PROJECT REPORT ON

Genetic Algorithm based Cluster Head Selection for Optimized Communication in Wireless Sensor Network.

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

BACHELOR OF TECHNOLOGY
IN
INFORMATION TECHNOLOGY

Submitted by

118015151-Yarlagadda Venkata Subba Rao-IT

118015055-Kanampalli Sai Kiran Reddy-IT

Under the Guidance of

Prof. Dr. Saravanan. P
AP - III/CSE

School of Computing
SASTRA DEEMED TO BE UNIVERSITY

(A University established under section 3 of the UGC Act,1956)
Tirumalaisamudiram
Thanjavur-613 401

April (2018)

SHANMUGHA
ARTS, SCIENCE, TECHNOLOGY & RESEARCH ACADEMY
(SASTRA DEEMED TO BE UNIVERSITY)
(A University Established under section 3 of the UGC Act, 1956)
TIRUMALAISAMUDRAM, THANJAVUR – 613401

BONAFIDE CERTIFICATE

Certified that this project work entitled "**Genetic Algorithm based Cluster Head Selection for Optimized Communication in Wireless Sensor Network"** submitted to the Shanmugha Arts, Science, Technology & Research Academy (SASTRA Deemed to be University), Tirumalaisamudram -613401 by 118015151 - Yarlagadda Venkata Subba Rao-IT and 118015055 - Kanampalli Sai Kiran Reddy-IT, in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in **INFORMATION TECHNOLOGY**. This work is an original and independent work carried out under my guidance, during the period December 2017 - April 2018.

Dr Saravanan P	**ASSOCIATE DEAN**
	DR. K S RAVICHANDRAN
SCHOOL OF COMPUTING	**SCHOOL OF COMPUTING**

Submitted for Project Viva Voice held on___________________

Examiner –I	**Examiner-II**

ACKNOWLEDGEMENT

We would forever remain grateful to honourable **Prof. R. Sethuraman, Vice Chancellor** for his encouragement in our academic life at SASTRA University.

We wish to express my profound gratitude **to Dr. S. Vaidhyasubramaniam, Dean - Planning & Development and Dr S. Swaminathan ,Dean Sponsored research** for his overwhelming support provided during our course span in SASTRA University

We are extremely thankful to **Dr. K. S. Ravichandran,** Associate Dean, Department of Information Technology for providing us the opportunity to do this project and for all the academic help done for the past three years.

We sincerely express our gratitude to our mentor **Dr Saravanan P**, Assistant Professor-III, School of Computing for his assistance and guidance for the successful implementation of project in a systematic and professional manner.

We also thank all the Teaching and Non-teaching staff, and those who have directly or indirectly helped us by extending their moral support and encouragement for completion of this project.

ABSTRACT

Wireless Sensor Network (WSNs) utilizes conveyed gadgets sensors for observing physical or natural conditions. It has been given to the steering conventions which may contrast contingent upon the application and system design. Vitality administration in WSN is of incomparable significance for the remotely sent vitality sensor hubs. The hubs can be obliged in the little gatherings called the Clusters. Clustering is done to accomplish the vitality effectiveness and the versatility of the system. Development of the group likewise includes the doling out the part to the hub based on their borders. In this paper, a novel strategy for cluster head selection based on Genetic Algorithm (GA) has been proposed. Every person in the GA populace speaks to a conceivable answer for the issue. Discovering people who are the best proposals to the enhancement issue and join these people into new people is a critical phase of the transformative procedure. The Cluster Head (CH) is picked using the proposed technique Genetic Algorithm based Cluster Head (GACH). The performance of the proposed system GACH has been compared with Particle Swarm Optimization Cluster Head (PSOCH). Simulations have been conducted with 14 wireless sensor nodes scattered around 8 kilometers. Results prove that GACH outperforms than PSOCH regarding throughput, packet delivery ratio, and energy efficiency.

TABLE OF CONTENTS

CHAPTER 7 TESTING

CHAPTER 8 IMPLEMENTATION

CHAPTER 1

INTRODUCTION

1.1 WIRELESS NETWORK

Wireless networks are computer networks that are not connected by cables of any kind. The use of a wireless network enables enterprises to avoid the costly process of introducing cables into buildings or as a connection between different equipment locations. The basis of wireless systems is radio waves, an implementation that takes place at the physical level of network structure. They use radio waves to connect devices such as laptops to the Internet, the network, and applications. When laptops are connected to Wi-Fi hot spots in public places, the connection is established to that business's wireless network.

1.2 TYPES OF WIRELESS NETWORKS

1.2.1 WIRELESS PAN

Wireless Personal Area Networks (WPANs) are internet devices within a relatively small area that is generally within a person's reach. ZigBee also supports WPAN applications.

1.2.2 WIRELESS LAN

Wireless LANs are often used for connecting local resources to the Internet. A wireless local area network (WLAN) links two or more devices over a short distance using a wireless distribution method, usually providing a connection through an access point for internet access. The use of spread-spectrum or OFDM technologies may allow users to move around within a local coverage area and remain connected to the network.

Products using the IEEE 802.11 WLAN standards are marketed under the Wi-Fi brand name. Fixed wireless technology implements point-to-point links between computers or networks at two distant locations often using the dedicated microwave or modulated laser light beams over the line of sight paths. It is often used in cities to connect networks in two or more buildings without installing a wired link.

1.2.3 WIRELESS AD HOC NETWORK

A wireless ad-hoc network, also known as a wireless mesh network or mobile ad hoc network (MANET) is a wireless network made up of radio nodes organized in a mesh topology. Each node forwards messages on behalf of the other nodes and each node perform routing. Ad hoc networks can "self-heal," automatically re-routing around a node that has lost power. Various network layer protocols are needed to realize ad hoc mobile networks, such as Distance Sequenced Distance Vector Routing, Associability – Based Routing, Ad hoc on – demand Distance Vector Routing and Dynamic Source Routing.

1.2.4 WIRELESS MAN

Wireless metropolitan area networks are a type of wireless network that connects several wireless LANs. Although the 802.16 family of standards is officially called Wireless MAN in IEEE, it has been commercialized under the name "WiMAX" (from "Worldwide Interoperability for Microwave Access") by the WiMAX Forum industry alliance. The Forum promotes and certifies compatibility and interoperability of products based on the IEEE 802.16 standards.

1.2.5 WIRELESS WAN

Wireless wide area networks are wireless networks that typically cover large areas, such as between neighboring towns and cities, or city and suburb. These networks can be used to connect branch offices of business or as a public Internet access system. The wireless connections between access points are usually pointing to point microwave links using parabolic dishes on the 2.4 GHz band, rather than omni directional antennas used with smaller networks. A typical system contains base station gateways, access points, and wireless bridging relays. Other configurations are mesh systems where each access point acts as a relay also. When combined with renewable energy systems such as photovoltaic solar panels or wind systems they can be stand-alone systems.

1.3 APPLICATIONS OF WIRELESS NETWORK

Applications of a wireless network include cell phones which are part of everyday wireless networks allowing easy personal communications. Another application is Intercontinental network systems which use radio satellites to communicate across the world. Emergency services such as the police and the ambulance utilize wireless networks to communicate effectively as well. Individuals and business people use wireless networks to send and share data rapidly whether it be in a small office building or across the world.

1.4 WIRELESS SENSOR NETWORK (WSN)

In a Wireless Sensor Network (WSN), sink(s) can be either static or mobile. Existing work shows that sink mobility brings many advantages. A representative example for mobile sinks is a group of soldiers with a handheld PDAs patrolling in the battlefield to gathering information from the sensor network deployed on the ground.

For data gathering, it is critical for sensors to report packets to nearby mobile sinks. However, sink mobility can cause unexpected changes of network topology, and the increase of sink number may bring excessive overhead for route maintenance. Therefore, the performance of a mWSN highly depends on how the routing protocol can be designed to increase the network performance while suppressing the protocol overhead caused by the mobility of sinks.

Effective routing with low protocol overhead in mWSNs is challenging as sinks move freely and unpredictably. Inappropriate selection of routes for data gathering in an mWSN could introduce much more unnecessary overhead, which can degrade the performance of the mWSN. Although many mechanisms have been designed for static wireless sensor networks, most of them are inappropriate for dynamic mWSNs. Flooding has high reliability in packet delivery but causes extremely high protocol overhead. Random walk (or gossip) has been used for information dissemination/discovery in WSNs when no knowledge about the network is known. A random walk works in the following way: When a node sends a packet to a target node, it randomly picks one node from its neighbors and then forwards the packet to the selected neighbor. This process continues until the packet reaches the target node or timed out and then discarded. A random walk is simple, localized, and robust but causes very large delivery delay. Data MULEs are used for collecting data from sparser sensor networks and targeted for delay-tolerant applications. Data mules can pick up data from sensors when in short range, buffer it, and drop off the data to wired access points when in proximity. Their advantage is protocol simplicity and very low protocol overhead for maintaining multi-hop network structure. Their primary disadvantage is long delivery latency (e.g., a few hours or even days). In some existing work, data gathering with controlled sink movement is to optimize the network performance such as network lifetime or ease the protocol design; controlled sink mobility are considered such that a sink moves along a predetermined optimized trajectory or paths meeting certain criteria.

1.5 CLUSTER HEADS (CH)

The set of CHs in a WSN forms its backbone, providing a scalable solution to various networking tasks, such as data collection and habitat monitoring. At each cluster, a CH is responsible for various tasks, e.g., node association, authentication, and task assignment. The CH also maintains the cluster structure when node-centric events occur, such as hardware failures and sensor mobility. Support for traffic sharing, cluster membership, and inter-cluster connectivity are provided by collaborative discussions over the inter-CH links of the network backbone. Therefore, as a central control point of a cluster, a CH has considerably higher energy consumption compared to cluster members. It requires a high load of CHs is distributed among all nodes.

