PERFORMANCE ANALYSIS AND OPTIMIZATION IN ROUTING FOR MANETS

THESIS

submitted in fulfilment of the requirement of the degree of

DOCTOR OF PHILOSOPHY

to

J.C. BOSE UNIVERSITY OF SCIENCE & TECHNOLOGY, YMCA FARIDABAD

by

PRASHANT DIXIT

Registration No: PhD-06-2K12

under the supervision of

Dr. ANURADHA
ASSISTANT PROFESSOR

Prof. RAHUL RISHI
DIRECTOR, UIET, MDU

Department of Computer Engineering
Faculty of Engineering and Technology
J.C. Bose University of Science & Technology, YMCA, Faridabad
Sector-6, Mathura Road, Faridabad, Haryana, India
February 2020

Dedicated to the protagonist of my life

*“**Ashu Bhaiya**”*

DECLARATION

I hereby declare that this thesis entitled **“PERFORMANCE ANALYSIS AND OPTIMIZATION IN ROUTING FOR MANETs”** by **PRASHANT DIXIT**, being submitted in fulfilment of requirement for the award of Degree of Doctor of Philosophy in the Department of Computer Engineering under Faculty of Engineering and Technology of J.C. Bose University of Science and Technology, YMCA, Faridabad, during the academic year 2019-20, is a bonafide record of my original work carried out under the guidance and supervision of **Dr. Anuradha, Assistant Professor, Department of Computer Engineering, J.C. Bose University of Science and Technology, YMCA, Faridabad** and **Prof. Rahul Rishi, Director, UIET, MDU, Rohtak, Haryana** has not been presented elsewhere.

I further declare that the thesis does not contain any part of any work which has been submitted for the award of any degree either in this university or in any other university.

PRASHANT DIXIT
PhD-06-2K12

CERTIFICATE

This is to certify that this thesis entitled **"PERFORMANCE ANALYSIS AND OPTIMIZATION IN ROUTING FOR MANETs"** by **PRASHANT DIXIT,** being submitted in fulfilment of the requirements for the Degree of Doctor of Philosophy in Department of Computer Engineering, under faculty of Engineering and Technology of J.C. Bose University of Science and Technology, YMCA, Faridabad, during the academic year 2019-20, is a bonafide record of work carried out under our guidance and supervision.

We further declare that to the best of our knowledge the thesis does not contain any part of any work which has been submitted for the award of any degree either in this university or in any other university.

More than two research papers in prestigious refereed journals have been published and all other academic requirements are fulfilled.

The thesis is based on the individual, original work of the candidate, which is previously unpublished research work and the thesis does not contain any material that infringes the copyright of any other individual or organization and does not hurt the sentiments of any individual(s) or religion(s). The information such as text, tables, equations, diagrams, figures, charts, and photographs taken from sources such as published work, like research papers, books, periodicals, web sites and other resources has been cited appropriately. Further, the opinions expressed or implied in the thesis are entirely of the candidate.

Dr. ANURADHA
Assistant Professor
J.C. Bose University of Science &
Technology, YMCA, Faridabad

Prof. RAHUL RISHI
Director, UIET, MDU,
Rohtak, Haryana

ACKNOWLEDGEMENT

I express my gratitude to my ***Guruji*** *"****Devrshi Yogtirth ji Maharaj****" and "****Almighty God****" for blessing me with inner strength to carry out this research.*

I wish to express my sincere thanks and gratitude to my supervisors **Dr Anuradha**, Assistant Professor, Department of Computer Engineering, J.C. Bose University of Science and Technology, YMCA, Faridabad and **Prof. Rahul Rishi**, Director, UIET, MDU, Rohtak, for their mentoring and guidance during this research work. Their profound knowledge, generous guidance, timely advice, insightful criticisms and encouragements made it possible for me to carry out the research work successfully.

I would like to concede my special thanks to Prof. Komal Bhatia, Dean and Chairman, Department of Computer Engineering, J.C. Bose University of Science and Technology, YMCA, for his constant support and encouragement over the period of my research work. I extend my sincere thanks to Prof. Atul Mishra, Prof. C.K. Nagpal, Prof. Manjeet Tomar and Prof. Naresh Chauhan for their critical comments and valuable suggestions rendered throughout my research work.

I would like to thank from the bottom of my heart to my colleague Sh Amit Chug for helping me to understand the MATLAB. I express my profound gratitude to Dr Urvashi Chug for her motivation that tows me out from the stress of PhD. I am very thankful to Dr Madhulika for her strict observations on my research paper writing.

I am immensely indebted to my parents Smt Shobha and Sh D.N. Dixit and my elder brother Prof. Ashutosh Dixit for blossoming me. Without their blessings I could not have been what I am today. I also express my gratitude to my sweet sisters Smt Poonam Tiwari and Dr Sandhya Dixit for their blessings and moral support. I convey my gratitude and love to my dear wife Dr Shefali for inspiring me, I am very blessed to have her near me. She constantly cheered me when I was down and helped me when I needed advice and aid on my research. I express my love to my son Madhav, I also want to say sorry to him because the priceless time that I should have spent with him, I have given to the research. I express my gratitude to "Swami Yogvrat ji" for his spiritual guidance to keep my mind peaceful and creative. I take this opportunity to thank all my friends and well-wishers who helped me directly or indirectly during this research work.

ABSTRACT

Over the last few decades, the market of wireless communication has experienced an extraordinary growth. The capability of wireless technology has made it possible to reach logically every location on the earth. Billions of individuals exchange massive amount of data every single day using cellular phones, laptops, and countless types of personal digital assistants (PDAs) and other kind of wireless communication devices, along with the remarkable success of wireless voice, video and messaging services. It is not surprising that wireless technology is applied to the dominion of ubiquitous computing. In near future, people will be able to access and exchange the information at worldwide scale in 'any-time-any-where' manner without the dependency on the wired or infrastructure based network. Mobile ad-hoc network comes with these necessary characteristics in which mobile terminals form temporary network without the necessity of any preexisting infrastructure. To formalize this statement a Mobile ad-hoc network can be defined as a self-governing system of mobile terminals (MTs) connected by wireless links. The unification of which forms a communication network exhibited in the form of a random communication graph. In this manner mobile ad-hoc network is in contrast to typical single hop cellular network that require the installation of base station as access point for wireless communication. In the cellular networks, communication between two mobile terminals is totally dependent on the wired backbone and the fixed base station, whereas, in case of mobile ad-hoc network, no infrastructure preexists and since the mobile terminals are free to move throughout the network, the topology changes dynamically in random manner. Due to its extemporized nature, the application of the mobile ad-hoc network is in the situations where the establishment of infrastructure is not possible due to the geographic or time constraints. Such situations include battle filed, rescue operation and disaster management etc. Apart from these, the potential application of a mobile ad-hoc network is in interactive conference, group of people in limited geographic area where individuals may share the real-time data like voice and video.

With inherent advantages of mobile ad-hoc network, the flip side of coin is considerable complexity, control overhead in routing of packets due to the random movement of the mobile terminals and arbitrary topology. Classic bellmen ford algorithm used in wired

networks is unsuitable to find path between source and destination in ad-hoc scenario as information packet forwarding decision is taken at real-time due to its dynamic topology. On account of this consideration, *Quality of Service and scalability in* mobile ad-hoc network are two major issues to grip. *Quality of Service* ensures the reliable and consistent communication between the communicating terminals and *scalability* ensures the effectual functionality of a mobile ad-hoc network with increasing number of mobile terminals.

Keeping these issues in mind, structural design of mobile ad hoc network has been studied comprehensively in this research work. *Location aided routing* and *cluster-based routing algorithms* have inherent advantages over the other class of algorithms. Mostly, all location-based routing algorithms focuses on minimising the number of participating nodes in path construction but overlook the Quality of Service. To harness the location aided routing with Quality of Service, a novel algorithm "*QoS Enabled Improved Location Aided routing(QEILA)*" has been proposed and developed that utilizes the improved location aided routing protocol (ILAR) and has been equipped with Quality of Service check while selecting the next forwarding node for path construction. In QEILA, a novel path preservation procedure has been followed which repairs the broken link locally by utilizing "candidate next node table".

To improve the efficiency of existing approach, another routing algorithm "*Location Information Based Destination Converging Routing Method (LIBDCR)*" has been proposed and developed, in which path discovery process move in forward direction only at each successive step to converge at destination. This research work moved forward with carrying in mind the 'scalability' issue of ad-hoc networks. Cluster based routing system has been studied carefully and it has been observed that in almost all cluster based routing systems, cluster-head becomes overburdened due to the additional responsibilities. Therefore, a novel light weight cluster based routing approach "*A Light Weight Efficient Cluster based routing Model for Mobile Ad-hoc Networks (LWECM)"* has been proposed and demonstrated. In LWECM, cluster-head is spared from unnecessary involvement in communication events which may be possible without it. Therefore, the life of cluster-head gets prolonged and cost of re-clustering due to the expiration of cluster-head has been minimized. It has also been observed that "*Weighted Clustering Algorithms*" proved better over the other clustering algorithms, but induce the huge control overhead in election process prior to the actual routings with high maintenance cost. To address these issues, a novel clustering framework "*Marking based Load balancing Weighted Clustering framework for Mobile AD-HOC Network (MLWCM)*" has been proposed and developed. MLWCM marks the weak nodes prior to the beginning of election algorithm and spare them from massive weight calculation and saves the network resources. The concept of inmate affiliation reduces the maintenance cost considerably. MLWCM also ensures fair load distribution among the cluster-heads.

All developed algorithms and corresponding existing algorithms have been simulated and compared on MATLAB 7.8.0. The findings show that developed algorithms outperform over the existing algorithms on account of reliability, Quality of Service and control overheads.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

MANET	Mobile Ad-hoc Network
OSPF	Open Shortest Path First
RIP	Routing Information Protocol
PRNet	Packet Radio Network
PAN	Personal Area Network
WAN	Wide Area Network
MAC	Medium Access Control
LAN	Local Area Network
DSDV	Destination-Sequenced Distance-Vector Protocol
WRP	Wireless Routing Protocol
LCT	Link-cost table
DT	Distance table
RT	Routing Table
MRL	Message Retransmission List table
LCT	The Link-cost table
STAR	Source-Tree Adaptive Routing Protocol
OLSR	Optimized Link State Routing Protocol
MPR	Multi-point relay
DSR	Dynamic Source Routing protocol
AODV	Ad-hoc On-Demand Distance Vector routing protocol
TORA	Temporally Ordered Routing Algorithm
IMEP	Internet MANET Encapsulation Protocol
CBRP	Cluster Based Routing Protocol
k-CONID	Connectivity based k-hop clustering in wireless networks
MobDhop	Mobility-based d-hop clustering algorithm for mobile ad hoc networks
LCC	least cluster change
3hBAC	3-hop between adjacent cluster-head
MFR	Most Forward within r
NFP	Nearest within Forwarding Progress

GPSR	Greedy perimeter Stateless Routing Protocol
LAR	Location aided routing
DREAM	Distance Routing Effect Algorithm for Mobility
LMANET	Large scale Mobile Ad-Hoc Network
HCAL	Hierarchical Clustering Algorithm
IPS	Instant Processing State
PDA	Personal Digital Assistant
QEILA	QoS Enabled Improved Location Aided routing
LIBDCR	Location Information Based Destination Converging Routing Method
LWECM	Light Weight Efficient Cluster based routing Model for Mobile Ad-hoc Networks
MLWCM	Marking based Load balancing Weighted Clustering framework for Mobile ad-hoc Network
CNN	Candidate Next Node
ABL	Available battery life
RBL	Remaining Battery Life
RERQ	Route request
R_RREQ	Revise route request
QoS	Quality of Service
ABW	Available Bandwidth
RB	Required Bandwidth
RREP	Route reply
TTL	Time to live
ILAR	Improved Location Aided Routing
LARDAR	Location-aware routing protocol with dynamic adaptation of request zone for mobile ad hoc networks
LBPAR	Location Based Power Aware Routing in MANET
NFN	Next Forwarding Node table
CH	Cluster Head
CF	Competence Factor
REQ_AFFI	Request for Affiliation
REPL_AFFI	Reply For Affiliation

CHAPTER 1

INTRODUCTION

1.1 GENERAL

In the past few decades, the worldwide availability of portable wireless transmission machines and evolving ubiquitous computing requires “anytime and anywhere" network connections which encouraged research on autonomous wireless networks that do not require pre-existing infrastructure like base stations. These self-organizing networks are called mobile ad-hoc networks. Mobile ad-hoc networks were invented from the DARPA Packet Radio Network (PRNet) [1] and SURAN project [2]. A mobile ad-hoc Network (MANET) is a collection of mobile nodes which cooperates each other without any fixed infrastructure for transmission of data from one node to the other node in network. Due to the limited transmission range, some terminals are unable to connect directly to each other. Therefore in MANET, path from one node to another may contain multiple hops, hence each node in the network needs to play the role of router, apart from sender or receiver. Example of an ad-hoc network is Bluetooth piconet. Typical applications of Mobile ad-hoc network are in military, battle field, disaster management, rescue operations where the infrastructure setup is not feasible. Apart from above mentioned situations, some other possible applications of Mobile Ad-hoc network are in an interactive conference and lecture or social network in a limited geographic region, where the participants with mobile devices like laptop, mobile phone, PDA need to share heavy data like videos, images, and voice in real time scenario. Structure of a Mobile ad-hoc network is shown in Fig. 1.1

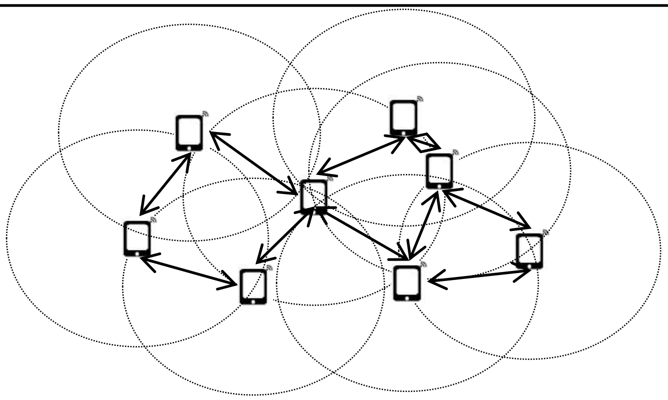


Figure 1.1: Structure of an ad-hoc network

Current research with respect to mobile ad-hoc network is booming in the field of routing, resource management, medium access control and power control. Due to the significance of routing protocols in the movable multi hop network, numerous routing strategies for mobile ad-hoc network have been proposed in last few decades.

