragent_] event_type 1 ;#EV_LIGHT
[$node_(68) set ragent_] phen_ 1 ;#EV_LIGHT

[$node_(11) set ragent_] event_rate 0.1
[$node_(11) set ragent_] event_type 1 ;#EV_LIGHT
[$node_(11) set ragent_] phen_ 1 ;#EV_LIGHT

[$node_(60) set ragent_] event_rate 0.1
[$node_(60) set ragent_] event_type 1 ;#EV_LIGHT
[$node_(60) set ragent_] phen_ 1 ;#EV_LIGHT
```

```
[$node_(17) set ragent_] actor_ 1 ;#EV_LIGHT
[$node_(0) set ragent_] cl_head 1
[$node_(1) set ragent_] cl_head 1
[$node_(2) set ragent_] cl_head 1
[$node_(3) set ragent_] cl_head 1
[$node_(4) set ragent_] cl_head 1
[$node_(5) set ragent_] cl_head 1
#======================================================================
puts "Loading connection pattern..."
# the number of nodes are defined with X and Y coordinates
$node_(11) set X_ 5.0
$node_(11) set Y_ 5.0
$node_(11) set Z_ 0.0
$ns_ at .0 "$node_(11) setdest 5 5 0.0"


$node_(9) set X_ 5.0
$node_(9) set Y_ 200.0
$node_(9) set Z_ 0.0
$ns_ at .0 "$node_(9) setdest 5 200 0.0"

$node_(13) set X_ 5.0
$node_(13) set Y_ 400.0
$node_(13) set Z_ 0.0
$ns_ at .0 "$node_(13) setdest 5 400 0.0"

$node_(36) set X_ 5.0
$node_(36) set Y_ 600.0
$node_(36) set Z_ 0.0
$ns_ at .0 "$node_(36) setdest 5 600 0.0"

$node_(4) set X_ 1600.0
$node_(4) set Y_ 200.0
$node_(4) set Z_ 0.0
$ns_ at .0 "$node_(4) setdest 1600 200 0.0"

$node_(5) set X_ 1600.0
$node_(5) set Y_ 600.0
$node_(5) set Z_ 0.0
$ns_ at .0 "$node_(5) setdest 1600 600.0 0.0"

$node_(6) set X_ 200.0
$node_(6) set Y_ 200.0
$node_(6) set Z_ 0.0
$ns_ at .0 "$node_(6) setdest 200 200 0.0"

$node_(7) set X_ 200.0
$node_(7) set Y_ 400.0
$node_(7) set Z_ 0.0
$ns_ at .0 "$node_(7) setdest 200 400 0.0"


$node_(8) set X_ 200.0
$node_(8) set Y_ 600.0
$node_(8) set Z_ 0.0
$ns_ at .0 "$node_(8) setdest 200 600 0.0"


$node_(49) set X_ 200.0
$node_(49) set Y_ 800.0
```

```
$node_(49) set Z_ 0.0
$ns_ at .0 "$node_(49) setdest 200 800 0.0"

$node_(10) set X_ 400.0
$node_(10) set Y_ 5.0
$node_(10) set Z_ 0.0
$ns_ at .0 "$node_(10) setdest 400 5 0.0"

$node_(0) set X_ 400.0
$node_(0) set Y_ 200.0
$node_(0) set Z_ 0.0
$ns_ at .0 "$node_(0) setdest 400 200 0.0"

$node_(12) set X_ 400.0
$node_(12) set Y_ 400.0
$node_(12) set Z_ 0.0
$ns_ at .0 "$node_(12) setdest 400 400 0.0"

$node_(1) set X_ 400.0
$node_(1) set Y_ 600.0
$node_(1) set Z_ 0.0
$ns_ at .0 "$node_(1) setdest 400 600 0.0"

$node_(14) set X_ 400.0
$node_(14) set Y_ 800.0
$node_(14) set Z_ 0.0
$ns_ at .0 "$node_(14) setdest 400.0 800 0.0"