1.6 TRAFFIC HOT-SPOTS

Periodic reassignment of the CH role to different nodes helps prevent the problem of a single point of failure in the event of node energy depletion. However, traffic hotspots in a WSN also pose error-prone situations. It is particularly important since clustered WSNs are mainly focused on data gathering applications (e.g., habitat monitoring and military surveillance), which involve periodic delivery of sensory data over multihop routes, creating highly congested areas, especially at locations close to a data sink (e.g., a control centre). Furthermore, there may also be other critically-located sensors not necessarily close to data sinks, which carry the burden of relaying large amounts of data traffic, especially when multiple high-rate routes pass through these sensors. Such nodes are usually frequently chosen to be data relays by routing algorithms and may serve a large portion of the network traffic, due to their convenient locations. Thus, avoiding the failure of such nodes caused by early energy depletion is critical to ensure a sufficiently long network lifetime.

1.7 LOCALIZATION

It is defined as the method of deciding the position of the hubs in a system. Since most applications rely upon a fruitful limitation, i.e. to figure their situations in some settled organize framework, it is of awesome significance to plan productive confinement calculations. If there should be an occurrence of the static system, the hubs are conveyed once the system has setup, in the event of the portable hubs the area of hubs changes ceaselessly. Confinement is evaluated throughout correspondence amongst restricted and un-limited hub for deciding their geometrical position; area is controlled by methods for separation and point between hubs. The sending is either deterministic or self-sorting out. In deterministic circumstances, the sensors are physically set, and the information is steered through pre-decided ways, where as in self-sorting out frameworks, the sensor hubs are scattered haphazardly making a foundation in a specially appointed manner.WSN issues, for example, hub sending, confinement, vitality mindful grouping and information accumulation are frequently planned as enhancement issues. The restriction is considered as the essential device for the arrangement of ease sensor organizes as it characterizes the location of the hub in the system. In a system thousand of hubs are accessible to every one of these hubs and they are battery controlled. Correspondence, handling and the detecting activity of the hubs are very costly.

So arrangement of the hubs in the system ought to be simple.

1.8 SENSOR NETWORK CLUSTERING

WSN base station dependably needs to produce an accumulated an incentive to the end clients and the anthology of the information to be sent can help in decreasing the communication overhead and the vitality utilization. To help the information accumulation in the system, the hubs can be suited in the little gatherings called the Clusters. Bunching can be characterized as the division of the hubs in the gatherings based on some component. Bunching has been appeared to enhance organize lifetime, an essential metric for assessing the execution of a sensor arrange. Grouping is done to accomplish the vitality proficiency and the versatility of the system. The arrangement of the bunch likewise includes the doling out the part to the hub based on their edges. The facilitator of the bunch which is in charge of the handling, accumulation, and transmission of the information to the base station is known as the Cluster Head (CH) or the pioneer, while alternate hubs which are in charge of detecting and transfer the gathered information to the CH are known as the Member Nodes.

1.9 LITERATURE SURVEY

In [1], Xue Wang et al., proposes a dynamic algorithm which utilizes numerous swarms to enhance distinctive segments of the arrangement vector and the speed of every molecule is refreshed by worldwide and nearby ideal arrangements as well as the virtual powers of sensor hubs.In [2], Guangshun Yao et al. analyzes the generating of invalid new individual problem. An improved genetic algorithm is proposed by considering the position and neighbor nodes in WSNs. This algorithm considers many factors like the residual energy of nodes, distance, and energy consumption between adjacent nodes to select suitable routing mechanism. In [3], Ali Norouziet.al., investigates Genetic Algorithm as an energetic approach to discover ideal states. In [4], Harmeet Kauret.al. proposes a localization approach based on genetic algorithm for WSN. They have proposed a new way to calculate the distance between anchor node and an unknown node. The algorithm has to cover every node. The condition is that no node should be repeated. In [5], Lieping Zhang et al. implements a cross-breed improvement in light of differential development and particle swarm streamlining calculations. The location and speed of the underlying populace are arbitrarily created by PSO, and the wellness work is developed. At that point, the transformation and choice task of DE estimation is executed to discover the perfect location of the populace. The present speeds and places of every particle of the populace are refreshed, and the hybrid activity and determination task of DE calculation are executed to refresh the current worldwide perfect location of the complete populace. In [6], Pei-Hsuan Tsai proposes an adaptable lattice-based confinement plot for building up relative facilitate arrangement of sensor systems. A circulated choice calculation is proposed to choose few hubs from the arrangement of sensor hubs as virtual network hubs. The virtual lattice hubs are then used to set up matrix arrange framework by utilizing molecule swarm advancement.

In [7], Y. Gu exploits sink versatility to draw out the system lifetime in remote sensor systems (WSNs) where the data delay caused by moving the sink ought to be limited. They have examined about the instigated sub-issues and present productive answers for them. At that point, summed up these arrangements and propose a polynomial-time ideal calculation for the caused issue. In [8], C. Tunca, proposes Ring Routing, a novel, circulated, vitality effective, versatile sink directing convention, reasonable for time-delicate applications, which expects to limit this overhead while protecting the upsides of portable sinks. In [9], Hosseinirad presented another improvement calculation called Imperialist Competitive Algorithm (ICA) which is roused by the socio-political procedure of imperialistic rivalry. Utilizations ICA for CH choice as indicated by the CE cost. It exhibits that ICA is a powerful strategy for choice of CH in WSN. In [10], Moslem presents a dynamic bunching estimate utilizing heritable computation. This computation thinks about various parameters to build the system lifetime. In [11], Omar Banimelhem presents another hereditary based approach that enhances the execution of the LEACH bunching convention utilized as a part of remote sensor systems. The proposed approach uses the portability highlight of sensor hubs to lessen the correspondence removes between the bunch heads and the base station. In each round, new areas of the bunch heads are resolved to utilize a hereditary calculation. In [12], Shekh describes the consumption of PSO calculation for ideal sensor sending in WSN. The planned work can accomplish the ideal collection of uniting issue with least number of sensors in the remote system. The outcomes demonstrate that PSO approach is powerful and vigorous for productive scale issue of sensor sending and is considered to give nearly the ideal arrangements in WSN. In [13], Erdogan proposed a genetic algorithm based method (GABEEC) to optimize the lifetime of wireless sensor networks. The method has 2 phases: Set-up and Steady-state phase. The results show that the proposed method is found to be more efficient than LEACH

CHAPTER 2

SOFTWARE PROJECT PLAN

TASK ID	TASK	DURATION IN WEEKS
1	Problem Identification	2
2	Literature Survey	2
3	Study of PSO and GA in Wireless Sensor Network	3
4	Installation of VM ware software and other requirements for the project	1
5	Implementation of PSO algorithm for selecting CH	2
6	Implementation of Genetic Algorithm for selecting CH	2
7	Evaluation of performance metrics for PSO and GA	1
8	Comparison of Throughput and other metrics	1
9	Document and thesis preparation along with manuscript preparation	3

CHAPTER 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 FUNCTIONAL REQUIREMENTS

3.1.1 Use Case

The Purpose of use case diagram is:

-To reach a common understanding of system behaviour

-To design elements that supports the required behaviour

-To identify test cases

-To plan and assess work

-To write user documentation

3.2 Non-Functional Requirements

3.2.1 Resource requirements

3.2.1.1 Hardware requirements

- Processor : Dual core processor 2.6.0 GHz.
- RAM : 1GB.
- Hard disk : 160 GB.
- Compact Disk : 650 MB.
- Keyboard : Standard keyboard.
- Monitor : 15 inch color monitor.

3.2.1.2 Software Requirements

- Operating System : Windows 7.
- Language : NS2 (TCL commands, C++)
- Tool : Cygwin / VM-Ware Workstation

3.2.2 Performance Requirements

This system functions by communicating data using Cluster Head. Cluster Head and Mobile Sink are used to communicate data from sender to receiver. The performance is evaluated using throughput and packet delivery ratio.

3.2.3 Software Quality Attributes

3.2.3.1 Maintainability

The system can do the work for which intended. The VM ware software should be maintained by the software perfectly. If the VM ware is not properly shut down, the software will be corrupted.

3.2.3.2 Security

A measure of system ability to resist the unauthorized attempt at usage or behavior modification, while still providing service to the user.

3.2.3.3 Usability

It is user-friendly.

3.2.3.4 Reliability

The system will perform the stated period without any error.