While designing a routing protocol for mobile ad-hoc network, researchers face many challenges. Primary challenge in MANET is dynamic topology due to the node mobility. Secondly, due to the dynamic and unpredictable capacity of wireless link, packet loss occurs frequently. Apart from that, hidden terminal and exposed terminal problems may arise due to the broadcast nature of wireless medium. Moreover, mobile terminals have limited resources like bandwidth, computation and battery power that requires efficient routing strategy. Characteristics of mobile ad-hoc network has been comprehended and tabularized in Table 1.1.

Table 1.1 Characteristics of an ad-hoc network

CHARACTERISTIC	**DETAILS**
Limited energy	Mobile nodes in the network are driven by the portable and exhaustible battery. Therefore the power consumption is the major consideration while designing the ad-hoc network
Varying topology	As the mobile nodes are free to move throughout the network the topology of network remains dynamic and unpredictable.
Limited Bandwidth	In comparison to wired links the wireless link has lower capacity of data transmission, because of the wireless side effects like multiple access, fading, noise, narrow band interference and hidden or exposed terminals.
Security	Wireless medium is more vulnerable than fixed-cable for security threat. The possibility of eavesdropping, spoofing etc. is more.

Constraints of classic routing mechanism used in wired network, in case of wireless ad-hoc network are discussed in next section.

1.2 CONSTRAINT OF TRADITIONAL ROUTING APPROACHES USED IN WIRED NETWORKS

Routing of packet is fundamental need of a network. For wired networks, numerous routing protocols have been devised and some of them are commonly used. Typical examples of wired network routing protocols are Distance Vector (also called Bellman-Ford routing algorithm) [3] and Link State Routing protocol [3].

In Distance Vector routing, every router maintains a routing table (i.e. vector), that contains all reachable destination nodes along with their respective distances. All the routers exchange the routing information periodically with their nearby nodes to refresh the routing information. Distance to a node generally measured in terms of hop count. If more than one path exists, the shortest path is chosen. The disadvantage of Distance Vector routing algorithm is count-to-infinity problem in which some routers continuously increase the hop count for a particular node due to looping. Open Shortest Path First (OSPF) [3] is a classic protocol which is based on Distance Vector algorithm.

In Link State routing algorithm, every terminal floods its link information to all the routers in the network. On receiving the link information, all routers recalculate the route accordingly. In this way routers remain aware about the topology changes. For link state, different parameters can be selected such as hop count, data rate and packet congestion. Dijkstra’s algorithm is then used to calculate the shortest path. Routing Information Protocol (RIP) [4] is a good example of Link State routing algorithm.

Distance vector routing and link state routing are suitable for wired network because of predictable network characteristics like stable network topology and link quality. On the other hand, the dynamic nature of Mobile ad-hoc network makes standard wired routing inappropriate for it. Due to the changing topology, use of Distance vector or Link state routing in mobile ad-hoc network escalate the control overhead considerably and consumes the scarce network resources like bandwidth and battery power. In addition Wired routing algorithms may suffers from looping of packet due the continuous changing topology of an ad-hoc network.

1.3 APPLICATIONS OF AD-HOC NETWORK

There are several potential applications of MANETs. As a matter of fact any application in daily routine such as e-mail, file transfer, voice mail and video calling etc. can be considered to be easily deployable in a mobile ad-hoc network scenario. The other evident examples of mobile ad-hoc network is wide range of military applications, such as battlefield in an indefinite region where an infrastructure based network is nearly impossible to establish due to geographical and time limitations. In such circumstances, ad-hoc network having self-governing capabilities can competently be used effectively. Readiness of advanced features of a wireless mobile device, such as data rate regulation with multimedia applications, global roaming capacity and harmonization with other network structures, are inviting new applications. Following are the well-known applications of mobile ad-hoc network are:

- **Cooperative Work**

Some business structures require the cooperative computing outside the office, more than inside a building. As there are the cases where team members do need to have outdoor meetings to collaborate and exchange information for a specified project. e.g. any civil project.

- **Disaster management applications**

Another potential application of mobile ad-hoc network is in case of catastrophe. Natural disaster like tsunamis, hurricanes and earth quake where the entire communication infrastructure gets collapsed and quickly reestablishing the communication is vital. Mobile ad-hoc network allows the communication to be setup in hours in comparison of wired network which requires weeks.

- **Personal Area Networking**

Personal area network is personalized short-proximity network formed by the mobile devices that belong to a single person. These devices could be paired with person’s mobile phone, pulse watch, belt etc. Example of personal area network (PAN) can be seen in a Bluetooth head phone or printer, paired with person's mobile phone or laptop.

1.4 AD-HOC NETWORK: DESIGN ISSUES

Ad-hoc networking has become a prevalent field among the researchers in last few decades. In this respect nearly every characteristic and design challenges of ad-hoc network has been explored in some level of understanding. But no ideal resolution of these challenges are found yet or at least agreed upon. Rather many questions have been unanswered and various issues remain to be addressed. Inherent design issues for ad-hoc network that ought to be resolved has been discussed as follows.

(i) Scalability

Most of the intellectuals showing application which utilizes the ad-hoc network technology take the scalability as granted. As an example, ubiquitous computing where network is anticipated to be of any size. In reality, it is unclear so far that how much such network can actually proliferate. Characteristically ad-hoc network experience the problem of scalability i.e. number of nodes that it can accommodate. To support this statement interface studies reveals that in a wireless network where omnidirectional antenna were used, throughput at every node drops at the rate of $1/\sqrt{N}$ where N is the number of nodes in the network [7]. That means for a network with 100 nodes, single node have at max nearly 1/10 of the total network data transfer rate. This problem can be resolved by improving the physical layer having directional antennas. As there is limited bandwidth for communication, control packets exchange like route request, location beacons and encryption keys are the few examples which impose the considerable amount of control overhead with the increase of network size. Therefore scalability is the most critical issue that needs to be addressed while designing the protocols for ad-hoc network.

(ii) Quality of Service

Due to the heterogeneous nature of existing internet applications, network designer manages to provide only the best effort delivery. Video conferencing, voice chat and file transfer are some examples that have different requirements. Quality of Service aware protocols are being proposed and developed in this research to fulfill the evolving requirements of these kinds of applications. Quality of Service ensures the definite performance of network for a given transmission in terms of QoS parameters like certain bandwidth, power, delay etc. QoS enabled routing makes effort to select path that meets the given performance requirement and reserves the resources for uninterrupted data transmission. In spite of the effort done by the researchers in area of Quality of Service, QoS in ad-hoc network is still an unrevealed area.

(iii) Unsuitability of client-server model

In the classical internet model, a network client takes the service of its associated server for network transaction. The sever can be located dynamically or statistically. Conversely in ad-hoc network, every node exists on the same level (software and hardware capability) and it is hard to collect the IP addresses into subnet. There cannot be a specific server for name resolution and services like authentication as the topology is changing over the time. Due to the absence of infrastructure and presence of mobility of nodes, a different addressing approach is required. Moreover it is hard to determine who will be responsible for performing various network services such as cluster head, location server etc.

(iv) Privacy and security

Ad-hoc networks mostly exist in open wireless environment which makes them prone to malicious attacks. Absence of any centralized network controller or authentication service makes this dynamic network more vulnerable to infiltration, interference etc. Security is one of the major “barrier” for the commercialization of ad-hoc network technology. Therefore security is indeed the most challenging task to enforce in the ad-hoc network. The zenith age of ad-hoc network research can be expected only after the better security system is explored.

(v) Inter functioning of ad-hoc network with WAN

It is very unlikely that any application of ad-hoc network does not require connection with internet. Therefore defining the interoperability among inherently two different networks is a very challenging task. It can only be achieved by a single node with internet connection that could offer the internet connectivity to the other nodes. In this case, node that provides the internet connectivity, defines itself as default router and the other nodes are logically at single hop for internet connectivity although physically they are multi hop. This can be achieved by the fusion of mobile IP technology [8] and ad-hoc networking technology.

(vi) Power consumption

Power consumption has a great concern in mobile ad-hoc networking technology because every node in the network relies on a small battery. In the present time, researchers are concerned about the power consumption at every layer of the protocol stack. There are mainly two concerns of research regarding power consumption: maximization of lifetime of a single node and maximization of lifetime of whole network. Maximization of single node is related to commercialization of technology, whereas the maximization of lifetime of whole network is more vital. For example, consider a battlefield where every soldier have a battery driven mobile device and cooperate with each other. Life of network can be prolonged, either by providing heavy battery (which is not portable) or by making control functioning more energy efficient. For a single device power energy consumption can be optimized by evolving the low power hardware using the advance technique like variable clock speed CPUs, flash memory, disk spindown [9]. Therefore keeping the ad-hoc networking in the mind, researcher's interest is more on the devices network interface which is the main source of energy consumption. At the network interface, power consumption can be minimized by improving the transmission/ reception technology on the physical layer (like antenna) with efficient algorithms. Research is going on for power saving at physical layer, medium access control layer, network layer, and on transport layer. Nevertheless there is many more power issues which need to be resolve.

(vii) Cooperation among nodes

Like the security, mobile nodes mutual cooperation is also an issue for the commercialization of the ad-hoc network technology. The first question arises as to why mobile device belonging to one person should relay the data of other person's device? The answer is as simple as that, to avail the corresponding service from the others you must cooperate with them. Though, when the priority and amount of data is considered, the scenario becomes more complex. For instance, a fire alarm device should not waste its scarce power for relaying the movie video data.

(viii) Interference between two ad-hoc networks

The autonomy of an ad-hoc network is a tedious task when two independent ad-hoc networks reach into each other's range. This is an unaddressed research area that has effect on all layers of system design.

The question is what happens if two independent ad-hoc networks are encountered. In an ideal situation, the network should be able to diagnose the situation and be combined. Unfortunately it is not simple because network may be utilizing different MAC or routing protocol and synchronization mechanism. Moreover reliability and security among different networks are major issues.

(ix) Compatibility with other wireless technology

Another vital issue is compatibility of ad-hoc network with other existing wireless technologies. When taking in to the consideration all wireless networks, an all-in-one integration of all kind of networks is imperative. This issue includes that how an ad-hoc network should be designed so that it is compatible with wireless LANs, 4G cellular network etc.

1.5 MOTIVATION

With the easy accessibility of hand held communication devices and popularity of mobile ad-hoc network technology, it is imperative to facilitate the user and provide the Quality of Service for routing of heavy and real-time data like video and voice with minimal control overhead in ad-hoc scenario. Therefore an efficient routing algorithm for mobile ad-hoc network environment is needed that ensures the reliable end-to-end transmission with certain Quality of Service. Also, as discussed earlier, scalability for a given ad-hoc network is primary requirement, which upholds the basic functionality of an ad-hoc network. With increasing number of terminal an efficient scalable routing structure is necessary.

1.6 OBJECTIVES OF THE RESEARCH

This research is aimed to design a location aided protocol which supports the Quality of Service in terms of available bandwidth and battery power and at the same time ensures the minimal control overhead. This research work also aims to design an efficient weight based clustering framework which selects the healthy cluster head with fair load distribution among the elected clusters-head and ensures the minimal cluster maintenance cost.

The main objectives of the work are listed below:

1. To study the Quality of Service issues in existing location based routing algorithms.
2. To study the classic weighted based clustering algorithms.
3. To articulate the problem definition based on literature study.
4. To harness the location aided routing and builds a protocol which provides Quality of Service in terms of required battery life and available bandwidth.
5. To design an efficient load balanced weight based clustering framework for mobile Ad-hoc network
6. To validate the developed algorithms through simulation.

1.7 CHALLENGES AND SOLUTIONS

A critical look at the available literature indicates the following issues need to be addressed towards building of efficient routing algorithms for mobile ad-hoc networks.

(i) Involvement of immaterial nodes in path construction process

In the all reactive routing algorithms for mobile ad-hoc networks, route request packets are flooded in whole network irrespective of the direction of destination. This imposes the control overhead on those nodes that cannot be the candidate for path construction at all, consequently occupies the scarce bandwidth unnecessarily.

Solution: *In path construction, to ensure the involvement of significant nodes only, a novel location aided routing algorithm has been developed by improving the existing restricted directional flooding approach. In this approach path discovery process moves in forward* *direction only to converge at destination i.e. transmitting node excludes the nodes that are in the backward direction, hence saves the network resource.*

(ii) Quality of Service in location aided routing

The existing location aided approaches overlook the QoS parameters of participating nodes that constitute the path causing frequent link break and poor transmission quality.