$node_(15) set X_ 600.0
$node_(15) set Y_ 5.0
$node_(15) set Z_ 0.0
$ns_ at .0 "$node_(15) setdest 600 5 0.0"

$node_(16) set X_ 600.0
$node_(16) set Y_ 200.0
$node_(16) set Z_ 0.0
$ns_ at .0 "$node_(16) setdest 600.0 200 0.0"

$node_(17) set X_ 600.0
$node_(17) set Y_ 400.0
$node_(17) set Z_ 0.0
$ns_ at .0 "$node_(17) setdest 600.0 400 0.0"

$node_(18) set X_ 600.0
$node_(18) set Y_ 600.0
$node_(18) set Z_ 0.0
$ns_ at .0 "$node_(18) setdest 600.0 600 0.0"

$node_(19) set X_ 600.0
$node_(19) set Y_ 800.0
$node_(19) set Z_ 0.0
$ns_ at .0 "$node_(19) setdest 600.0 800 0.0"

$node_(20) set X_ 800.0
$node_(20) set Y_ 5.0
$node_(20) set Z_ 0.0
$ns_ at .0 "$node_(20) setdest 800.0 5 0.0"


$node_(21) set X_ 800.0
```

```
$node_(21) set Y_ 200.0
$node_(21) set Z_ 0.0
$ns_ at .0 "$node_(21) setdest 800 200 0.0"


$node_(22) set X_ 800.0
$node_(22) set Y_ 400.0
$node_(22) set Z_ 0.0
$ns_ at .0 "$node_(22) setdest 800 400 0.0"

$node_(23) set X_ 800.0
$node_(23) set Y_ 600.0
$node_(23) set Z_ 0.0
$ns_ at .0 "$node_(23) setdest 800 600 0.0"

$node_(24) set X_ 800.0
$node_(24) set Y_ 800.0
$node_(24) set Z_ 0.0
$ns_ at .0 "$node_(24) setdest 800 800 0.0"

$node_(25) set X_ 1000.0
$node_(25) set Y_ 5.0
$node_(25) set Z_ 0.0
$ns_ at 0 "$node_(25) setdest 1000 5 0.0"
$node_(26) set X_ 1400.0
$node_(26) set Y_ 200.0
$node_(26) set Z_ 0.0
$ns_ at 0 "$node_(26) setdest 1400 200 0.0"

$node_(27) set X_ 1000.0
$node_(27) set Y_ 400.0
$node_(27) set Z_ 0.0
$ns_ at 0 "$node_(27) setdest 1000 400 0.0"


$node_(28) set X_ 1400.0
$node_(28) set Y_ 600.0
$node_(28) set Z_ 0.0
$ns_ at 0 "$node_(28) setdest 1400 600 0.0"

#======================================================================
# Here the traffic is attached with the nodes

#Create a udp agent and attach it to node n0
set udp0 [new Agent/UDP]
$ns_ attach-agent $node_(0) $udp0
# Create a CBR traffic source and attach it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.1
$cbr0 attach-agent $udp0
#Create a Null agent (a traffic sink) and attach it to node n0
set null0 [new Agent/Null]
$ns_ attach-agent $node_(0) $null0

#======================================================================
#Create a udp agent and attach it to node n1
set udp1 [new Agent/UDP]
#$udp1 set class_ 2
$ns_ attach-agent $node_(1) $udp1
```

```
# Create a CBR traffic source and attach it to udp1
set cbr1 [new Application/Traffic/CBR]
$cbr1 set packetSize_ 500
$cbr1 set interval_ 0.1
$cbr1 attach-agent $udp1

#Create a Null agent (a traffic sink) and attach it to node n1
set null1 [new Agent/Null]
$ns_ attach-agent $node_(1) $null1