CHAPTER 4

SYSTEM ANALYSIS

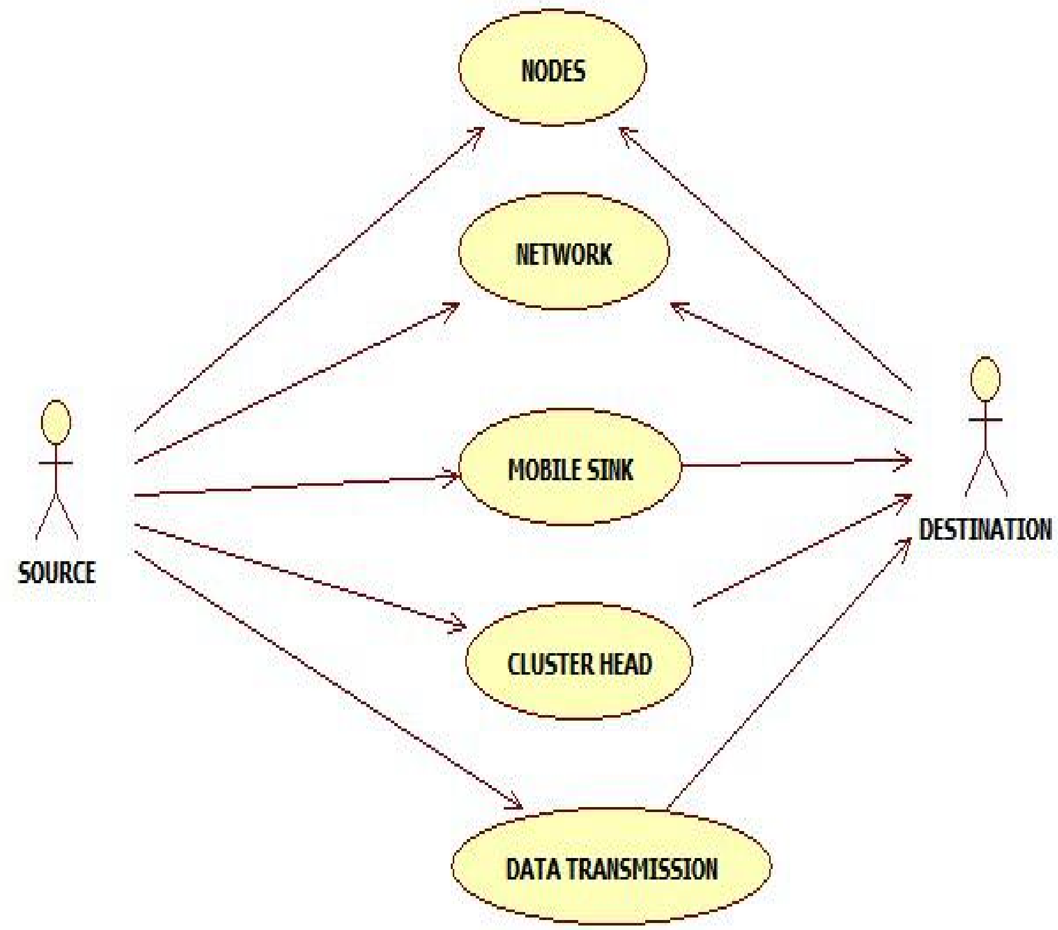


Fig:4.1 USE CASE DIAGRAM

The Use Case Diagram explains how the actors are connected with the use cases. Here, the actors are Sender and Receiver. Source and Destination along with other nodes form a network. The source sends the data to the Cluster Head (CH). The CH forwards the data to the Mobile Sink(MS). MS will send the data to the appropriate CH. Then, the CH will transmit the data to the destination.

CHAPTER 5

SYSTEM DESIGN

5.1 Front End Design

In simple terms, front-end development is used to create the visual display that the end user of an application experiences. When a user attempts to access an application via the front-end interface, the relevant information is verified via the back-end, and the proper information is then presented to the user. The front end typically includes the HTML, CSS, and JavaScript. In our project, we have implemented the front end using Network Simulator 2. The coding is written using Tool Command Language. Since this is a NS2 project, there is no need for back-end design.

5.2 DATA FLOW DIAGRAM

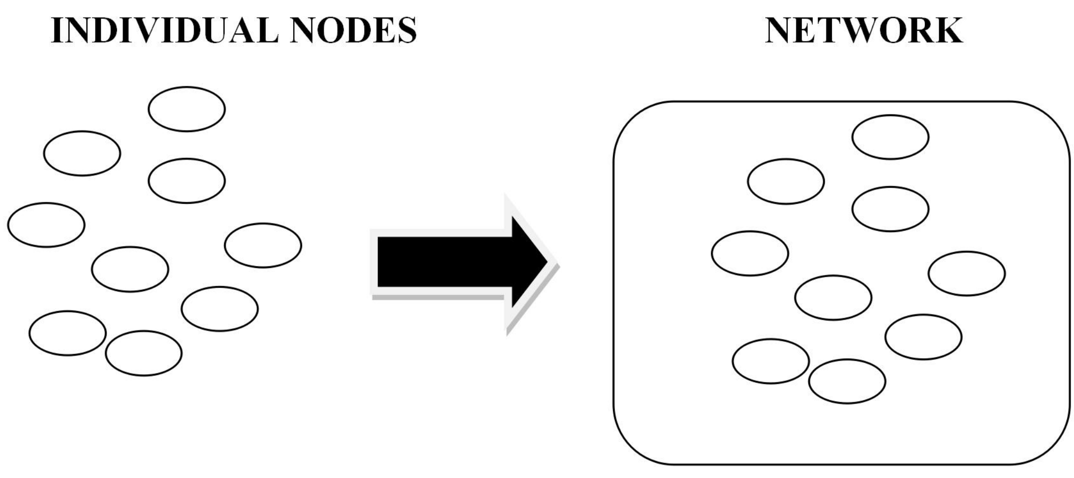
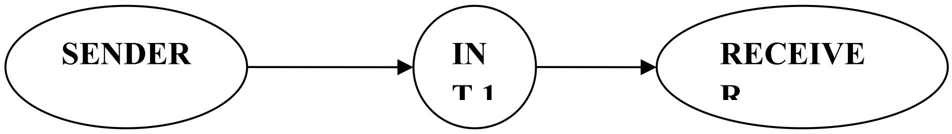


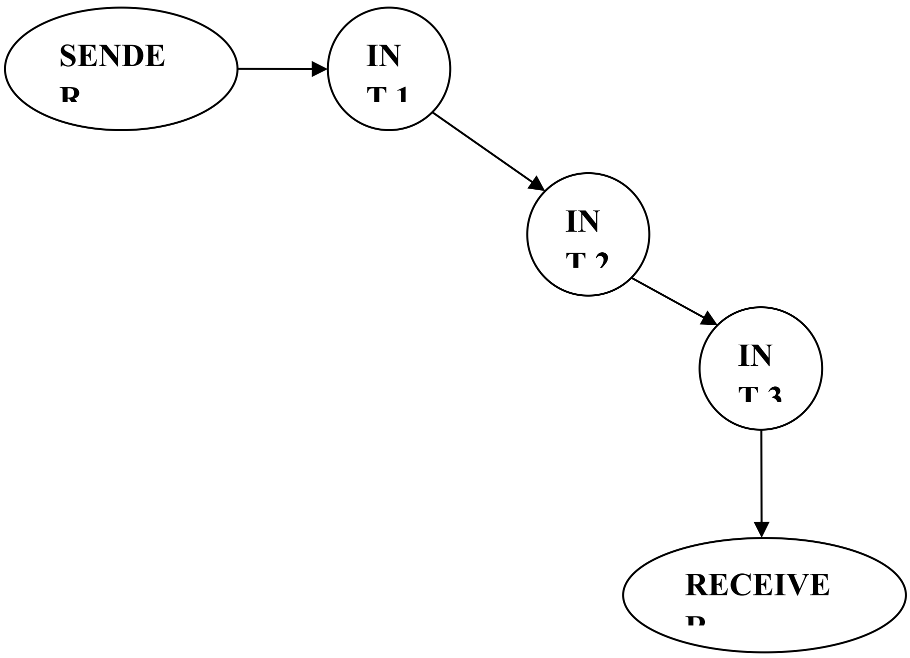
Fig: 5.1 – NETWORK FORMATION

This diagram explains how the individual nodes form the network. Individual nodes will form as a cluster inorder to communicate with other clusters.

SINGLE HOP COMMUNICATION



MULTI HOP COMMUNICATION



INT – Intermediate Nodes

Fig: 5.2 – DATA COMMUNICATION

This diagram explains how the nodes communicate within the cluster. There are two types of communication viz., Single Hop and Multi-Hop Communication. In Single Hop Communication, there will be only one intermediate node between the source and the destination. In Multi-Hop Communication, there will be more than one intermediate nodes between the source and the destination.

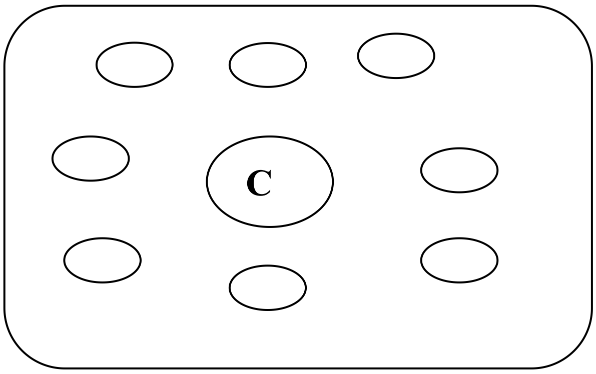


Fig: 5.3 – CLUSTER HEAD SELECTION

This diagram explains how the CH is selected within the cluster. CH is selected based on few parameters. They are (i) residual energy of a node (ii) distance of the node in a cluster (iii) distance among clusters.