Solution**:** *An improved location aided routing protocol equipped with Quality of Service check while selecting the next forwarding node for path construction has been developed. In this approach, nearest node w.r.t. the reference line that fulfills the required QoS parameters is selected as next forwarding node.*

(iii) Complete path reconstructions for partial path break

In mobile ad-hoc network, most of the routing model comprises of complete path reconstructions even if the single link breaks, thereby efforts done in previous route construction gets wasted that result in high maintenance cost.

Solution: *In order to reduce the maintenance cost, a novel path maintenance procedure has been developed that repairs the broken link locally instead of complete path reconstruction.*

(iv) Involvement of cluster-head in trivial communication

In cluster based routing, if two non-cluster-head nodes want to communicate with each other, they must transmit packet via their respective cluster-heads, even if the communicating nodes are in the transmission range of each other (trivial communication). This model of routing, overloads the cluster-heads by those communications that can be accomplished even without involving them.

Solution: *In order to lighten the load of cluster-heads, a novel communication model has been developed, that spares the cluster-head from trivial communication and increases their life.*

(v) Huge computational overhead in weight based clustering.

In cluster-head selection process, every node has to compute their weights based on certain parameters and exchange the control messages in its locality for claiming to be cluster head. This not only induces the communication overhead but also loses scarce battery power therefore reduces the network life time.

Solution: A *novel weight based clustering algorithm has been developed that marks the incompetent nodes and spare them from participating in cluster head election process. In this manner, algorithm blocks the unnecessary injection of traffic and saves the battery power at node level thus increases the lifetime of network.*

(vi) Unfair load distribution among the cluster-heads

In existing weighted based clustering algorithms, load distribution on cluster-heads remain unfair because it takes the generalized ideal number of nodes that a cluster-head can handle. However, in heterogeneous ad-hoc networks, nodes are distributed randomly, so degree of node remains different for every node which may leave many cluster-head overburdened.

Solution: *To ensure the fair load distribution over the cluster-heads, instead of generalized upper bound, in the proposed approach the fair number of nodes that a cluster-head can handle pertaining to its available battery power has been computed.*

(vii) High maintenance cost due the restriction on cluster size up to one hop

Another issue with the existing weighted clustering algorithm is that it allows only one-hop that leads to the situation where if a node is unable to hear a cluster-head with direct link, it triggers the election process in its locality and progressively induces the election process in entire network that results in higher maintenance cost.

Solution*: To minimize the maintenance cost, a new event called inmate affiliation has been introduced. The concept of inmate node overcomes the ripple effect of re-clustering. If a member node is unable to hear its cluster-head then it affiliates itself with nearby cluster as member node or inmate node, silently rather than to make a separate cluster and trigger the re-clustering in entire network.*

(viii) Large number of clusters

The limitation of cluster size up to the single-hop creates larger number of clusters than required, i.e. the nodes that remain uncovered or not able to join a cluster head directly forms their own cluster with only one node. Consequently hop count over the path between the terminal nodes increases.

***Solution**: The inmate affiliation allows the merging of trivial clusters (cluster having only one node, resultant of initial clustering) with primary clusters consequently minimizing the number of clusters in the network.*

1.8 ORGANIZATION OF THESIS

This thesis contains nine chapters and the contents of the chapters are as follows:

Chapter 1: Introduction

In this chapter, the basic idea and concept of mobile Ad-hoc network with potential applications is presented. Along with the basic idea, comparison with its wired counterpart is also discussed. Thereafter, the design challenges and routing issues are touched. At the end of the chapter, motivation of research work along with identified objectives and challenges and corresponding solution are outlined.

Chapter 2: Literature Review

This chapter begins with an extensive review of literature published in various books, national and international journals and conference proceedings on topics relevant to the present research theme. It also discusses basic assumption, model with respect to mobile ad-hoc networks, classification of routing approaches for MANETs and classic routing algorithms. Clustering model in mobile Ad-hoc network and comparative studies of typical clustering strategies are also discussed. It also comprehends the comparative study of various location based routing protocols. Chapter ends with the discussion of various QoS aware routing protocols.

Chapter 3: QoS Enabled Improved Location Aided Routing (QEILA)

This chapter presents a novel location based routing algorithm “QoS Enabled Improved Location Aided Routing (QEILA)”. It also discusses simulation outcomes of proposed algorithm and findings are compared with corresponding existing approach.

Chapter 4: Location Information Based Destination Converging Routing Method (LIBDCR)

This chapter proposes a novel location based routing algorithm “Location Information Based Destination Converging Routing Method (LIBDCR)”. The simulation outcomes of proposed algorithms are also presented and findings are compared with corresponding existing approach.

Chapter 5: Light Weight Efficient Cluster based routing Model for Mobile Ad-hoc networks (LWECM)

In this chapter, a Light Weight Efficient Cluster based routing Model for Mobile Ad-hoc networks (LWECM) is proposed and demonstrated. Also the comparison with standard cluster based routing model is presented.

Chapter 6: Marking based Load balancing Weighted Clustering framework for Mobile Ad-hoc network

This chapter proposes a novel cluster based routing framework “Weighted Clustering framework for Mobile Ad-hoc network. The simulation outcomes of proposed algorithm are also presented and findings are compared with corresponding existing approach.

Chapter 7: Conclusion and future scope

The conclusions that have emerged from the results and discussion are summarized in this chapter. Suggestions for future work are also presented.

CHAPTER 2

LITERATURE SURVEY

2.1 INTRODUCTION

A mobile ad-hoc network is established by battery driven mobile nodes, which cooperates with each other for data transmission. Communication between two nodes is done in multi hope fashion with limited bandwidth. A typical scenario of an arbitrary ad-hoc network is depicted in Fig 2.1(a) and 2.1(b). Where the mobile nodes connected with a variable wireless link results in dynamic topology that is connectivity between two mobile nodes is unpredictable. For example in Fig. 2.1.(a) node N1 is connected with a direct link with N4 and as the nodes are free to move in Fig.2.1.(b) node N4 move out of the range of N1, meanwhile N3 reaches in the range of N1 then N1 got connected with N4 through N3 with 2 hop. From the example it can be comprehended that, standard bellman-ford (distance vector routing) or link state routing (designed for wired network) are unsuitable for mobile ad-hoc network. [3]

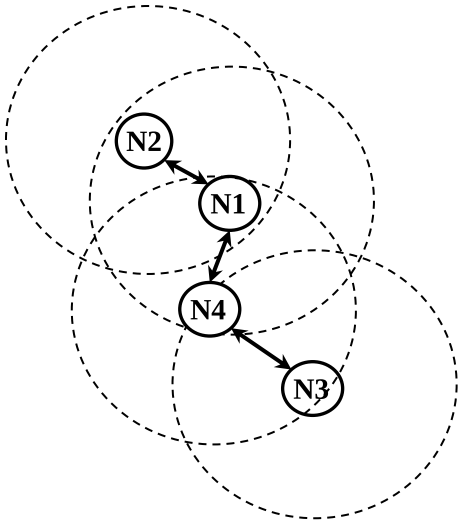


Figure 2.1(a): N1 connected with N4 by a direct link

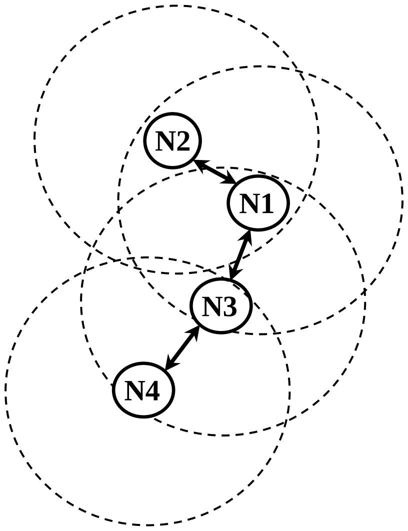


Figure 2.1(b): N1 connected with N4 through N3

Researchers have proposed numerous routing algorithms for mobile ad-hoc network. Routing in MANETs depends on several parameters such as topology, selection of path, physical position of route request initiator, and particular fundamental characteristics that can utilize for heuristic in rapidly and efficiently path finding. The primary challenge in scheming a routing algorithm for MANETs [10] is that a node must require to know the vicinity information of its neighbors for determining the route of the packet, nevertheless, the topology of network may change frequently. Moreover as the number of nodes in the network is unbounded and determining the path to a destination node may require exchange of control information among the nodes, therefore the amount of control traffic can be substantial and may grow exponentially with increasing number of nodes and it may be the case where mobile node inject control traffic such a way that no bandwidth left for the data traffic [11].

2.2 CLASSIFICATION OF ROUTING ALGORITHMS FOR MOBILE AD-HOC NETWORK

There are many ways to classify the routing algorithms for mobile ad-hoc networks. In this work the attempt has been made to distinguish the major characteristics of classic routing paradigm. Routing algorithms for MANETs can be classified primarily as the topology based and the location based. In topology based routing algorithm, routing decision is taken by a mobile node on the basis of its local topological information, that is existing radio link with direct or indirect neighbors. Topology based routing further classified in Flat /uniform routing and Hierarchical/cluster based routing. In flat/uniform routing all mobile terminals in the network have equal privilege and functionalities. Whereas in hierarchical//cluster based routing algorithms some nodes holds the special coordination and/or routing functions. Flat routing is further classified in table driven (proactive) and on-demand (reactive). In proactive routing algorithms a given mobile node maintains the path to all possible destinations in its routing table. For this it follow the standard distance vector or link state routing mechanism, any topological changes are propagated throughout the network and nodes update there routing table accordingly. In this way, the path to any destination is readily available (proactively) with a node to transmit the data. The advantage of table driven routing approach is that a node can obtain the route immediately from its routing table whenever it needs to transmit the data. However the problem with proactive approach is that a node has to maintain the path even if it has no data to transmit. Therefore proactive approach may effect adversely in case of high mobility of network nodes. Because for keeping the routing information (table) fresh, a substantial amount of network resources is needed. Apart from the mobility the quality of link keeps changing due to the shadowing or fading [12]. Conversely in on-demand routing algorithms, a mobile node discovers the rout, only when it needs to transmit the data to destination. In this reactive approach the algorithm consumes the minimal network resources such as channel bandwidth. But the disadvantage of reactive approach is that it faces the delay to transmit the first packet to destination, as the time taken to construct the pat is substantial. Once the path is determined the transmission takes place as usual. Another issue with reactive approach is the upswing of control traffic to maintain the path in case of dynamic topology.

Another topology based routing tactic is hierarchical/cluster based approach, which utilizes the advantages of proactive as well as reactive routing, approaches, and finds the optimized configuration of both. In this approach a given node communicate by some sort of proactive routing algorithm in it local topology and for out of the locality it follow a reactive routing algorithm. Hybrid approach is inherently adaptable with the increasing number of node and ensures the scalability of ad-hoc network. Commonly in a hybrid approach, network is partitioned in some kind of hierarchy on the basis of topology or responsibility of nodes. The major advantage of hybrid approach is that, the dynamic network topology seems less dynamic from the point of view a single node. Because a node need to aware about its local topology only and any changes to topology dose not propagates in entire network.

Topology based routing approaches have their side effects, such as large control overhead due to the flooding of control packets in entire network, irrespective to the direction of destination. Which leads to wastage of network resources (power and bandwidth) Location based routing algorithms overcomes that problem of topology based routing approach, by associating the physical location information of mobile node, along with the basic routing information. In location based routing system, every mobile node need to equip with a tiny position tracking hardware like GPS module. Every node is able to determine its physical location of itself and all other nodes in the network through some sort of location service

[13](discussed later in this chapter). Whenever a node wants to communicate with another node, it first determines the physical location info of destination node and enclosed this information into a request packet and forward in its vicinity. In succession, the routing decision of this packet is based on the destination nodes location (wrapped in packet) and the location of the forwarding node. It is comprehensible that location based routing approach does not need the construction and maintenance of path that saves the control overhead. But it comes with the cost of extra tiny location hardware. Location based routing approach can also be utilized in geo-casting that is; delivery of a packet to a sub group of nodes belongs to a particular geographic region. The taxonomy of routing protocol for mobile ad-hoc network is depicted in Fig. 2.2

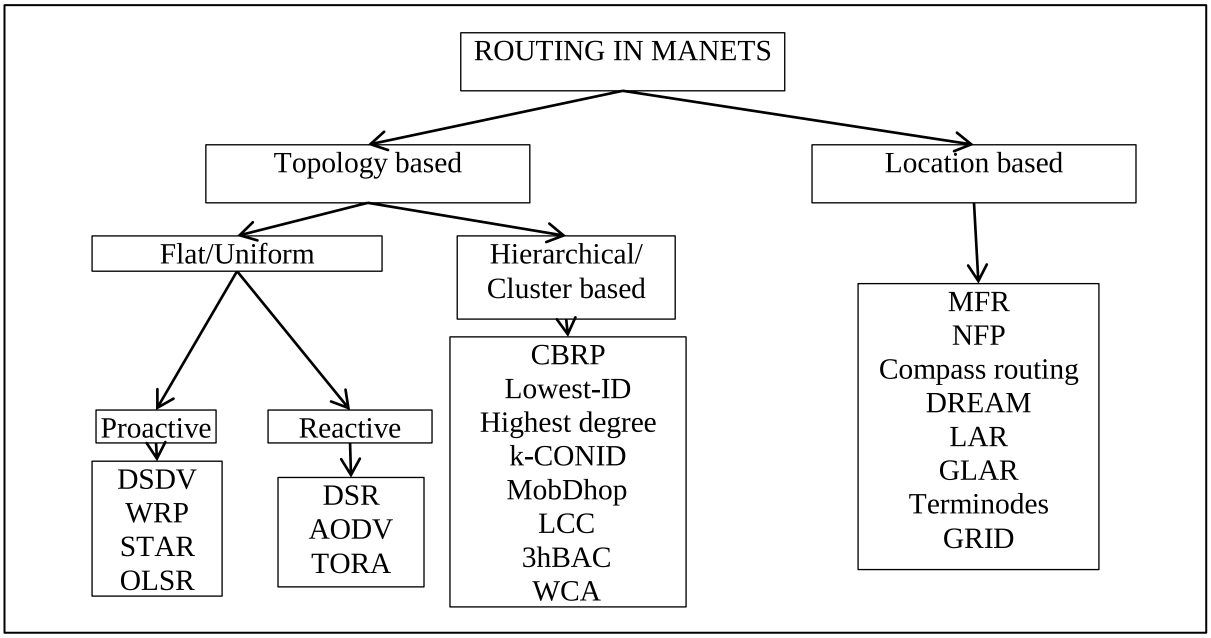


Figure 2.2: Taxonomy of routing protocol for mobile ad-hoc network

2.2.1 Topology based routing approaches

In this section, classic topology based routing strategies has been discussed and distinguished according there inherent characteristics with their associated merits and demerits. Topology based routing approaches can be flat where all node work on the same level or hierarchical in which nodes have specific responsibilities at different level.