#========================================================================
#Create a udp agent and attach it to node n1
set udp2 [new Agent/UDP]
#$udp1 set class_ 2
$ns_ attach-agent $node_(2) $udp2

# Create a CBR traffic source and attach it to udp1
set cbr2 [new Application/Traffic/CBR]
$cbr2 set packetSize_ 500
$cbr2 set interval_ 0.1
$cbr2 attach-agent $udp2

#Create a Null agent (a traffic sink) and attach it to node n2
set null2 [new Agent/Null]
$ns_ attach-agent $node_(2) $null2
#========================================================================
#Create a udp agent and attach it to node n3
set udp3 [new Agent/UDP]
#$udp1 set class_ 3
$ns_ attach-agent $node_(3) $udp3
#========================================================================
#Create a udp agent and attach it to node n1
set udp21 [new Agent/UDP]
#$udp1 set class_ 2
$ns_ attach-agent $node_(21) $udp21

# Create a CBR traffic source and attach it to udp1
set cbr21 [new Application/Traffic/CBR]
$cbr21 set packetSize_ 500
$cbr21 set interval_ 0.1
$cbr21 attach-agent $udp21

#Create a Null agent (a traffic sink) and attach it to node n1
set null21 [new Agent/Null]
$ns_ attach-agent $node_(21) $null21
#========================================================================

#Create a udp agent and attach it to node n1
set udp22 [new Agent/UDP]
#$udp1 set class_ 2
$ns_ attach-agent $node_(22) $udp22

# Create a CBR traffic source and attach it to udp1
set cbr22 [new Application/Traffic/CBR]
$cbr22 set packetSize_ 500
$cbr22 set interval_ 0.1
$cbr22 attach-agent $udp22

#Create a Null agent (a traffic sink) and attach it to node n1
```

```
set null22 [new Agent/Null]
$ns_ attach-agent $node_(22) $null22
#========================================================================

#Create a udp agent and attach it to node n1
set udp23 [new Agent/UDP]
#$udp1 set class_ 2
$ns_ attach-agent $node_(23) $udp23

# Create a CBR traffic source and attach it to udp1
set cbr23 [new Application/Traffic/CBR]
$cbr23 set packetSize_ 500
$cbr23 set interval_ 0.1
$cbr23 attach-agent $udp23

#Create a Null agent (a traffic sink) and attach it to node n1
set null23 [new Agent/Null]
$ns_ attach-agent $node_(23) $null23
#========================================================================

#Create a udp agent and attach it to node n1
set udp24 [new Agent/UDP]
#$udp1 set class_ 2
$ns_ attach-agent $node_(24) $udp24

# Create a CBR traffic source and attach it to udp1
set cbr24 [new Application/Traffic/CBR]
$cbr24 set packetSize_ 500
$cbr24 set interval_ 0.1
$cbr24 attach-agent $udp24

#Create a Null agent (a traffic sink) and attach it to node n1
set null24 [new Agent/Null]
$ns_ attach-agent $node_(24) $null24
#========================================================================

# Create a CBR traffic source and attach it to udp3
set cbr3 [new Application/Traffic/CBR]
$cbr3 set packetSize_ 500
$cbr3 set interval_ 0.1
$cbr3 attach-agent $udp3

#Create a Null agent (a traffic sink) and attach it to node n2
set null3 [new Agent/Null]
$ns_ attach-agent $node_(3) $null3

#========================================================================

#Create a udp agent and attach it to node n0
set udp4 [new Agent/UDP]
$ns_ attach-agent $node_(4) $udp4
# Create a CBR traffic source and attach it to udp0
set cbr4 [new Application/Traffic/CBR]
$cbr4 set packetSize_ 500
$cbr4 set interval_ 0.1
$cbr4 attach-agent $udp4
#Create a Null agent (a traffic sink) and attach it to node n0
set null4 [new Agent/Null]
$ns_ attach-agent $node_(4) $null4
```