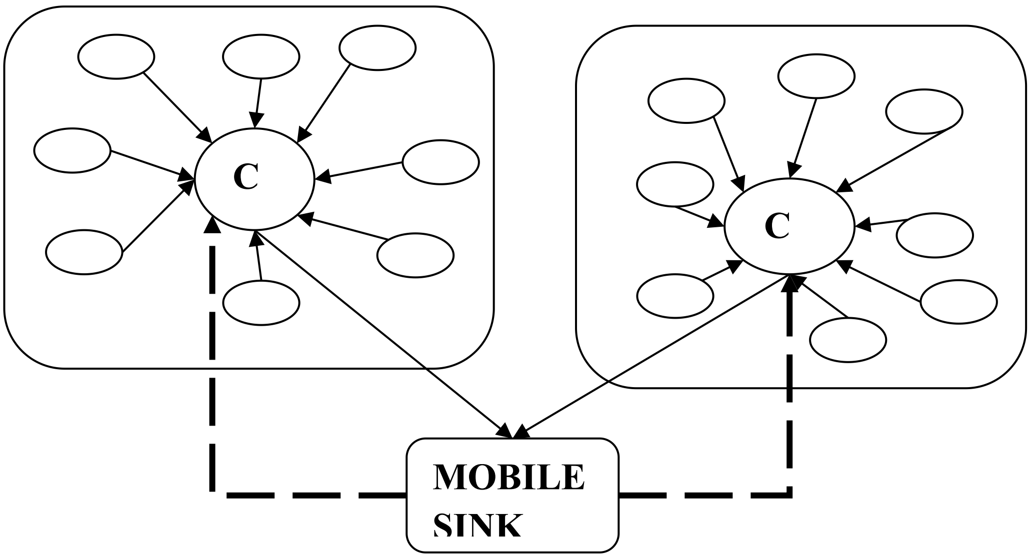


Fig: 5.4 – COMMUNICATION THROUGH CLUSTER HEAD & MOBILE SINK

This diagram explains how the communication is established between the source and the destination. The CH in a cluster will collect the data from the nodes cluster. Then, it will forward the data to the MS. MS will communicate with the appropriate CH. That CH will transmit the data to the respective destinations.

CHAPTER 6

ALGORITHM & CODING

6.1 ALGORITHMS USED

6.1.1 PARTICLE SWARM OPTIMIZATION

PSO is a sort of transformative calculation. The calculation is motivated by the exploration of the social conduct of feathered creature rushing. PSO reproduces the practices of fowl running. In PSO, each single arrangement is a "fledgling" in the inquiry space. We call it "particle". All of particles have wellness esteems which are assessed by the wellness capacity to be improved, and have speeds which coordinate the flying of the particles. The particles are "flown" through the issue space by following the present ideal particles. PSO is a cycle based ideal calculation. The framework is instated with a populace of irregular arrangements and looks for optima by refreshing ages. In the inquiry procedure, PSO consolidates nearby data with worldwide data, that is, a molecule alters its present position as per its verifiable data as well as indicated by the related data of neighboring particles, and finds ideal arrangement through emphasis.

In each emphasis, every molecule is refreshed by following two "best" qualities. The first is the best arrangement (wellness) it has accomplished up until this point. This esteem is called singular best. Another "best" esteem that is followed by the molecule swarm enhancer is the best esteem, got so far by any molecule in the populace. This best esteem is a worldwide best. At the point when a molecule removes a portion of the populace as its topological neighbors, the best esteem is a nearby best. The unbalance of vitality consumption caused diverse separation from the bunch heads is inclined to lead group individuals to kick the bucket effortlessly and along these lines lessen the system lifetime. To guarantee reasonable vitality dispersal in the system, extra parameters could be considered to adjust the transmission vitality utilization of hubs that are both low-vitality and long-separate from group heads. We address the issue by utilizing PSO-based ideal grouping which is spurred from the DSDV. With PSO cycle iterative calculation, ideal bunch heads would be chosen. The portrayal of the calculation is as per the following Cluster - Head definition, Cluster - Heads gathering the data from the hubs, Optimize and decide ideal bunch heads utilizing PSO procedure and optimal group detailing are the different strategies utilized as a part of ideal PSO based bunching.

6.1.2 GENETIC ALGORITHM

Genetic Algorithm is an inspiration algorithm that is derived from science. Data is framed as chromosomes and is consolidated by exceptional tasks, for example, legacy, change, choice, and hybrid. The best chromosome in the populace is chosen considering a wellness work. A common streamlining technique by hereditary calculation may have the accompanying parts:

Population

Genetic algorithm chips away at an arrangement which is named populace. People in populace are a progression of numbers called chromosome and contains double information of the arrangement's parameters.

Fitness function

It gives us a list of person's execution in issue zone. This crude data normally is connected as a center stage for deciding person's relative proficiency in genetic algorithm’s calculation.

$$f(C) = \sum_{i=1}^{N} d_i^2$$

Where f(C) is the energy of a chromosome, N is a number of genes and di is the distance between the i^{th} node and the other nodes.

Selection

Selection is the procedure that decides how frequently every individual can take an interest in multiplication organizes. The quantity of children that each parent can repeat is resolved at this stage.

Crossover

Crossover is the primary specialist in growing new age. Coming about children from this stage is similar to chromosomes in nature, convey a bit of data of their folks. Numerous hybrid procedures exist, the least complex frame is one-point hybrid, and the other kind is multi-focuses hybrid. In one-point hybrid, each parent is separated into two segments from a particular place, and afterward, two youngsters are made by swapping one a player in each parent. In a multi-point hybrid, a few division focuses on chromosomes are chosen. Each parent is partitioned into the few segments in light of the quantity of portion focuses.

Mutation

Mutation is a procedure in which one a player in the chromosome is changed arbitrarily. In the hereditary calculation, it is viewed as that the likelihood of transformation in chromosomes is around 0.01 to 0.001. Utilizing this operator, it is trusted that great chromosomes, evacuated in choice or propagation stages. Transformation additionally ensures that the populace isn't excessively comparable, making it impossible to each other, so GA can maintain a strategic distance from of getting in nearby minima.

The flowchart for Genetic Algorithm is illustrated above. The nodes will form a network. Calculate the residual energy for all the nodes. Then, calculate the average residual energy. Choose a node I. if I’s residual energy is greater than the average residual energy, proceed to the next step. Else, choose another node.

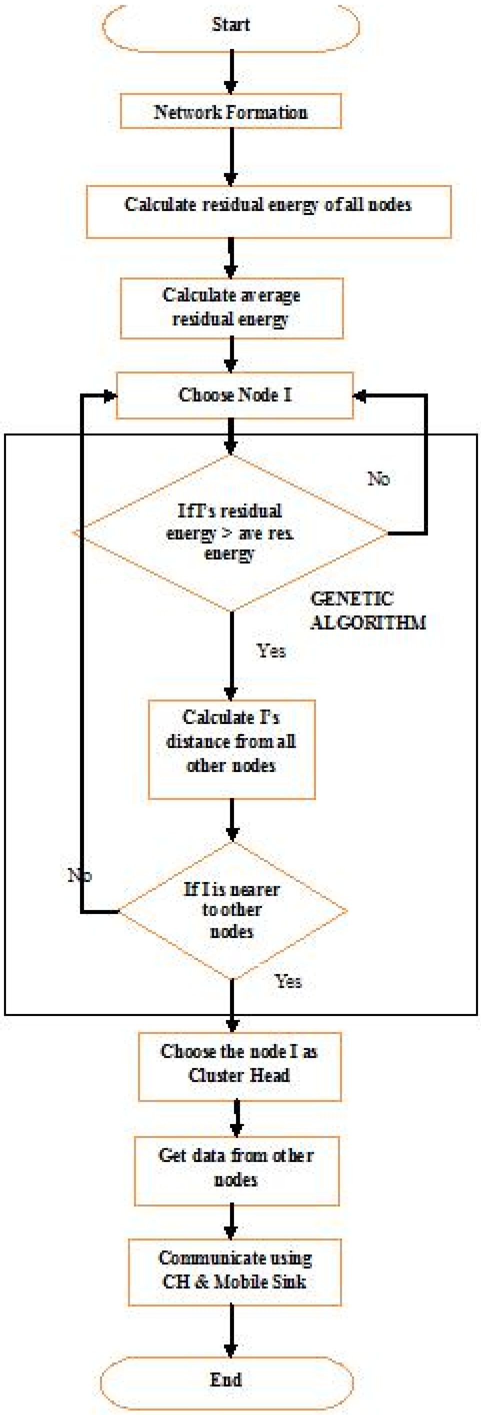


Fig: 6.2 – Flow Chart for Genetic Algorithm

This is the first condition. Now, calculate the distance of node I with other nodes.

If the node I am easily accessible, then choose I as CH. This is the second condition. Else, choose another node and follow the same procedure. Once CH is selected, the nodes in the cluster will send their data to the CH. The Communication will be done using the CH and MS.

6.1.3 Particle Swarm Optimization based Cluster Head selection (PSOCA)

PSO is a sort of transformative calculation. The calculation is motivated by the exploration of the social conduct of feathered creature rushing. PSO reproduces the practices of fowl running. In PSO, every single arrangement is a "fledgling" in the inquiry space and has been called as "particle." They have been evaluated by wellness capacity. The wellness capacity is estimated by speed coordinates of flying particles.

The particles are "flown" through the issue space by following the present ideal particles. PSO is a cycle based ideal calculation. The framework is instated with a populace of irregular arrangements and looks for optima by refreshing ages. In the inquiry procedure, PSO consolidates nearby data with worldwide data, that is, a molecule alters its present position as per its verifiable data as well as indicated by the related data of neighboring particles, and finds ideal arrangement through emphasis. In each emphasis, every molecule is refreshed by following two "best" qualities. The first is the best arrangement (wellness) it has accomplished up until this point. This esteem is called singular best. Another "best" esteem that is followed by the molecule swarm enhancer is the best esteem, got so far by any molecule in the populace. This best esteem is a worldwide best. At the point when a molecule removes a portion of the populace as its topological neighbors, the best esteem is a nearby best. The step-by-step procedure is shown in Fig 6.2.

The particles are "flown" through the issue space by following the present ideal particles. PSO is a cycle based ideal calculation. The framework is instated with a populace of irregular arrangements and looks for optima by refreshing ages.