2.2.1.1 Flat proactive (table driven) routing approaches

Flat proactive routing algorithms are the wireless version of wired routing algorithm. In these approaches, every node maintains the global state of network in the form of routing table and updates it periodically with fresh topological information to have consistent view of network. Whenever a node has data to send, it consults it routing table to get the appropriate rout to destination. The typical examples of this category are DSDV, WRP, TBRPF, OLSR.

a) *Destination-Sequenced Distance-Vector Protocol (DSDV)*

The Destination-Sequenced Distance-Vector Protocol (DSDV)[[14] is the very first member, in the ad-hoc network routing protocol family that imitate the classic bellman-ford routing algorithm used in wired network, in which every node manages a routing table containing shortest path (in hop count) along with first node in this path, to every possible destination. For consistent view of network routing table at each node, needs to update periodically with fresh routing information, with ascending sequence number. Sequence number used to prevent the count-to-infinity problem and to overcome the looping of packet. As in this routing approach, mobile nodes maintains the routing table proactively, they can get the path to any other node without any delay. These tables may be shared with other if a node found a significant change in topology. Table can be updated in two ways namely; incremental update and full dumps. In incremental update a single information packet is shared as the node observed a trivial change in topology, while in full dump, a given node observes, significant changes of topology and more than one information packets are needed to be shared. Update of tables are being instigated by a node, with a different sequence number, is always larger than the previous sequence number. On receiving multiple version of a table update from multiple neighbors a node hold it for some time to determine the best matric or updates its table based on the received tables. By checking the sequence number of table update, a node can reject it or relay it to its neighbors. Fig. 2.3 depicts the typical scenario of an arbitrary mobile ad-hoc network where solid circle with numbers depicted as mobile node, dotted circles are representing the radio range of nodes and bidirectional arrows are representing the wireless link between nodes. Here N2 is source node and N4 is destination node.

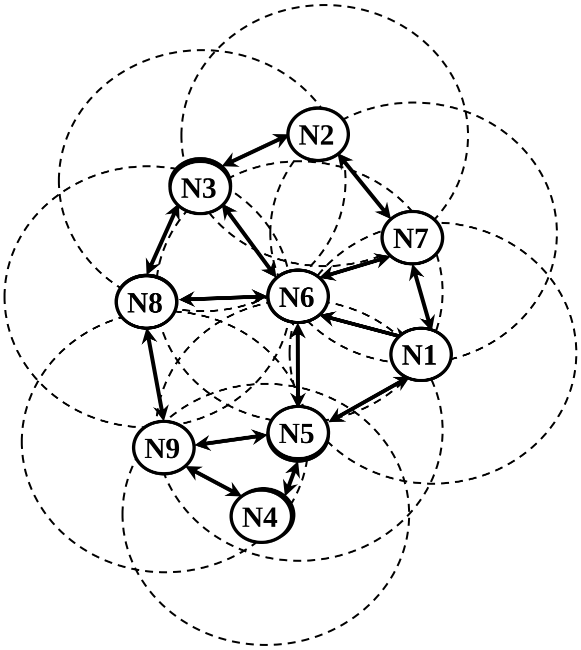


Figure 2.3: Topology of an arbitrary ad-hoc network

Table 2.1 showing the routing table at N2 in which global topological information is maintained. It can be seen that the shortest route to destination node N4 is shown with sequence no 144 (emphasized with dashed box) having hop_count 4 and next node is 7.

Table 2.1: Routing table at node N2

Destination_No	Next_node	Hop_count	Sequence_No
1	7	2	133
3	3	1	35
4	7	4	144
5	3	3	56
6	3	2	63
7	7	1	121
8	3	2	42
9	3	3	24

The advantage of DSDV is that given source has the readily available route to every possible destination. Therefore, if a node has something to transmit it could be sent without any delay. The disadvantage is that a node has to maintain path to all possible destinations, all the time, even when it has no data to send. This leads to huge maintenance cost.

b) *The Wireless Routing Protocol:*

In table driven routing protocol family, another standard routing protocol is Wireless Routing Protocol (WRP) [15]. In this protocol also, the entire node in the network, proactively maintain the global state of network. To get the more precise view of network, every node maintains four tables: Link-cost table (LCT), Distance table (DT), Routing Table (RT), Message Retransmission List table (MRL). The Link-cost table (LCT) contains the cost associated with a given route, in terms of hop count. For broken link the cost is considered as infinity (∞). It also records the interval between two consecutive update. Distance table (DT) maintains the topological view of network, in the form of a two dimensional matrix in which ever entry contains the hop-count and the second last node towards a given destination node. The routing table (RT) maintains the consistent view of the network-state (routes) for all possible destinations. Topology of an ad-hoc network with link cost is shown in Fig. 2.4.

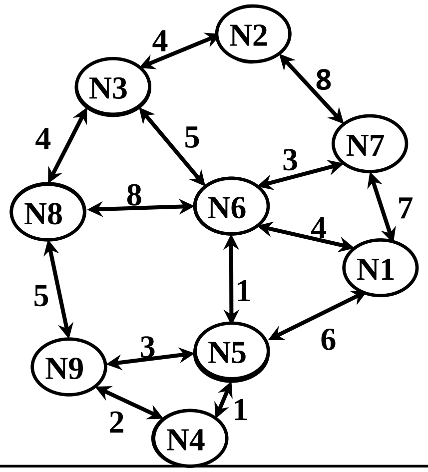


Figure 2.4: Arbitrary topology of an ad-hoc network with link cost

Routing table contains the shortest distance, second-last node to destination, the succeeding node from source and a flag representing the condition (Correct, erroneous, NULL) of route. A correct route means that all the links up to destination are alive, erroneous route means the existence of loop and NULL means the destination not located. The message retransmission list table (MRL) maintains the record of each update message to be relay along with the counter for each such entry. To ensure the link break this counter is decremented for each transmission of update message. Once the counter becomes the zero, message is deleted and updated message retransmitted. The routing record at every node for destination node 2 is shown in table 2.2.

Table 2.2: Routing record at every node for node N2

Node #	Next node	Predecessor	Cost
N1	N6	N3	13
N2	N2	N2	0
N3	N2	N3	4
N4	N9	N3	15
N5	N9	N3	16
N6	N3	N3	9
N7	N2	N7	8
N8	N3	N3	8
N9	N8	N3	13

Source-Tree Adaptive Routing Protocol is discussed in next sub section.

c) *Source-Tree Adaptive Routing Protocol*

The major side effect of all table-driven routing approach is huge control overhead. As every node has to update its routing table periodically, which is leads to depletion of scarce resources. [16] attempt to minimize the control overhead in table driven routing approach and proposed, Source-Tree Adaptive Routing Protocol [STAR]. Unlike the other table driven routing algorithms, in STAR nodes does not need periodic update of routing table. In this approach each mobile node shares its source-tree-information. Source tree information of a node, conveys the preferred path used to reach some destination by that node. Each node determines a sub-graph of network topology with its local topology and the source-tree-information from neighboring nodes. In the beginning, a given node broadcast update messages to its neighbors, containing route to new destination, probability of loop, and the cost. In this way every node gets the route to every possible destination. STAR follows the least control overhead approach and yields a compromised path than an optimal path. When a source node S wants to communicate with a destination node D, It consult its routing table (source-tree-information) for path. If there is no entry is found for that destination it broadcast an update message in its vicinity to inform all neighbors the absence of path to D. Source node S keeps re sending the update message with growing interval until it gets the path to D. In continuation of this, if neighbors have a path to D, They respond with an update message. Source node S then updates its source tree information. In this manner it gets the path to all possible destinations. Every node in the network follows the same process and eventually each node gets the path to remaining nodes.

d) *The Optimized Link State Routing Protocol*

Optimized Link State Routing Protocol (OLSR) [17] is another table driven routing approach where the optimization over the standard Link Stat Routing approach used in wired network. In classic link state routing protocol, all direct wired link with neighboring device are determined and are propagated throughout the network. While in case of optimized link state routing protocol (OLSR) packet size of control packet is reduced by keeping the subset of direct link instead of all. Apart from that it prorogates the link state information to the some selected nodes (multipoint relay) only rather in entire network. The concept of Multipoint Relay [18] is used, for minimizing the control traffic over the network by omitting the proliferation of duplicate packets on the same set of nodes. Every node in the network picks the certain set of nearby nodes to propagate packet, these set of neighboring nodes called multi-point relay (MPR) of that transmitting node. During the propagation of packet if a node receives the packet, first it test whether it is in the MPR list of transmitting node or not, if no it receives the packet but not forward it and forward otherwise. A given node pick its MRP members such that, it can cover all two-hop away neighbors. Furthermore, in case of link break, OLSR does not produce extra control messages like other proactive routing protocol. The provision of proactively maintained routes and optimized control traffic by multipoint relay makes OLSR suitable for network with large number of nodes. OLSR is inherently distributed in nature and does not need any centralized body. OLSR works on best effort delivery approach for control messages, as each node broadcast control messages periodically and tolerate occasionally loss of packets due to transmission problem like collision, hidden and exposed terminal, fading etc. Also OLSR need not to deliver the packet in order; this is because the packet itself contains a sequence number and based on that the ordering of packets is the responsibility of receiver. Routing in OLSR takes place as hop-by-hop, that is, every node utilizes the current information to relay the packet, therefore, even a node is in movement can communicate with other nodes.

2.2.1.2 Flat Reactive (On-demand) routing protocols

In contrast to proactive (table driven) routing approach, in reactive (On demand) routing approach, node does not maintain the route to other nodes all the time. Whenever a node wants to communicate with a certain destination node it invokes a route construction routine. In this way route between communicating nodes is determined on demand. Some classic on demand routing protocols are discussed below.

a) *Dynamic source routing protocol (DSR)*

The Dynamic Source Routing protocol (DSR) [19][20] is a reactive routing protocol. The objective of DSR (and other on-demand protocol) is to minimize the control traffic over the network and save the network bandwidth. DSR achieve the control traffic minimization by omitting the periodic update messages that were required in proactive routing approaches. In DSR communication takes place in two parts, first is 'route establishment' in which route-request packets are flooded throughout the network. The route request packet contains the source address, destination address, sequence number as primary field and other supporting information. The destination node eventually receives the route-request packet and responds with a respective route-reply packet back to the originating node of route request packet. Route-reply packet contains the list of nodes, that route-request packet have traversed. Suppose there is source node S that wants to transmit the data to destination node D. First, node S look up its neighboring nodes, if D is one of them, if so, it send the data directly to D. Otherwise if D is not in the neighboring nodes of S then S has to determine the route up-to node D. Node S then broadcasts a route-request packet, in perpetuation, this route-request packet is flooded in entire network. Every intermediate node I on receiving this route-request packet tests it, if this packet contains the destination ID as its own ID i.e. node I found itself as destination, it respond with a route-reply packet back to the source node S. Otherwise I ensures the freshness of packet by checking if it is already forwarded the packet and the time to live (TTL) has not reached the limit. After ensuring the freshness of packet it forwards the packet in its vicinity. Every route-request packet contains a sequence number created by source node and the list of node that it has passes through. An intermediate node test the sequence number to prevent the duplicate transmission of packet, received from multiple routes. Also the sequence number is utilized to ensure that the packet has not being looped. For example in Fig. 2.5 an arbitrary ad-hoc network is shown, where labeled node is representing the mobile node and both headed arrows representing the bidirectional radio links. Suppose source node N4 wants to transmit data to destination node N2.