```
#======================================================================
#Create a udp agent and attach it to node n1
set udp5 [new Agent/UDP]
#$udp1 set class_ 2
$ns_ attach-agent $node_(5) $udp5

# Create a CBR traffic source and attach it to udp1
set cbr5 [new Application/Traffic/CBR]
$cbr5 set packetSize_ 500
$cbr5 set interval_ 0.1
$cbr5 attach-agent $udp5

#Create a Null agent (a traffic sink) and attach it to node n1
set null5 [new Agent/Null]

#======================================================================

$ns_ connect $udp63 $null63
$ns_ at 10.1 "$cbr0 start"
$ns_ at 200 "$cbr0 stop"

$ns_ connect $udp0 $null1
$ns_ at 0.1 "$cbr0 start"
$ns_ at 90 "$cbr0 stop"

$ns_ connect $udp1 $null2
$ns_ at 0.5 "$cbr1 start"
$ns_ at 80.0 "$cbr1 stop"


$ns_ connect $udp2 $null3
$ns_ at 0.5 "$cbr2 start"
$ns_ at 80.0 "$cbr2 stop"


$ns_ connect $udp3 $null4
$ns_ at 0.5 "$cbr3 start"
$ns_ at 80.0 "$cbr3 stop"

$ns_ connect $udp4 $null5
$ns_ at 0.5 "$cbr4 start"
$ns_ at 800.0 "$cbr4 stop"

$ns_ connect $udp5 $null6
$ns_ at 0.5 "$cbr5 start"
$ns_ at 90.0 "$cbr5 stop"

$ns_ connect $udp6 $null7
$ns_ at 0.5 "$cbr6 start"
$ns_ at 90.0 "$cbr6 stop"


$ns_ connect $udp7 $null8
$ns_ at 0.5 "$cbr7 start"
$ns_ at 80.0 "$cbr7 stop"


$ns_ connect $udp8 $null9
$ns_ at 0.5 "$cbr8 start"
$ns_ at 80.0 "$cbr8 stop"
```

```
$ns_ connect $udp9 $null10
$ns_ at 0.5 "$cbr9 start"
$ns_ at 80.0 "$cbr9 stop"

$ns_ connect $udp10 $null11
$ns_ at 0.5 "$cbr10 start"
$ns_ at 80.0 "$cbr10 stop"

$ns_ connect $udp11 $null12
$ns_ at 0.5 "$cbr11 start"
$ns_ at 80.0 "$cbr11 stop"

$ns_ connect $udp12 $null13
$ns_ at 0.5 "$cbr12 start"
$ns_ at 90.0 "$cbr12 stop"

$ns_ connect $udp13 $null14
$ns_ at 0.5 "$cbr13 start"
$ns_ at 80.0 "$cbr13 stop"

$ns_ connect $udp14 $null15
$ns_ at 0.5 "$cbr14 start"
$ns_ at 90.0 "$cbr14 stop"

$ns_ connect $udp15 $null16
$ns_ at 0.5 "$cbr15 start"
$ns_ at 80.0 "$cbr15 stop"

$ns_ connect $udp16 $null17
$ns_ at 0.5 "$cbr16 start"
$ns_ at 90.0 "$cbr16 stop"

$ns_ connect $udp17 $null18
$ns_ at 0.5 "$cbr17 start"
$ns_ at 90.0 "$cbr17 stop"

$ns_ connect $udp18 $null19
$ns_ at 0.5 "$cbr18 start"
$ns_ at 80.0 "$cbr18 stop"

$ns_ connect $udp19 $null20
$ns_ at 0.5 "$cbr19 start"
$ns_ at 80.0 "$cbr19 stop"

$ns_ connect $udp20 $null21
$ns_ at 0.5 "$cbr20 start"
$ns_ at 90.0 "$cbr20 stop"

$ns_ connect $udp21 $null22
$ns_ at 0.5 "$cbr20 start"
$ns_ at 90.0 "$cbr20 stop"