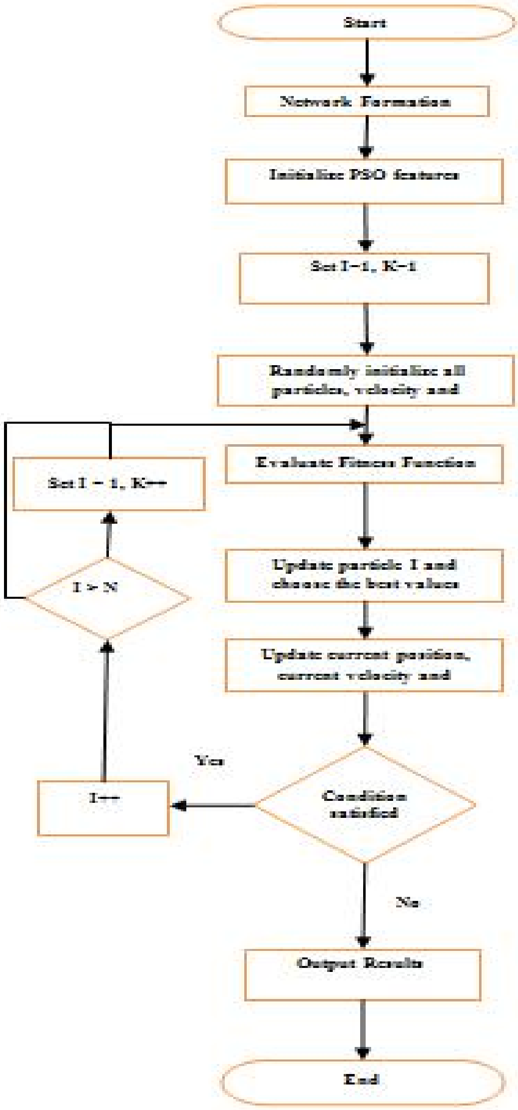


Fig 6.3. Implementation steps in PSOCH

In the inquiry procedure, PSO consolidates nearby data with worldwide data, that is, a molecule alters its present position as per its verifiable data as well as indicated by the related data of neighboring particles, and finds ideal arrangement through emphasis.

In each emphasis, every molecule is refreshed by following two "best" qualities. The first is the best arrangement (wellness) it has accomplished up until this point. This esteem is called singular best. The step-by-step procedure is shown in Fig 6 3.

The unbalance of vitality consumption caused diverse separation from the bunch heads is inclined to lead group individuals to kick the bucket effortlessly and along these lines lessen the system lifetime. Extra parameters could be considered to adjust the transmission vitality utilization of hubs that are both low-vitality and long-separate from group heads. We address the issue by utilizing PSO-based ideal grouping which is spurred from the DSDV. With PSO cycle iterative calculation, ideal bunch heads would be chosen. The portrayal of the calculation is as per the following Cluster - Head definition, Cluster - Heads gathering the data from the hubs, Optimize and decide ideal bunch heads utilizing PSO procedure and optimal group detailing are the different strategies utilized as a part of ideal PSO based bunching.

6.2. RESULTS

Experiments are conducted for 14 nodes with two base station situated in around 7 square kilometers. Geographical Routing Protocol is used, and the cluster head process is done using PSOCH and GACH. The various parameters are tabulated in Table 1.

Table 1. Simulation parameters in NS2

Parameters	Values
Number of sensor nodes	14
Routing Protocol used	GRP
Algorithms used	PSOCH and GACH
Base station used	Two
Quality of Service parameters	Throughput
Mobility Model	Random way point

Fig 6.4 illustrates the formation of a network. The network compromises of nodes and mobile sink. The nodes will communicate among themselves using Mobile Sink.

Fig 5 explains how the Cluster Head is chosen. The Cluster Head is chosen based on few parameters: 1) distance of the chosen node residual energy of all nodes 3) distance of all the nodes from the sink.

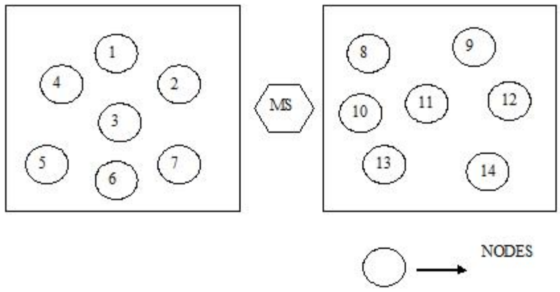


Fig 6.4. Wireless sensor network created in NS2.

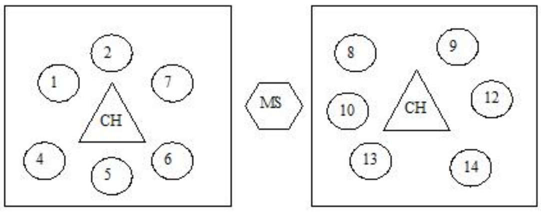


Fig 6. 5. Cluster Head Selection for each cluster

In the figure Fig 6.6, it is explained how data is transferred from source to destination. Consider node four as source and node 14 as a destination. The data from source will transfer to the CH. From CH, it will transfer to MS. From MS; it will be transferred to the CH of the appropriate cluster. The form that CH, the data will be transferred to the destination node.

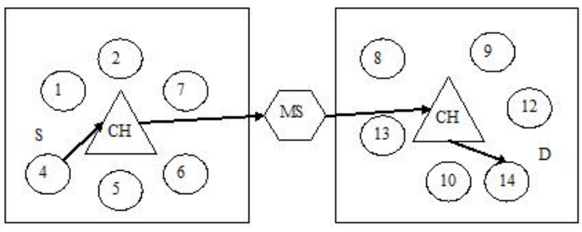


Fig 6.6. Data transmission between source and destination using cluster heads

The performance factors such as throughput, packet delivery ratio, and energy efficiency are calculated and compared. It is inferred that Genetic Algorithm is effective when compared with PSO Algorithm.