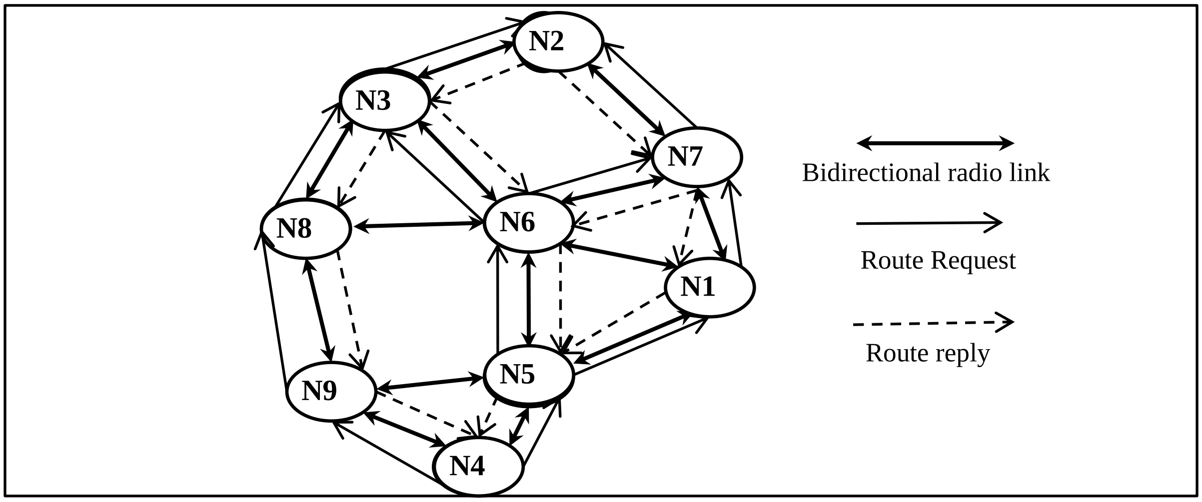


Figure 2.5: Arbitrary ad-hoc network with bidirectional link

Then N4 initiate a *route- request* RERQ packet and broadcast it in its vicinity (to direct neighbors) i.e. N9, N5. Upon receiving the RERQ, N5 and N9 test the destination-ID field of RERQ, whether they are the intended destination or not. In continuation they further broadcast the RERQ to their direct neighbors. In this way the RERQ eventually proliferated in entire network. It can be seen that node N3 has received the RERQ twice from its two neighbors N8 and N6. In this case N3 will drop the one of these two RERQ which has arrived later. A node identified a duplicate packet by knowing the sequence number contained in that packet. For a given route- construction round the sequence number remains the same in RERQ. When RERQ arrives at a destination node, it responds with a route-reply RREPLY packet back to the source node by back tracking of nodes from which the RERQ was passes up to destination. It can be clearly understood that source node may receive more than one route-reply RREPLY, reflecting the existence of multiple routes from source to destination. In this situation source node has the choice to select the optimal route. In the network shown in Fig. 2.5 there are four possible routes that are N4→N5→N6→N3→N2, N4→N9→N8→N3→N2, N4→ N5→N1→N7→N2 and N4→ N5→N6→N7→N2.

DSR employs a simple and straight forward route maintenance mechanism. Whenever an intermediate node found a broken link, it initiates an error RERR message and sends back to the source node. On receiving the RERR, source node re initiate the route construction procedure. As a reactive routing protocol DSR comes with the many inherent advantages.

As this protocol belongs to reactive class of routing, it does not require periodic update for table which saves the network resources i.e. battery power and bandwidth. Another advantages is that, a node need not to maintain the route to every node in the network, all the time as route is constructed whenever it necessary. The main disadvantage of this protocol is the delay in sending the first data packet because the source node has to wait until the path is constructed.

b) *Ad Hoc On-Demand Distance-Vector Routing Protocol*

The Ad-hoc On-Demand Distance Vector routing protocol (AODV) [21] is a fusion of reactive and proactive routing approach. Therefore it possesses the quality of both the paradigm. It imitates the Dynamic Source Routing (DSR) protocol in route construction and route maintenance, along with the use of periodic beaconing and sequence number like Destination-Sequenced Distance-Vector Protocol (DSDV). In AODV a node discovers the path whenever it wants to communicate with a destination node like in DSR. But it differs from DSR in maintaining the route information for transmitting the data. In DSR, a data packet contains the complete route information i.e. list of node in path from source to destination. However in AODV, a data packet travels in hop-to-hop manner i.e. source or an intermediate node maintains the next hop information to forward the data packet for a particular flow of transmission. So AODV is the pure on-demand routing protocol that does not even maintains the complete path information but transmit the data in ad-hoc basis. Like other reactive routing protocols, in AODV a node flooded the route-request packet in entire network if it has no path to a given destination and may yields the multiple route-replies (path) for same route-request. The packet format for route-request packet is shown in Fig.

2.6. The difference between other standard reactive routing protocols and AODV is that, it has the provision of destination sequence number contained in a packet like DSDV. Destination sequence number ensures the freshness of a route a node updates its routing information if it gets the packet with a larger destination-sequence-number than it already has with last packet.

I--32 bits --I

<table>
<tr><td>Type</td><td>Reserved</td><td>TTL</td></tr>
<tr><td colspan="3">Broadcast ID</td></tr>
<tr><td colspan="3">Destination IP Address</td></tr>
<tr><td colspan="3">Destination Sequence Number</td></tr>
<tr><td colspan="3">Source IP Address</td></tr>
<tr><td colspan="3">Source Sequence Number</td></tr>
</table>

Figure 2.6 : Packet format for route request packet in AODV

As shown in Fig. 2.6 in AODV route request packet contains the source IP address, destination IP address, source sequence number, destination sequence number, broadcast id, and time to live TTL sections. The source IP address section contains the unique IP address of originator of route request packet and destination IP address section contains the IP address of intended destination node. The destination sequence number represents the newness of the path that is established by the source node. The rationality of path, at a given node is determined by matching the sequence number of path in routing information that node with the sequence number of path in the received route request packet. In path construction process, source node initiates a route request packet and broadcast it in its vicinity, if a given node encounters a route request packet, it has to perform one of three operations, first It may forward the route-request packet further in the network, second: it may be the case, where an intermediate node has received the same route-request packets multiple times (indicated by broadcast ID – Source ID pair) in this case duplicate packet is discarded and third: it may create a route-reply packet (packet format of route-reply is shown in Fig. 2.7), if it is the intended destination or have the valid path up to the destination and pass this route-reply packet on the reverse path back to the source node. As the route-reply packet travelled along the reverse path every node in this reverse path make an onward route entry in their routing table. Each route entry is accompanying with a route timer for the aliveness of path for specified of time. As a matter of fact AODV has the provision of symmetric link only, as route reply traverses the revers path travelled by the route-request packet.

Type	R	A	Reserved	Prefix Sz	Hop Count
Destination IP address					
Destination Sequence Number					
Source IP Address					
Lifetime					

Figure 2.7: Packet format of route reply in AODV

In AODV a complete path repaired mechanism is used for the path maintenance. Whenever an intermediate node happens to know about the breakage of link, it reports the source node, by sending a special route error message on reverse path. The packet format of route error is shown in Fig. 2.8. When a source node come to know about the path break, it rebuilds the complete path again. The primary gain of this protocol is establishment of route on purely on-demand basis. The destination sequence number takes care of the freshness of route. The route establishment delay is less as compare to other on demand routing protocols. The disadvantage of AODV is the periodic beaconing which results in wastage of bandwidth. Also the multiple route reply packet for same route request packet leads to huge control overhead.

I---32--I

Type	N	Reserved	Dest Count
Unreachable Destination1 IP Address			
Unreachable Destination1 Sequence Number			

Figure 2.8: Packet format of route error in AODV

c) *Temporally Ordered Routing Algorithm*

The Temporally Ordered Routing Algorithm (TORA) [22] is another source initiated loop free routing approach and designed to counter the side effect of dynamic topology of ad-hoc network. TORA utilizes the concept of *link-reversal*. TORA is somewhat reactive and somewhat proactive, it is reactive because route is constructed on demand basis and proactive because, it maintains the multiple paths in advance to cope up path break. In TORA every mobile node maintains the local topological information of their direct neighbors and has the ability to identifying the logical partitioning of network. To minimize the control traffic, TORA has a unique feature of controlling the boundary of propagation of control messages, while the maintenance of path due to the link breakage. TORA has three actions regarding the routing process that are: construction, maintaining and removing the routes.

The route construction is invoked only when a source node wants to communicate with an intended destination node, but dos not have a direct link to that node. The route construction process yield a destination focused *acyclic directed graph* (so that multiple route to destination) by utilizing QUERY/UPDATE mechanism. Whenever a source node S wants to communicate with destination node D, it initiates a QUERY message containing the address of D and broadcast it in its locality. The QUERY message proliferates throughout the network and eventually arrives at the destination node D or an intermediate node I which has a valid route up to the destination. On receiving a QUERY message, every intermediate node I creates an UPADTE message containing the height of I pertaining to the destination node D and broadcast it throughout the network. On receiving the UPDATE message every node sets its height greater than the height of neighbor from which it receives the UPADTE message. This process yields the trail of directed links from source node S to the initiator of UPDATE message.

When an intermediate node realizes that the a route to the destination D is no longer alive, it tuned its height to local maxima pertaining to its neighbors' height and broadcast an UPDATE message. Also if a node N does not surrounded by the neighbors with valid height pertaining to destination D. Then N tries to build a new route with same process. When a given node identifies the logical partition of network, it creates a CLEAR message to reset the network state and omits the invalid routes. In route removing process clear message (CLR) are flooded across the network to remove the invalid routes. The Temporally Ordered Routing Algorithm (TORA) is stacked on the top of internet MANET Encapsulation Protocol (IMEP) [23] that is developed for *in-order* delivery of control messages between peers along with the announcement of creation or breakage of link. The advantage of TORA is, it generates comparatively less control overhead, by limiting the proliferation of control traffic. The disadvantage of TORA is, it occasionally oscillations due to the parallel detection of partition by deferent nodes.

2.2.1.3 Hierarchical or Cluster based approach routing approaches

It can be comprehended that both the flat routings reactive or proactive imposes the great control overhead over the network especially when number of nodes is large and dense, therefore flat routing schemes are not scalable.

To deal with such problems, hierarchical routing is used. In hierarchical routing, network is divided into group of neighboring nodes called clusters [24]. Partitioning the network into clusters reduces the control overhead because a node merely required knowing the routing information in its cluster and not of the complete network. Hence cluster based routing make the large network appear smaller and the topology which was dynamic appear less dynamic.

In cluster based routing, every cluster has a node called cluster-head which acts as coordinator and responsible for intra cluster and inter cluster transmission [25][26][27].

A cluster-head selection policy is used to elect a cluster-head, while setting up cluster structure. The main objective of a clustering algorithm is to select a healthy and stable cluster-head to prolong the network life and efficient routing [28]. Cluster members who listen more than one cluster-head or a member of nearby cluster, act as a doorway node [29][30] and utilized for inter cluster communication. Rest of the members act as ordinary node. An ordinary nodes is the member of a cluster which neither the cluster-head not the doorway node. A standard cluster structure is depicted in Fig. 2.9.

The partitioning of network is basically the selection of cluster-head because it is the cluster-head who form its cluster by following, some sort of criteria. Here is a standard cluster based routing approach CBRP [31] has been presented. Thereafter some standard cluster head election mechanism has been discussed.

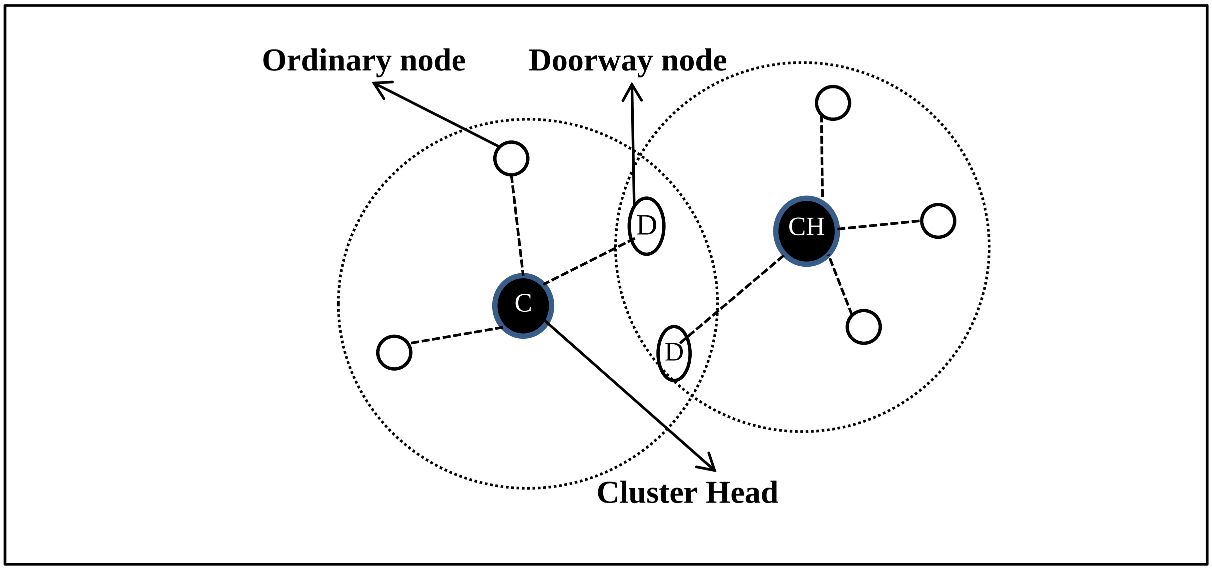


Figure 2.9: Standard cluster structure in MANET

a) *Cluster Based Routing Protocol (CBRP)*

Cluster Based Routing Protocol (CBRP) [31] is a definitive protocol which is based on hierarchical approach. In CBRP, mobile nodes are grouped into sub networks known as *cluster*. Every cluster contains a node with additional responsibility called cluster-head. Cluster-head is elected among neighboring mobile node by lowest ID heuristic (presented ahead in the chapter).