$ns_ connect $udp22 $null23
$ns_ at 0.5 "$cbr20 start"
$ns_ at 90.0 "$cbr20 stop"

$ns_ connect $udp23 $null24
$ns_ at 0.5 "$cbr20 start"
$ns_ at 90.0 "$cbr20 stop"
```

```
$ns_ connect $udp24 $null25
$ns_ at 0.5 "$cbr20 start"
$ns_ at 90.0 "$cbr20 stop"

$ns_ connect $udp25 $null26
$ns_ at 0.5 "$cbr20 start"
$ns_ at 90.0 "$cbr20 stop"

$ns_ connect $udp26 $null27
$ns_ at 0.5 "$cbr20 start"
$ns_ at 90.0 "$cbr20 stop"

$ns_ connect $udp27 $null28
$ns_ at 0.5 "$cbr20 start"
$ns_ at 90.0 "$cbr20 stop"
#======================================================================

# Tell nodes when the simulation ends

for {set i 0} {$i < $val(nn)} {incr i} {
   # $ns_ at $val(stop).0 "$node_($i) dump";
    $ns_ at $val(stop).2000 "$node_($i) reset";
}
$ns_ at $val(stop) "stop"
$ns_ at $val(stop).1 "puts \"NS EXITING...\" ; $ns_ halt"

proc stop {} {
    global ns_ tracefd
    $ns_ flush-trace
    close $tracefd
puts "dd..."
}
#======================================================================
puts "Starting Simulation..."
$ns_ run
#======================================================================
```

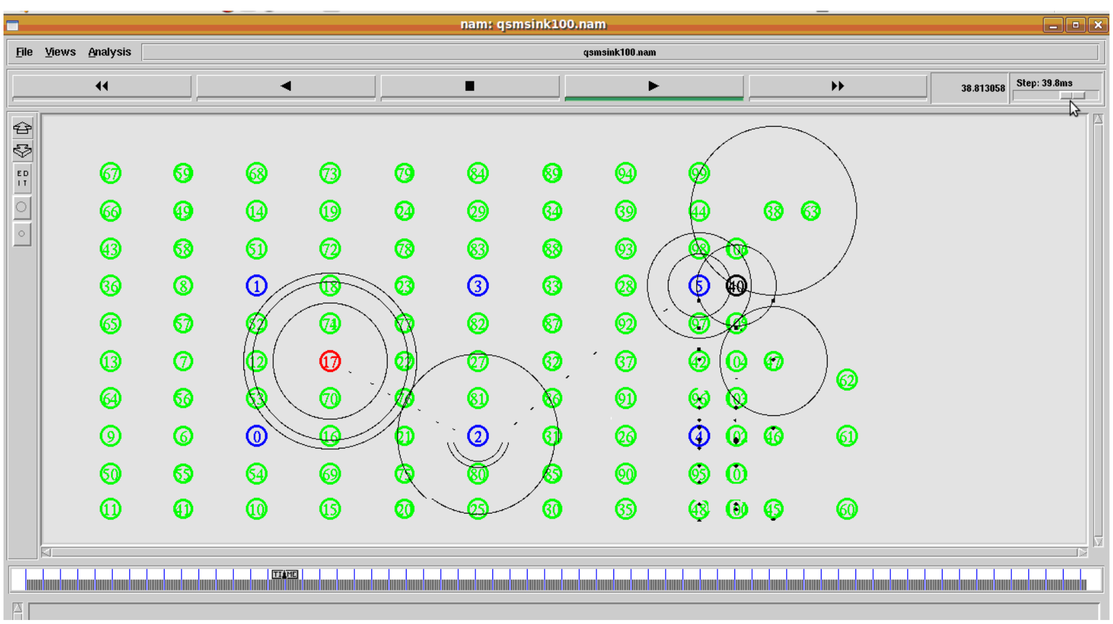

Figure 20: Simulation in NS-2 NAM