6.3 COMPLETE CODING

```
#OPTIMIZATION
set val(chan)   Channel/WirelessChannel    ;# channel type
set val(prop)   Propagation/TwoRayGround   ;# radio-propagation model
set val(netif)  Phy/WirelessPhy            ;# network interface type
set val(mac)    Mac/802_11                 ;# MAC type
set val(ifq)    Queue/DropTail/PriQueue    ;# interface queue type
set val(ll)     LL                         ;# link layer type
set val(ant)    Antenna/OmniAntenna        ;# antenna model
set val(ifqlen) 50                         ;# max packet in ifq
set val(nn)     21                         ;# number of mobilenodes
set val(rp)     DSDV                       ;# routing protocol
set val(x)      861                        ;# X dimension of topography
set val(y)      594                        ;# Y dimension of topography
set val(stop)   35.0                       ;# time of simulation end
#Create a ns simulator
set ns [new Simulator]
#Setup topography object
set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)
#Open the NS trace file
set tracefile [open optimization.tr w]
$ns trace-all $tracefile
#Open the NAM trace file
set namfile [open optimization.nam w]
```

```
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
set chan [new $val(chan)];#Create wireless channel
$ns node-config -adhocRouting  $val(rp) \
        -llType       $val(ll) \
        -macType      $val(mac) \
        -ifqType      $val(ifq) \
        -ifqLen       $val(ifqlen) \
        -antType      $val(ant) \
        -propType     $val(prop) \
        -phyType      $val(netif) \
        -channel      $chan \
        -topoInstance  $topo \
        -agentTrace    ON \
        -routerTrace   ON \
        -macTrace      ON \
        -movementTrace ON
#Create 21 nodes
set n0 [$ns node]
$n0 set X_ 569
$n0 set Y_ 316
$n0 set Z_ 0.0
$ns initial_node_pos $n0 20
set n1 [$ns node]
$n1 set X_ 488
$n1 set Y_ 238
$n1 set Z_ 0.0
$ns initial_node_pos $n1 20
set n2 [$ns node]
$n2 set X_ 646
$n2 set Y_ 236
$n2 set Z_ 0.0
$ns initial_node_pos $n2 20
```

```
set n3 [$ns node]
$n3 set X_ 472
$n3 set Y_ 400
$n3 set Z_ 0.0
$ns initial_node_pos $n3 20
set n4 [$ns node]
$n4 set X_ 648
$n4 set Y_ 406
$n4 set Z_ 0.0
$ns initial_node_pos $n4 20
set n5 [$ns node]
$n5 set X_ 358
$n5 set Y_ 241
$n5 set Z_ 0.0
$ns initial_node_pos $n5 20
set n6 [$ns node]
$n6 set X_ 382
$n6 set Y_ 147
$n6 set Z_ 0.0
$ns initial_node_pos $n6 20
set n7 [$ns node]
$n7 set X_ 461
$n7 set Y_ 140
$n7 set Z_ 0.0
$ns initial_node_pos $n7 20
set n8 [$ns node]
$n8 set X_ 490
$n8 set Y_ 290
$n8 set Z_ 0.0
$ns initial_node_pos $n8 20
set n9 [$ns node]
$n9 set X_ 546
$n9 set Y_ 165
```

```
$n9 set Z_ 0.0
$ns initial_node_pos $n9 20
set n10 [$ns node]
$n10 set X_ 734
$n10 set Y_ 261
$n10 set Z_ 0.0
$ns initial_node_pos $n10 20
set n11 [$ns node]
$n11 set X_ 736
$n11 set Y_ 150
$n11 set Z_ 0.0
$ns initial_node_pos $n11 20
set n12 [$ns node]
$n12 set X_ 665
$n12 set Y_ 118
$n12 set Z_ 0.0
$ns initial_node_pos $n12 20
set n13 [$ns node]
$n13 set X_ 363
$n13 set Y_ 323
$n13 set Z_ 0.0
$ns initial_node_pos $n13 20
set n14 [$ns node]
$n14 set X_ 332
$n14 set Y_ 414
$n14 set Z_ 0.0
$ns initial_node_pos $n14 20
set n15 [$ns node]
$n15 set X_ 401
$n15 set Y_ 479
$n15 set Z_ 0.0
$ns initial_node_pos $n15 20
set n16 [$ns node]
```

```
$n16 set X_ 540
$n16 set Y_ 460
$n16 set Z_ 0.0
$ns initial_node_pos $n16 20
set n17 [$ns node]
$n17 set X_ 583
$n17 set Y_ 362
$n17 set Z_ 0.0
$ns initial_node_pos $n17 20
set n18 [$ns node]
$n18 set X_ 654
$n18 set Y_ 494
$n18 set Z_ 0.0
$ns initial_node_pos $n18 20
set n19 [$ns node]
$n19 set X_ 734
$n19 set Y_ 468
$n19 set Z_ 0.0
$ns initial_node_pos $n19 20
set n20 [$ns node]
$n20 set X_ 761
$n20 set Y_ 369
$n20 set Z_ 0.0
$ns initial_node_pos $n20 20

$ns at 1.0 "$n0 add-mark n0 green hexagon"
$ns at 1.0 "$n1 add-mark n1 #FF8C00 circle"
$ns at 1.0 "$n2 add-mark n2 #FF8C00 circle"
$ns at 1.0 "$n3 add-mark n3 brown circle"
$ns at 1.0 "$n4 add-mark n4 brown circle"
$ns at 1.0 "$n5 add-mark n5 #FF8C00 circle"
$ns at 1.0 "$n6 add-mark n6 #FF8C00 circle"
$ns at 1.0 "$n7 add-mark n7 #FF8C00 circle"
```

```
$ns at 1.0 "$n8 add-mark n8 #FF8C00 circle"
$ns at 1.0 "$n9 add-mark n9 #FF8C00 circle"
$ns at 1.0 "$n10 add-mark n10 #FF8C00 circle"
$ns at 1.0 "$n11 add-mark n11 #FF8C00 circle"
$ns at 1.0 "$n12 add-mark n12 #FF8C00 circle"
$ns at 1.0 "$n13 add-mark n13 brown circle"
$ns at 1.0 "$n14 add-mark n14 brown circle"
$ns at 1.0 "$n15 add-mark n15 brown circle"
$ns at 1.0 "$n16 add-mark n16 brown circle"
$ns at 1.0 "$n17 add-mark n17 brown circle"
$ns at 1.0 "$n18 add-mark n18 brown circle"
$ns at 1.0 "$n19 add-mark n19 brown circle"
$ns at 1.0 "$n20 add-mark n20 brown circle"
$ns at 1.0 "$n0 label MS"

$ns at 2.0 "$n5 label Source"
$ns at 2.0 "$n5 delete-mark n5"
$ns at 2.0 "$n5 add-mark n5 #8B008B  circle"
$ns at 2.0 "$n9 label Destination"
$ns at 2.0 "$n9 delete-mark n9"
$ns at 2.0 "$n9 add-mark n9 #8B008B  circle"
$ns at 4.4 "$n5 label.”
$ns at 4.4 "$n5 delete-mark n5"
$ns at 4.4 "$n5 add-mark n5 #FF8C00 circle"
$ns at 4.4 "$n9 delete-mark n9"
$ns at 4.4 "$n9 label.”
$ns at 4.4 "$n9 add-mark n9 #FF8C00  circle"
$ns at 4.4 "$n6 label Source"
$ns at 4.4 "$n6 delete-mark n6"
$ns at 4.4 "$n6 add-mark n6 #8B008B  circle"
$ns at 4.4 "$n8 label Destination"
$ns at 4.4 "$n8 delete-mark n8"
$ns at 4.4 "$n8 add-mark n8 #8B008B  circle"
```

```
$ns at 6.7 "$n6 label ."
$ns at 6.7 "$n6 delete-mark n6"
$ns at 6.7 "$n6 add-mark n6 #FF8C00 circle"
$ns at 6.7 "$n8 delete-mark n8"
$ns at 6.7 "$n8 label ."
$ns at 6.7 "$n8 add-mark n8 #FF8C00  circle"
$ns at 6.7 "$n1 delete-mark n1"
$ns at 6.7 "$n1 label CH"
$ns at 6.7 "$n1 add-mark n1 red square"
$ns at 6.7 "$n9 label Source"
$ns at 6.7 "$n9 delete-mark n9"
$ns at 6.7 "$n9 add-mark n9 #8B008B  circle"
$ns at 6.7 "$n10 label Destination"
$ns at 6.7 "$n10 delete-mark n10"
$ns at 6.7 "$n10 add-mark n10 #8B008B  circle"
$ns at 9.0 "$n9 label ."
$ns at 9.0 "$n9 delete-mark n9"
$ns at 9.0 "$n9 add-mark n9 #FF8C00 circle"
$ns at 9.0 "$n10 delete-mark n10"
$ns at 9.0 "$n10 label ."
$ns at 9.0 "$n10 add-mark n10 #FF8C00  circle"
$ns at 9.0 "$n8 label Source"
$ns at 9.0 "$n8 delete-mark n8"
$ns at 9.0 "$n8 add-mark n8 #8B008B  circle"
$ns at 9.0 "$n11 label Destination"
$ns at 9.0 "$n11 delete-mark n11"
$ns at 9.0 "$n11 add-mark n11 #8B008B  circle"
$ns at 11.3 "$n8 label ."
$ns at 11.3 "$n8 delete-mark n8"
$ns at 11.3 "$n8 add-mark n8 #FF8C00 circle"
$ns at 11.3 "$n11 delete-mark n11"
$ns at 11.3 "$n11 label ."
$ns at 11.3 "$n11 add-mark n11 #FF8C00  circle"
```

```
$ns at 11.4 "$n2 delete-mark n2"
$ns at 11.4 "$n2 label CH"
$ns at 11.4 "$n2 add-mark n2 red square"

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n5 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n1 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 2.1 "$ftp1 start"
$ns at 2.9 "$ftp1 stop"

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n9 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 3.1 "$ftp1 start"
$ns at 4.0 "$ftp1 stop"

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n6 $tcp1
```

```
set sink1 [new Agent/TCPSink]
$ns attach-agent $n1 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 4.4 "$ftp1 start"
$ns at 5.3 "$ftp1 stop"

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n8 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 5.4 "$ftp1 start"
$ns at 6.3 "$ftp1 stop"

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n9 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n2 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
```

```
$ns at 6.7 "$ftp1 start"
$ns at 7.6 "$ftp1 stop"

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n2 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n10 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 7.7 "$ftp1 start"
$ns at 8.6 "$ftp1 stop"

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n8 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n2 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 9.0 "$ftp1 start"
$ns at 9.9 "$ftp1 stop"

#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n2 $tcp1
set sink1 [new Agent/TCPSink]
```

```
$ns attach-agent $n11 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 10.1 "$ftp1 start"
$ns at 11.0 "$ftp1 stop"

## cluster communication
#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n6 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n1 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 20.5 "$ftp1 start"
$ns at 21.4 "$ftp1 stop"
#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n0 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 21.5 "$ftp1 start"
```

```
$ns at 22.4 "$ftp1 stop"
#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n0 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n2 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 22.6 "$ftp1 start"
$ns at 23.5 "$ftp1 stop"
#Setup a TCP connection
set tcp1 [new Agent/TCP]
$ns attach-agent $n2 $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n11 $sink1
$ns connect $tcp1 $sink1
$tcp1 set packetSize_ 1500
#Setup a FTP Application over TCP connection
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 23.7 "$ftp1 start"
$ns at 24.6 "$ftp1 stop"
$ns at 1.0 "$ns trace-annotate \"Network Created.....\""
$ns at 2.1 "$ns trace-annotate \"Node Communication\""
$ns at 6.7 "$ns trace-annotate \"CH selection\""
$ns at 20.4 "$ns trace-annotate \"MS and CH based Communication\""
proc finish {} {
   global ns tracefile namfile
   $ns flush-trace
   close $tracefile
```

```
   close $namfile
   exec nam optimization.nam &
exec awk -f performance.awk optimization.tr > graph.xgr &
   exec awk -f through.awk optimization.tr > throughput.xgr &
   exec xgraph Throughput &
   exec xgraph delay &
   exec xgraph pdr &
   exec xgraph energygh &
   exit 0
}
for {set i 0} {$i < $val(nn) } { incr i } {
   $ns at $val(stop) "\$n$i reset"
}
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "finish"
$ns at $val(stop) "puts \"done\" ; $ns halt"
$ns run
```

6.4 SNAP SHOTS

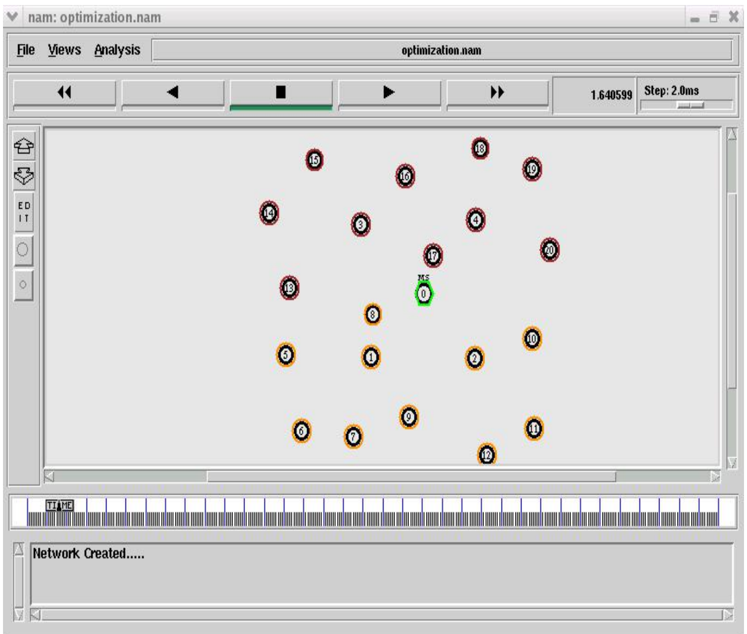


Fig: 6.7 - NETWORK FORMATION

This diagram explains how the network is formed. The network consists of nodes and a Mobile Sink. There are two clusters. Each cluster will be separated using different colors.