After cluster-head election phase, entire network get partitioned into the overlapped or non-overlapped clusters eventually. Cluster-head keeps the record of its cluster member and facilitate them for inter and intra cluster communication. All non-cluster-head node in a cluster considered as cluster members and assumed to be attached with their respective cluster-head with bidirectional link. Nodes who all are able to hear the member nodes or cluster-head of more than two cluster act as a doorway nodes and assist the inter cluster data transmission.

In CBRP every node is able to determine the unidirectional or bidirectional link to its neighbor along with its status. For this very purpose every node maintains a neighboring table as shown in Table 2.3 and broad casts it at regular interval though *hello message* .

Table 2.3 : Neighboring Table in CBRP

Neighbor_ID	STATUS_OF_LI NK	ROLE
Neighbor_1	Unidirectional / bidirectional link	Is 1 is cluster head?
Neighbor_2	Unidirectional / bidirectional link	Is 2 is cluster head?
-	-	-
Neighbor_N	Unidirectional / bidirectional link	Is N is cluster head?

The packet format of hello message is shown in Fig. 2.10. The hello message is used for cluster formation and link set up.

<table>
<tr><td>MY_ID</td><td>MY_STATUS</td></tr>
<tr><td colspan="2">Neighboring Table</td></tr>
</table>

Figure 2.10: Packet format of hello message in CBRP

MY_ID: IP address of sender of hello message

MY_STATUS: cluster-head / cluster-member / undecided (undecided represents that the sender still the orphan node and yet to be member of any cluster)

Cluster formation in CBRP is very simple; initially every node broadcasts the *hello message* in their vicinity, on receiving the *hello message* every node tests the ID of itself and the sender of *hello message*. The node with lowest ID wins the election and changes its status in succeeding *hello messages* from undecided to cluster-head, consequently all neighbors gets aware about the newly elected cluster-head and changes their status from undecided to cluster member in *hello message*. Cluster-head then form the cluster and takes care of its entire member connected with bidirectional link. In cluster maintenance CBRP follows the straight forward approach that is, no cluster member can claim the cluster-head status even if it has the lower ID than its cluster-head. In this regard CBRP not follows the strict lowest ID mechanism. Only if two cluster-head arrives into each other’s radio range, one with lower ID has to relinquish the cluster-head status. In this way CBRP reduces the frequency of cluster-head change. If a node dies or loses the bidirectional link (due to movement) from its cluster-head, in both the cases cluster-head and node remove the corresponding link entry from their neighboring table.

When a node switched on or comes into the radio range of a cluster-head and determines the bidirectional link, in either case host cluster-head or newly added member node update their neighboring table which reflects in subsequent hello messages. If a node is orphan and unable to determine the bidirectional link to any cluster-head, it declares itself as cluster-head.

Like other reactive routing protocol, routing in CBRP is source initiated and takes place in three phases; route discovery, message routing and route removal. Cluster based routing limits the flooding of route request message while route discovery. Whenever a source node S wants to transmit data to destination node D. S initiates a route request message contained only its own address and unicast the this RERQ to its host cluster-head. Each intermediate node who forward the RERQ append it's address (if already not contained) in the message. On receiving the RERQ, cluster-head determines the presence of destination, if destination D present in its neighboring table it unicasts the RERQ direct to destination. Otherwise, cluster head unicast this RERQ to all its gateway nodes. Gateway nodes forward RERQ for further route discovery. This process is sustained until the destination D is encountered the RERQ or a node that have the valid route to destination D. On receiving the RERQ destination node D stores the complete path through which the RERQ has arrived and respond with a REPLY message which follow the reverse path and reaches back to the sender node S. On receiving the REPLY message, intermediate cluster-heads may modify the incoming and outgoing link information in REPLY message according to some optimization criteria.

In CBRP route request always follow the standard order that is $S \rightarrow CH_S \rightarrow GTW \rightarrow CH_n \rightarrow$ - - - $GTW \rightarrow CH_D$. A given route is removed from S if it is no longer used or topology has changed. CBRP is the classic hierarchical routing approach which limits the flooding of control messages, therefore work well with increasing number of nodes. But due to the lowest id heuristic unstable node may selects as cluster head and leads to higher maintenance cost. With the above discussion it is clear that, for any cluster based routing mechanism the crucial part is the selection of cluster-head because a cluster head performs additional responsibility without any additional hardware or software capability. In this regard standard cluster head election mechanism has been discussed and compared next in this chapter.

i) *Lowest-ID heuristic*

Lowest id heuristic [32] is a localized clustering algorithm. A node need to know the status of its direct neighbour. In lowest id heuristic every node in network assigned a unique ID. At regular interval nodes broadcast their ID to their immediate neighbours. Every node compare its neighbours unique ID with its own unique ID. A node is elected as cluster head if it has the lowest ID among its neighbours IDs. The shortcoming with this algorithm is that, the highly mobile node with lowest ID can be selected as cluster-head causing frequent re-clustering. The advantage of this algorithms is its simplicity and easy to implement with lesser control overhead.

ii) *Highest degree heuristic*

The highest degree heuristic [33] is also a localized clustering algorithm. Number of direct neighbours a node has, termed as the degree of node. Every node periodically broadcast its degree to its immediate neighbours. Afterwards every node compares their degree with their neighbour's degree. A node with highest degree will be elected as cluster-head. Unstable clusters also formed in this algorithm in the same way it was in lowest ID heuristic.

iii) *k-CONID heuristic*

The k-CONID heuristic [34] combines the lowest ID heuristic and highest degree heuristic . For cluster-head selection the degree of a node is measured as a primary criterion and lowest ID as secondary criterion. This algorithm attempts to overcome the drawbacks of previous two algorithms. Considering only the node degree leads to many ties among nodes and considering lowest ID produces more clusters than required. In k-CONID cluster formed by cluster-head may include all nodes that are at most k-hop from the cluster-head. The k-CONID generalized the degree of node to k-hop connectivity. Thus if k=1 then k-hop connectivity is same as node degree. Initially every node is assign a 'did'= (d,ID) also called cluster-head priority where ***d*** is the k-hop connectivity of node and ID is unique ID. A node is selected as cluster head if it has highest connectivity. In case of tie, node with lowest ID is selected as cluster-head.

iv) MobDHop

MobDhop clustering algorithm [29] partitioned the network into d-hop cluster based on mobility metric. The objective of this distributive algorithm is making the cluster's diameter more adaptable with respect to mobility of node [35]. In this algorithm, a node can measure the closeness of its neighbours by received signal strength. Strong received signal strength means more closeness between two nodes. The MobDhop algorithm needs the calculation of five terms: *estimated distance* among nodes, *relative mobility* among nodes, and variation *of estimated distance over time, local stability*, and the *estimated mean distance*. The relative mobility is the difference of estimated distance between two nodes at two consecutive times instant. Relative mobility signifies if two nodes are coming closer or moving away from each other. Local stability is calculated by using the variation of estimated distance and the relative mobility. Therefore a node with lowest value of local stability becomes the cluster-head. Other routing approaches in which researchers are attempted to overcome the effect of mobility such as DDV-hop algorithm [36]

v) *Least Cluster Change (LCC)*

To reduce this maintenance cost [37] proposed a least cluster change (LCC) approach in which cluster construction and maintenance are carried out in two different stages. Cluster construction follows the simple lowest Id heuristic but re-clustering takes place in only two circumstances, first when two cluster-head, come into the transmission range of each other, second if a node remains un-clustered i.e. unable to hear any cluster-head. LCC approach has proposed significant improvement over lowest ID and highest degree heuristic as cluster-head need not to hold some specific attribute all the time. But still second case may arise where high maintenance cost may increase due the movement of single node which require re-clustering of entire network.

vi) *3-Hop Between Adjacent Cluster-head (3hBAC)*

[38] proposed an algorithm called 3-hop between adjacent cluster-head (3hBAC) in which ripple effect of re-clustering is overcome through keeping the cluster-heads separate by 3-hop. Initially one-hop clusters are created by following the highest degree heuristic. When

two cluster-head arrives in each other’s transmission range, any one of them has to relinquish its cluster head status. In this way neighboring cluster-heads remain separated by 3-hop. In 3hBAC a new status of node called ‘cluster guest’ is presented. A node who is not the member of any cluster (unable to hear any cluster-head) and cannot serve as cluster-head but it can hear some member node then it attach itself with that member node as cluster guest. 3hBAC reduces the maintenance cost considerably as re-clustering of disturbed cluster not raises the ripple effect in entire network. Also concept of guest node allow the un-clustered node to join the network silently rather to create a separate cluster. But the cluster-guest may overload the cluster-head.

vii) *Weighted based clustering Algorithms (WCA)*

Weighted clustering algorithm (WCA) [39] considers the four factors for weight calculation first is degree difference $\Delta \boldsymbol{v} = | \mathbf{d}_v - \boldsymbol{\delta} |$ where d_v is the degree of node and $\boldsymbol{\delta}$ is the ideal number of node that a cluster-head can handle efficiently, this parameter controls the load on cluster-head. Second is D_v where is the sum of distance from a node to its all adjacent nodes. D_v indicates the energy requirement as more energy is required for larger distance. Third is M_v which is the measurement of mobility of a node. It is the displacement of node n for a specified time T. Fourth is P_v which is the cumulative time for a node served as cluster-head. P_v indicates the power consumption, as cluster-head consumed the more power than a non-cluster-head node. Every node calculates its weight by the equation 2.1.

$$W_v = '\Delta \boldsymbol{v} + w_2 D_v + w_3 M_v + w_4 Pv \qquad (2.1)$$

w_1, w_2, w_3, w_4 are the weight factor for each parameter such that the $w_1 + w_2 + w_3 + w_4 = 1$.

Initially all nodes calculate their weight value and share with their neighbors. Node with minimum weight selected as cluster-head. This process is continued until all nodes become the member of some cluster.

The advantage of Weighted Clustering Algorithm is that it selects stable cluster-head. The disadvantage of WCA is that it puts the huge calculation overhead on the nodes in cluster-head election prior to the actual routing of data. Therefore nodes who have the less battery life becomes the victim of election process even if possibility of winning is negligible.

In Table 2.4 above discussed cluster formation schemes have been tabularized and compared.

Table 2.4: Comparison of clustering schemes

Protocol	**Criteria**	**Election of CH**	**CH change frequency**	**Overlying of cluster**	**Merit**	**Demerit**
Lowest-ID heuristic	Node ID	Node with Lowest ID	High	Moderate	Simple to implement	Highly mobile may elected as CH
Highest degree	Degree of node	Node with Highest degree	High	Rare	Low rate of CH change	Poor in high mobility
k-CONID	Connectivity	Highest degree / lowest ID	High	NO	High control overhead	Yields Less number of CH
MobDhop	Mobility	Node with low mobility	low	NO	Adaptable with mobility	Complex
LCC	Node ID	Node with Lowest ID	low	NO	Low control overhead	Poor in High mobility
3hBAC	Connectivity	Node with highest degree	High	NO	Less number of cluster	Poor in High mobility
WCA	Collective Weight	Node with Lowest weight	Moderate	YES	Elects healthy CH	High calculation overhead

Location based routing is discussed in next sub section

2.2.2 Location Based Routing

In location based routing every node is able to determine its geographic position with the help of a location service like GPS. A location service (discussed in subsequent section) is used by the source node to determine the position of destination node. Source node then put destination’s location information in route request packet and broadcast it. At each intermediate node routing decision is taken on the basis of destination’s physical position contained in route request packet and position of forwarding node's neighbours. Thus the location based routing does not need to establish the path and maintain it in the table. Location based routing supports geocasting [40] that is the delivery of packets in given geographic region.

Basic idea and challenges:

In location aided routing, before a packet can be sent, location of destination nodes need to be determined. A location service is required for this purpose. Available location service may be classified based on how many nodes host this service. There can be some special node that acts as a location server for remaining nodes in the network or all nodes maintain the location of all other nodes in the network. This can be classified as some-to-all, and all-to-all. location services as discussed below:

Some-to-all: In this scheme some specific nodes called location server maintains the location information of all other nodes in networks. Nodes in the network periodically update there location information to location server nodes. Whenever a source node wants to communicate with a destination nodes, it can send the data directly, if destination node present in source node's vicinity, otherwise source node ask the location server node for destination's location information, once source node gets the destination nodes location, it can execute a location aided path construction routine. Problem with this scheme is that, how the source node locate the location sever itself. To communicate with location server , node has to use non-location aware flat routing.

All-to-all : In this scheme all nodes in network maintain the geographical coordinates of all other nodes in the network, all nodes at regular intervals broadcast their location information with a hello message to update their location to other node. Advantage of this scheme is that the location of all nodes in readily available at every node. Disadvantage of this scheme is control overhead in periodic beaconing.For a node, packet forwarding decision in location aware routing, principally based on location information of destination node contained in packet header and the location of neighbouring node of transmitting node. Forwarding scheme in location aware routing can be classified in three categories: restricted directional flooding, greedy forwarding, and hierarchical forwarding as explained below.

i) Greedy forwarding:

In greedy forwarding, transmitting node forward the packet (control or data) to its neighbouring node which is closer to destination node then transmitting node itself, the selection of next forwarding node is based on optimization criteria of algorithm. Greedy forwarding may fail if there is no adjacent node closer to destination then transmitting node, this phenomenon call local maxima.

ii) Restricted directional flooding:

In restricted directional flooding, with the knowledge of destination node's location, source node determines a geometrical forwarding region containing source node and destination node. Source node then broadcast the packet (control or data) in its neighbourhood; nodes within the forwarding region only participate in packet forwarding process. Packet travelled in forwarding region eventually arrive at destination node.

iii) Hierarchical forwarding:

In this scheme network is organized in hierarchy that is, a node forwards packet using non-location aware routing in local topology and greedy forwarding for outside the locality. Network scalability is an advantage of hierarchical forwarding.

2.2.2.1 Protocol based on Greedy forwarding

Following location based routing protocols are based on Greedy forwarding

a) *Most forward within r (MFR)*

In *most forward within r* protocol [41], a given node selects a node in its transmission range that is most forward in direction of destination node. This protocol try to minimise the number of hops a packet has to visit between source node to destination node. This protocol is illustrated in Fig. 2.11 where S and D are the source and destination node respectively and circle with radius *r* representing the transmission range of source node. In MFR source node selects node E as next forwarding node as E is the most forward node in direction of destination node D. MFR performs good in the scenario where a sender cannot adjust its transmission range to destination node.

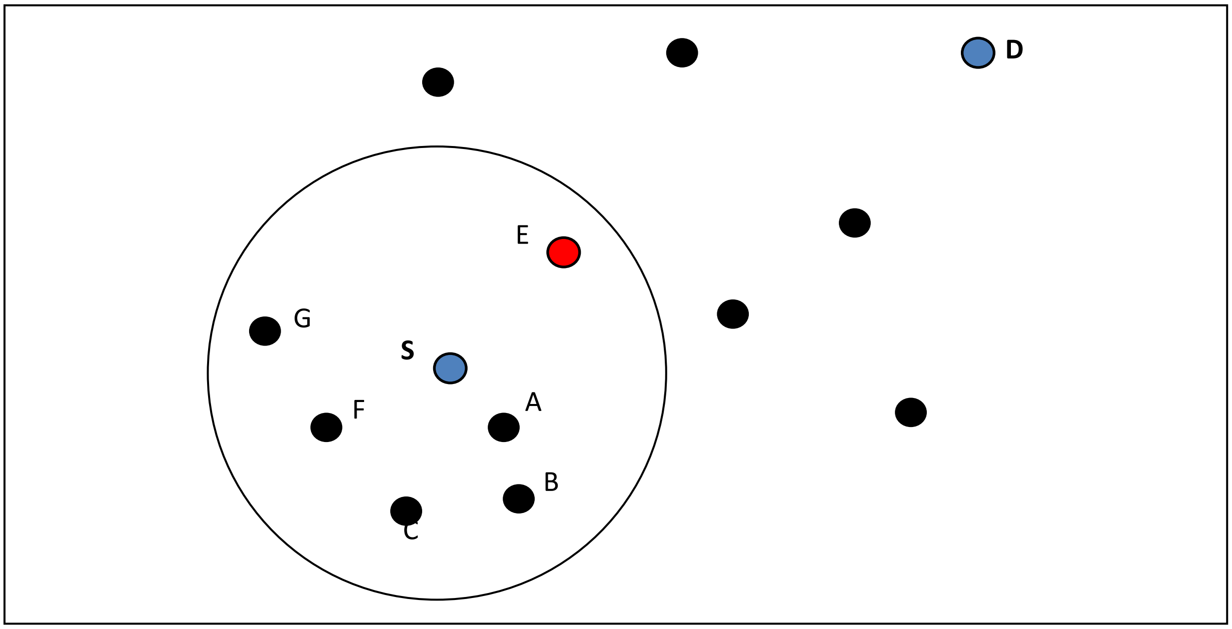


Figure 2.11: Most forward within r

Nearest within forwarding progress based on greedy forwarding is discussed in next sub section.

b) *Nearest within Forwarding Progress (NFP)*

In *nearest within forwarding progress* protocol, a sending node selects next forwarding node in its transmission range that is nearest to sending node and closer to destination node. The advantage of NFP the probability of packet collision reduced considerably. This protocol is illustrated in Fig. 2.12 where source node is S and destination node is D ,circle with radius r representing the radio range of node S. Source node S selects the node B as next forwarding node as B is nearest neighbour of S and closer to D then S.

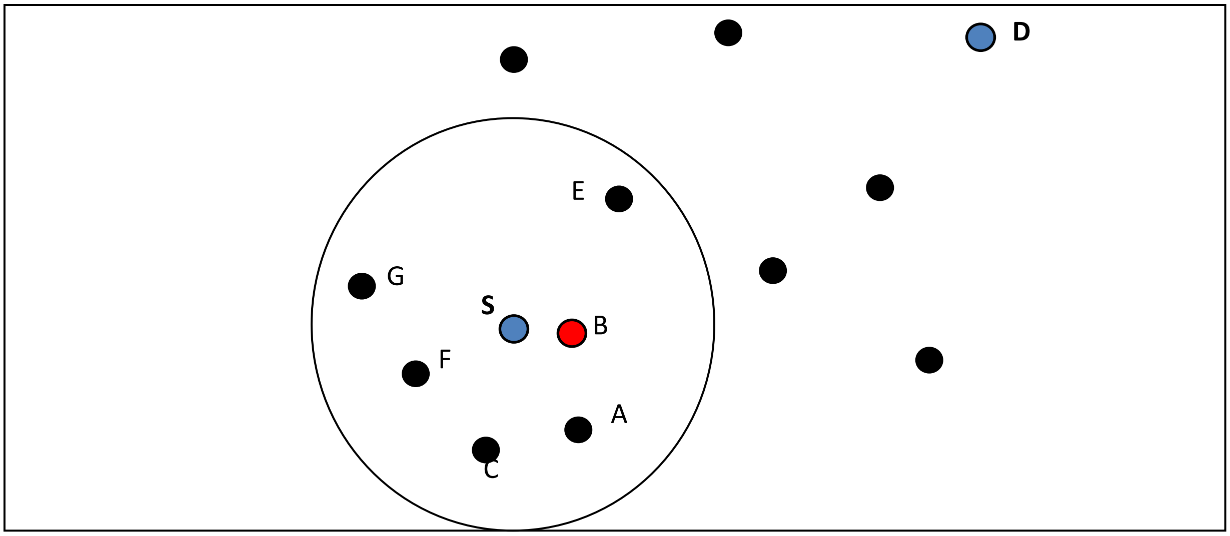


Figure 2.12: Nearest within Forwarding Progress

c) *Compass routing*

In *compass routing* [42]source node determines a straight line connecting source node and destination nod. Sender node selects the next forwarding node that is closest to straight line and closer to destination node then itself. Compass routing illustrated in Fig. 2.13 where source node S select next forwarding node C that is closest to straight line. Compass forwarding tries to minimise the geographical distance that a packet has to pass.

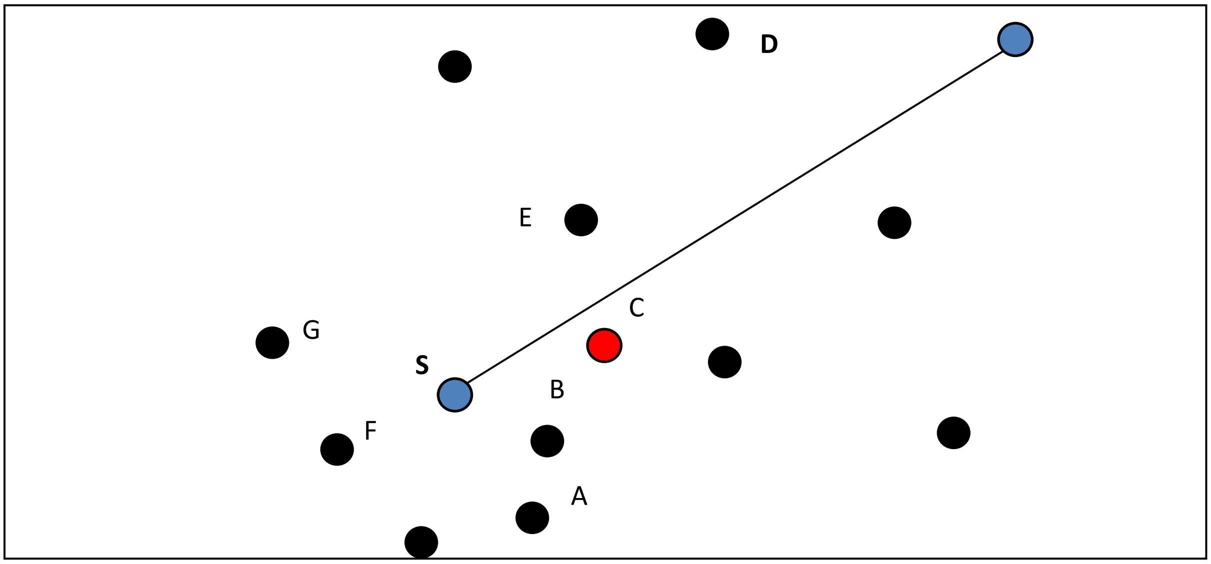


Figure 2.13: Compass routing

d) *Random forwarding*

In *random forwarding* [43] a given node can select next forwarding node randomly which is closer to destination node then itself. Random forwarding minimise the operation overhead to take the packet forwarding decision.

All above discussed greedy forwarding protocol may suffer from local maximum that is greedy forwarding may not be able to find path between source and destination that is actually does exist. As shown In Fig. 2.14 even there is a valid path from source S to destination D source node S unable to find path because S itself is closest to destination node in its radio range and unable to find next forwarding node using greedy forwarding and reaches to local maxima.

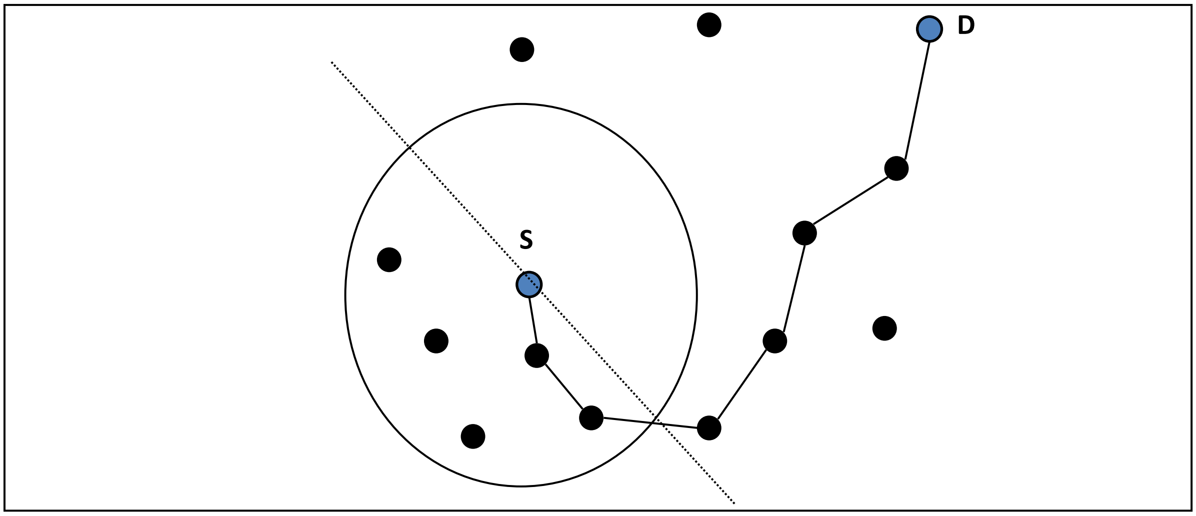


Figure 2.14: local maxima problem

Solution to the problem has been suggested that if sending node unable to locate any node in forward direction it can select a lease backward node as forwarding node[5]. however backward forwarding may enter in looping of packet. Some researchers suggest that drop the packet if local maxima problem arises [44]. Another recovery method Greedy perimeter Stateless Routing Protocol (GPSR) [45] suggested that packet may enter in backward progress (recovery) mode if it reaches to local maximum and return to greedy mode if it found a node closer to destination then the node where it entered in back ward progress(recovery).

2.2.2.2 Protocol based on restricted directional flooding

The terminologies used in restricted directional flooding methods used are discussed below.

Expected region:

Expected region is circular area determined by source node with the knowledge of location information of destination node, where, destination node is expected to be present as shown in Fig. 2.15. As the location information may be out-dated, radius of expected region is calculated by $r = v(t_1-t_0)$, where, r is radius, v is maximum velocity which by a mobile node can travel in network, t_1 is the current time stamp and t_0 is time stamp when source node get the location information of destination node.

Forwarding region:

Forwarding region is a geometric area (depending on protocol) calculated by source node containing source node and expected region as shown in Fig. 2.15. Forwarding region confine the forwarding process from entire network to a small region and reduces the routing overhead considerably.