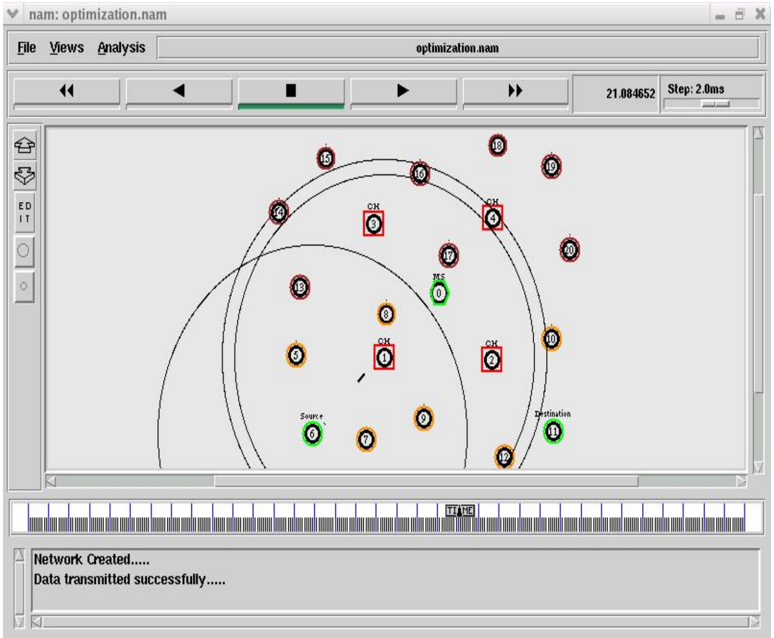


Fig : 6.8 - SOURCE TO CLUSTER HEAD

This diagram shows the data communication between the source and the CH. The CH is shown in red color. The source will transmit the data to the CH using data packets.

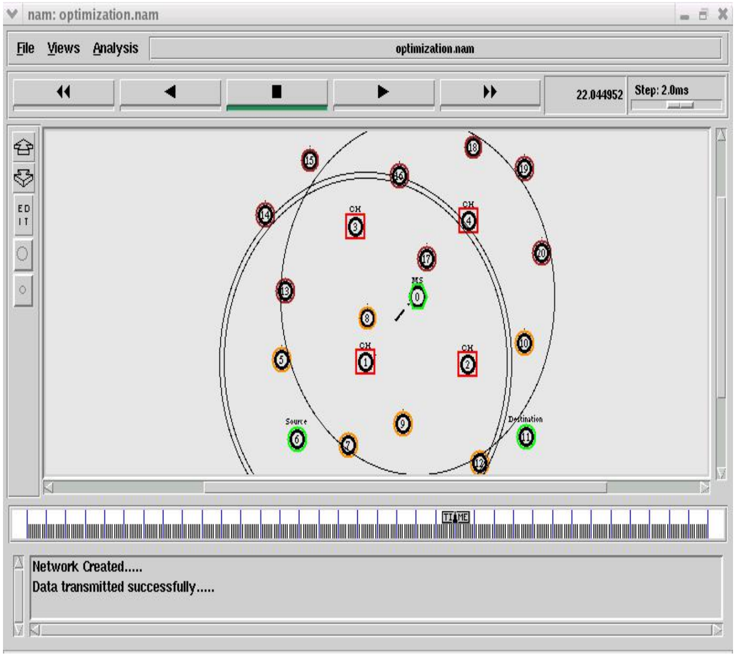


Fig: 6.9 - CLUSTER HEAD TO MOBILE SINK

This diagram shows the data communication between the CH and the MS. The CH is shown in red color and the MS in green color.

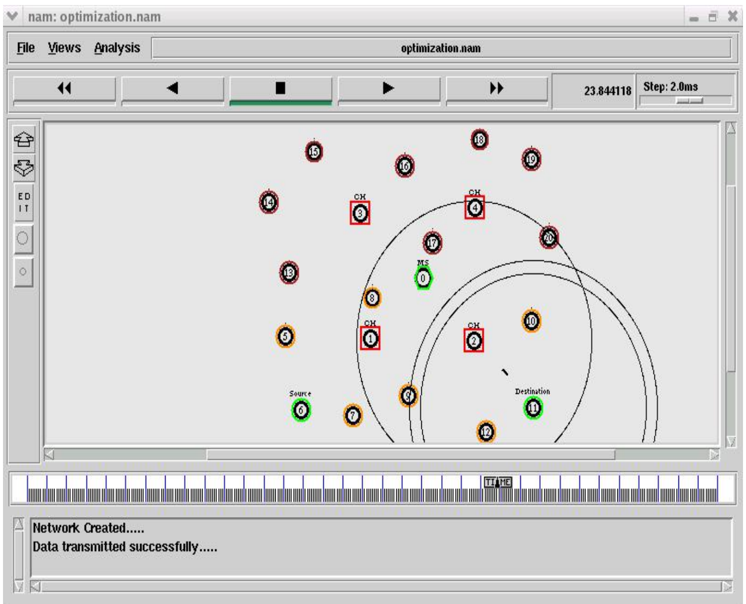


Fig: 6.10 - CLUSTER HEAD TO DESTINATION

The CH will transmit the data collected from the other nodes to the MS. This diagram shows the data communication between the MS, CH and the destination. The CH is shown in red color and both destination and MS in green color. The MS will transmit the received to the CH. The CH will transmit the data to the respective destination.

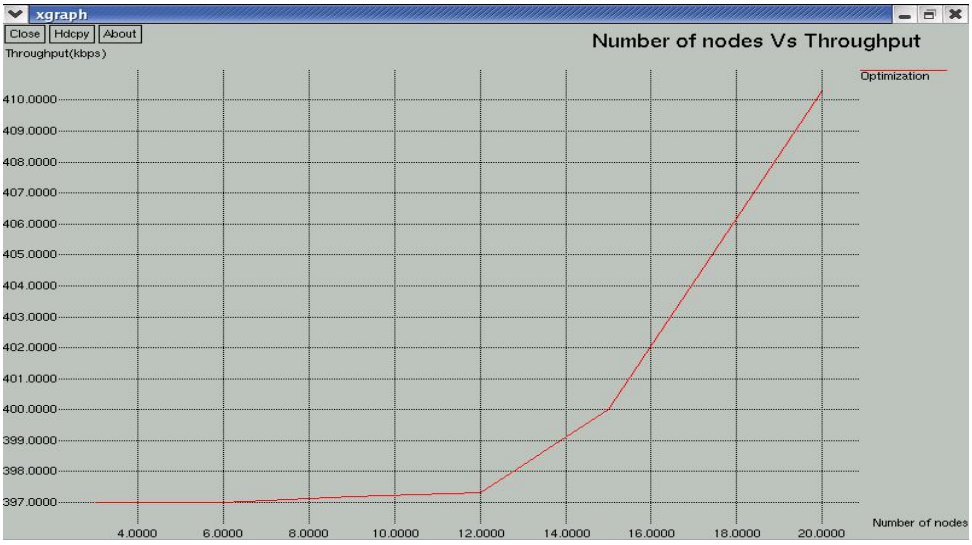


Fig: 6.11 - THROUGHPUT

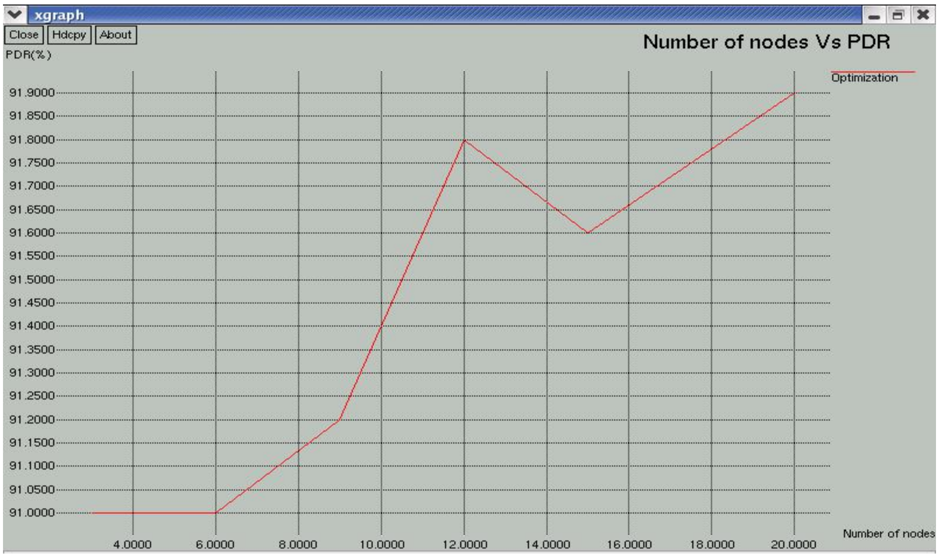


Fig: 6.12 - PACKET DELIVERY RATIO

The packet delivery ratio is the ratio of packets successfully received to the total sent.

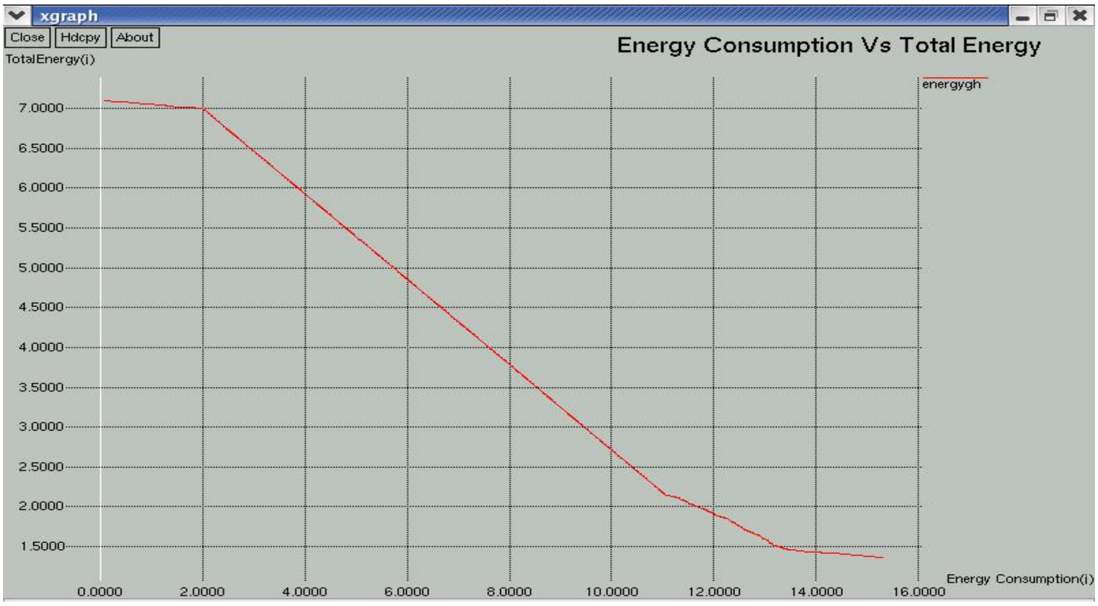


Fig: 6.13 - ENERGY CONSUMPTION

The energy consumption level of a node at any time of the simulation can be determined by finding the difference between the current energy value and initial energy value

CHAPTER 7

TESTING

7.1 Unit Testing

Unit testing is a level of software testing where individual units/ components of software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output.

7.1.2. Integration Testing

Integration testing is a logical extension of unit testing. Integration testing is defined as the testing of combined parts of an application to determine if they function correctly. Integration testing identifies problems that occur when units are combined. By using a test plan that requires you to test each unit and ensure the viability of each before combining units, you know that any errors discovered when combining units are likely related to the interface between units. This method reduces the number of possibilities to a far simpler level of analysis.

7.1.3. Validation Testing

Validation testing is to determine to check whether an implemented system fulfills its requirements. The checking of data for correctness or compliance with applicable standards, rules, and conventions from testing perspective is carried out. The code is checked for the possible faults and failures and rectified. Thus validation testing is checked successfully.

7.1.4. System Testing

System testing tests the system as a whole. Once all the components are integrated, the application as a whole is tested rigorously to see that it meets the specified Quality

Standards. System testing is the first step in the Software Development Life Cycle, where the application is tested as a whole. The application is tested thoroughly to verify that it meets the functional and technical specifications. System testing enables us to test, verify, and validate. It falls within the scope of black-box testing.

CHAPTER 8

IMPLEMENTATION

The implementation stage involves the following tasks.

- Careful planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.
- Training of the user in the changeover phase.
- Evaluation of the changeover method.

8.1 PROBLEMS FACED

During the system development, the problems faced are given below:

1. VM ware installation took a long time.

2. Performance metrics are different than the original one.

8.2 LESSONS LEARNT

While developing this project, we came across many experiences. They are

1. Before starting the project, we must have a proper plan for it.

2. Unit testing is very important as bugs are identified then and there itself.

3. We should not jump into code directly. First, the project should be analyzed thoroughly.

4. Larger coding must be split into much smaller coding so that it may be much efficient.

CHAPTER 9

CONCLUSION AND FUTURE WORK

9.1 CONCLUSION

The proposed methodology compares the two optimization algorithms viz., PSO and GA to find the CH for data transmission. It also indicates execution change in quality contrasted with the irregular organization and the current strategies. The point is to reason an ideal answer for hubs organization ensuring the accompanying targets: augmenting the scope zone, boosting the accuracy sound limitation at the level of the location flag. As a prospect of our work, we intend to streamline the proposed calculation keeping in mind the end goal to guarantee the redeployment issue while upgrading diverse targets other than the scope and the confinement, for example, the lifetime and the system network.

9.2 FUTURE WORK

Future work might include the addition of multiple communications between cluster heads to solve the problem of simultaneous sending and receiving data.

REFERENCES

[1] Maraiya K, Kant K, Gupta N. Efficient Cluster Head Selection Scheme for Data Aggregation in Wireless Sensor Network. International Journal of Computer Applications, vol. 23, no. 9, June 2011.

[2] Hussain S, Matin AW, Islam O. Genetic algorithm for energy efficient clusters in wireless sensor networks. in Fourth International Conference on Information Technology: New Generations (ITNG 2007), April 2007.

[3] W. Liang, J. Luo, X. Xu, Prolonging network lifetime via a controlled mobile sink in wireless sensor networks, in: Proc. of IEEE Global Telecommunications Conference, 2010, pp. 1–6.

[4] K. Tian, B. Zhang, K. Huang, J. Ma, Data gathering protocols for wireless sensor networks with mobile sinks, in: Proc. of IEEE Global Telecommunications Conference, 2010, pp. 7–12.

[5] H. Salarian, K.W. Chin, F. Naghdy, An energy-efficient mobile-sink path selection strategy for wireless sensor networks, IEEE Trans. Veh. Technol. 63 (5) (2014) 2407–2419.

[6] Z. Lin, H. Zhang, Y. Wang, F. Yao, Energy-efficient routing protocol on mobile sink in wireless sensor network, Adv. Mater. Res. 787 (2013) 1050–1055.

[7] Y. Gu, Y. Ji, J. Li, B. Zhao, ESWC: Efficient scheduling for the mobile sink in wireless sensor networks with delay constraint, IEEE Trans. Parallel Distrib. Syst. 24 (7) (2013) 1310–1320.

[8] C. Tunca, S. Isik, M.Y. Donmez, C. Ersoy, Ring routing: An energy-efficient routing protocol for wireless sensor networks with a mobile sink, IEEE Trans. Mob. Comput. (2014) 1–14.

[9] S. M. Hosseinirad, S. K. Basu, (2012). Imperialist Approach to Cluster Head Selection in WSN, International Journal of Computer Applications, 1-5.

[10] Moslem Afrashteh Mehr. (2011). Design and Implementation a New Energy Efficient Clustering Algorithm using Genetic Algorithm for Wireless Sensor Networks, World Academy of Science, Engineering and Technology, 430-433.

[11] Omar Banimelhem, Moad Mowafi, Eyad Taqieddin, Fahed Awad, Manar Al Rawabdeh, "An Efficient Clustering Approach using Genetic Algorithm and Node Mobility in Wireless Sensor Networks", 11th International Symposium on Wireless Communications Systems (ISWCS), 2014, pp: 858 – 862.

[12] Ying Liang and Haibin Yu, "PSO-Based Energy Efficient Gathering in Sensor Networks", International Conference on Mobile Ad-Hoc and Sensor Networks,2005, pp 362-369.

[13] Akyildiz IF, W. Su, Y. Sankarasubramaniam, Cayirci E. Wireless sensor networks: A survey. Computer Networks, vol. 38, no. 4, pp. 393–422, March 2002.

[14] Bandyopadhyay S, Coyle E. J. An energy efficient hierarchical clustering algorithm for wireless sensor networks. In Proceedings of the IEEE Conference on Computer Communications (INFOCOM), 2003.

[15] Estrin D, Culler D, Pister K, Sukhatme G. Connecting the physical world with pervasive networks. IEEE Pervasive Computing, pages 59 – 69, January-March 2002.

[16] Heinzelman W. R, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocol for wireless microsensor networks. In Proceedings of the Hawaii International Conference on System Sciences, January 2000.

[17] Heinzelman W, Chandrakasan A.P, Balakrishnan H. An Application-Specific Protocol Architecture for Wireless Microsensor Networks. IEEE Transaction on Wireless Communications, Vol. 1, No. 4, Oct. 2002.

[18] Hussain S, Matin AW. Base station assisted hierarchical cluster-based routing. in IEEE/ACM International Conference on Wireless and Mobile communications Networks(ICWMC), July 2006.

[19] Hussain S, Matin AW, Islam O. Genetic algorithm for energy efficient clusters in wireless sensor networks. in Fourth International Conference on Information Technology: New Generations (ITNG 2007), April 2007.

[20] D. Goldberg, B. Karp, Y. Ke, S. Nath, and S. Seshan, Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, 1989.

[21] Maraiya K, Kant K, Gupta N. Efficient Cluster Head Selection Scheme for Data Aggregation in Wireless Sensor Network. International Journal of Computer Applications, vol. 23, no. 9, June 2011.

<u>Paper published</u>

Saravanan Palani., Yarlagadda Venkata Subba Rao., Kanampalli Sai Kiran Reddy., "Genetic Algorithm based Cluster Head Selection for Optimimized Communication in Wireless Sensor Network", International journal of Pure and Applied Mathematics. (In Review).