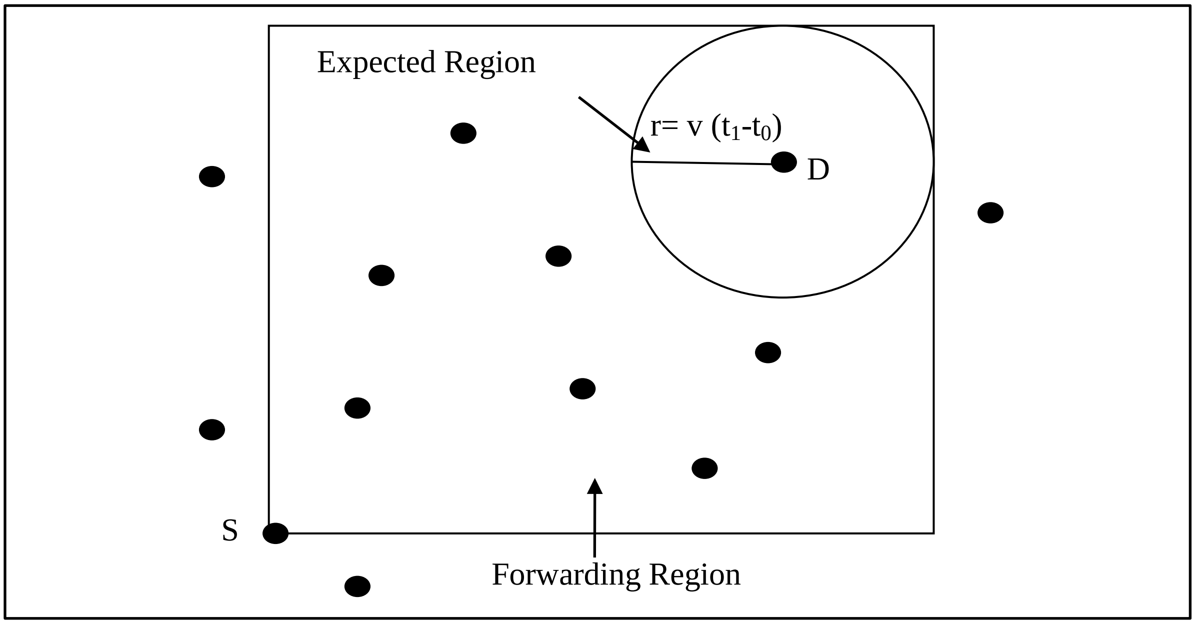


Figure 2.15: Illustration of forwarding region and expected region

a) *Location Aided routing (LAR)*

In location aided routing [46] source node S determines expected region and rectangular forwarding region as shown in Fig. 2.16. Source node S then put this information in packet header and forwards it in its vicinity. On receiving the packet nodes in network determine whether they belongs to forwarding region or not by comparing their coordinates with the information contained in packet header. Nodes within rectangular region only, participate in packet forwarding process. LAR reduces the forwarding region from complete network to small rectangular region which reduces the network control overhead and increases network life.

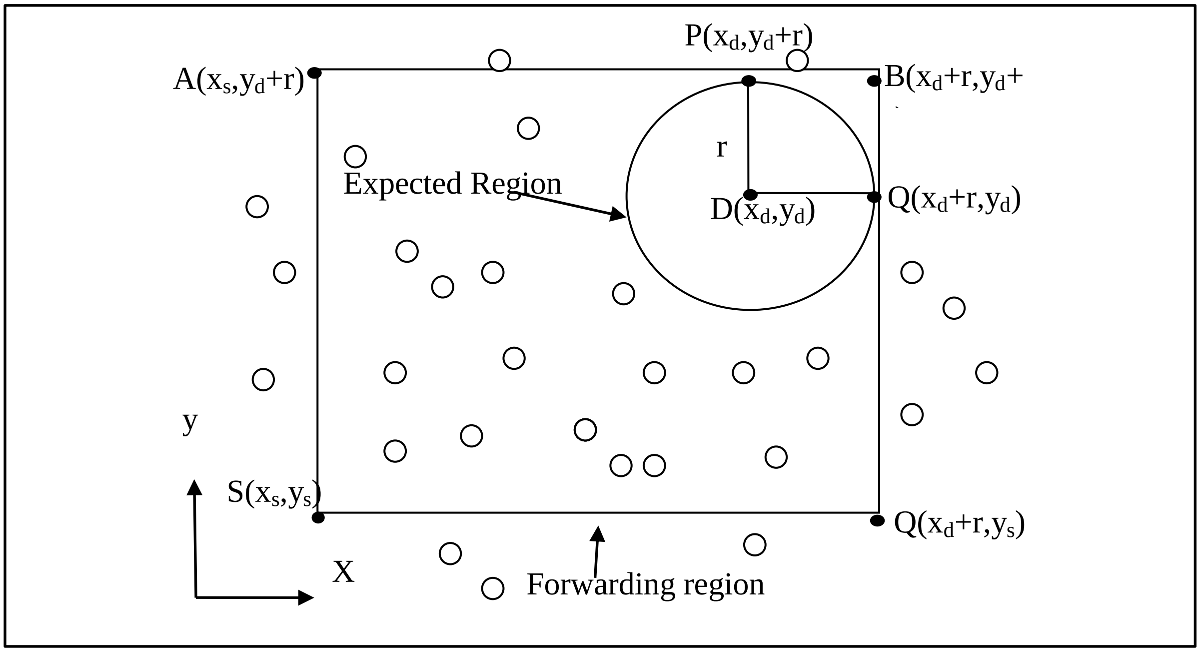


Figure 2.16: Location Aided Routing

Distance Routing Effect Algorithm for Mobility is discussed in next sub section.

b) *Distance Routing Effect Algorithm for Mobility (DREAM)*

In DREAM [47] protocol source node S determines expected region and two tangents with angle α and β as shown in Fig. 2.17. Source node then put this information in header of packet and forward it to its one hop neighbours in direction of destination node D. Neighbouring nodes determine whether they are in direction of destination or not, by comparing their own position with range given by [α-β , α+β]. If nodes are within the range then they forward the packet otherwise drop it. This process is repeated until destination node is arrived.

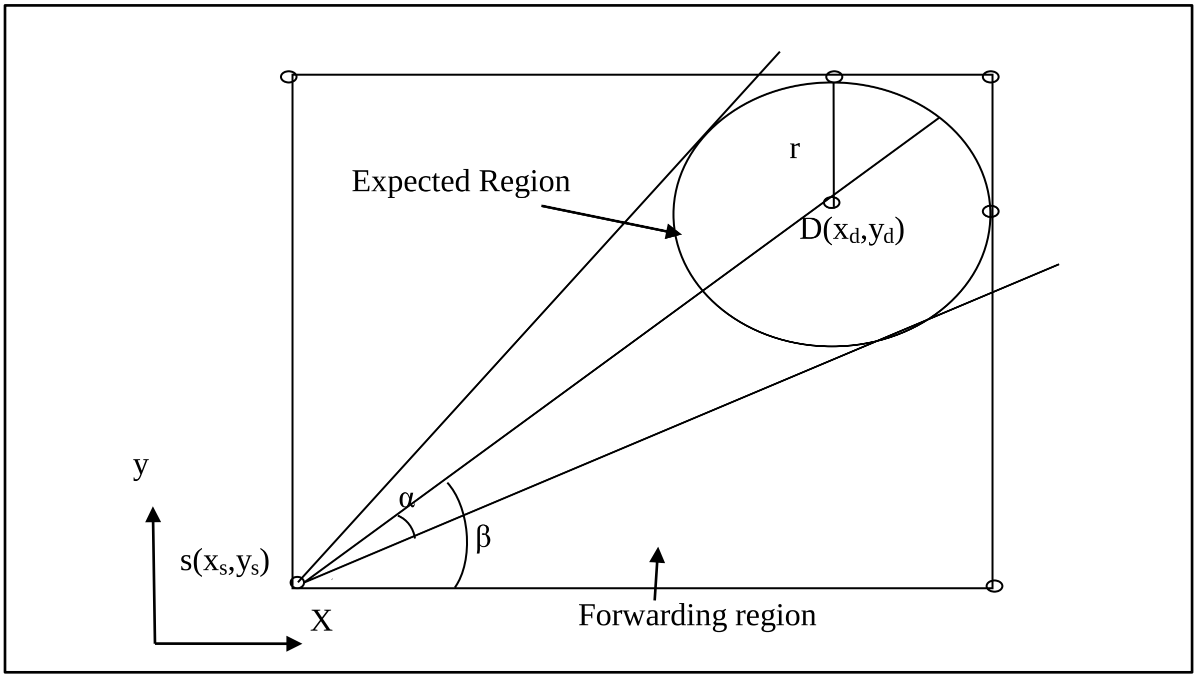


Figure 2.17: Distance Routing Effect Algorithm for Mobility

Greedy Location Aided routing also known improved location aided routing (ILAR) is discussed in next subsection.

c) *Greedy Location Aided routing [GLAR]*

GLAR [48] utilize the both greedy forwarding and restricted directional flooding method .GLAR combines the LAR routing and compass routing as shown in Fig. 2.18. In addition to forwarding region and expected region GLAR determines a straight line connecting to source node S and destination node D called reference line. A sending node selects the next forwarding node in its neighbourhood which is in forwarding region and closest to reference line. Therefore on one hand GLAR restrict the proliferation of route request up to forwarding region on one hand by utilizing the restricted directional flooding and reduces the geographical distance on other hand by utilizing the greedy compass routing. Due to the use of greedy approach, this protocol may suffers from the local maxima problem

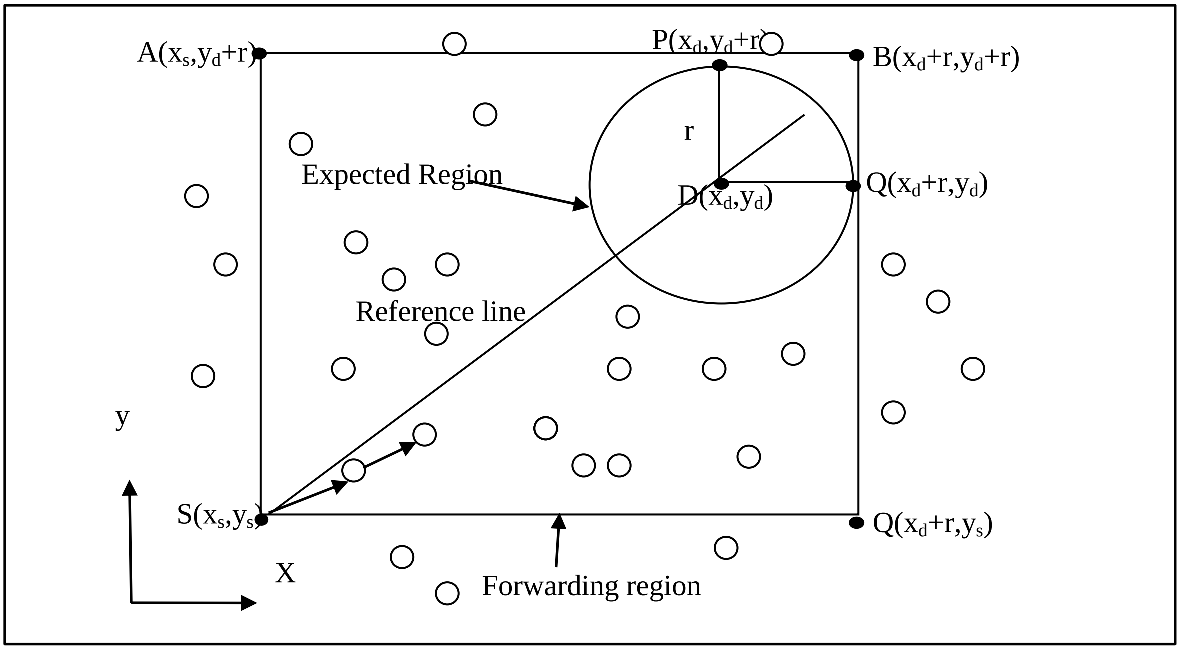


Figure 2.18: Greedy Location Aided routing [GLAR]

2.2.2.3 Protocol Based on Hierarchical forwarding

In an Ad-hoc network at given node routing related calculation can be reduced considerably by introducing the some form of hierarchy. Hierarchical approach also allows a network to scale up to very large number of nodes. Researchers are interested to combine location aided routing with hierarchical approach so that routing overhead can be minimize with scalability of network. Such type of routing mechanisms are discussed below

a) *Terminodes Routing*

Terminodes Routing [49] combines hierarchical routing with location aided routing by establishing two level of hierarchy. A proactive distance vector routing is used for forwarding the packet in local topology of sending node and for long distance a greedy-position based forwarding is used. When the packet reaches at the local topology of destination node, it again enters in proactive routing mode and forwarded by distance vector routing. Terminodes uses a recovery mechanism to counter the local maxima problem in greedy forwarding i.e. sender includes position of some node in packet header. Whenever a node encounter local maximum, it send the packet through listed position back to the sender.

b) *Grid Routing*

Grid Routing [50] is another hierarchical-location based routing which partitions the network area in squares. A flat proactive routing is used for local forwarding [in squares] and a position based routing is used for long distance forwarding of packet. Grid routing exploits the hierarchical approach not only to scale the network but to allow nodes to use of location proxy, if node do not know its own position to participate in location aware routing. The idea of location proxy is that, for every node in network there is at least a position aware node (location proxy) in its local topology. Whenever a position-unaware node wants to communicate, it uses the location proxy as its own address i.e. packet destined to position-unaware node, therefore, arrive at location proxy and then forwarded with proactive routing to position unaware node.

2.2.2.4 Comparison of routing method

In this section a comparison of above discussed protocol has been comprehended in Table 2.5. It is clear that hierarchical strategies GRID and Terminodes are scalable, i.e. these protocol behave better with increasing number of mobile nodes. While other protocols LAR and DREAM have communication complexity O(n) therefore not scalable.

On the other hand LAR and other restricted flooding based protocol reduces the forwarding region to small geometric area which leads to saving of network energy and reduces the control overhead considerably. Also these protocols exhibit high level of robustness against failure of a particular node in path and position incorrectness. Greedy forwarding based protocol are loop free due to selection of next intermediate node in forward direction only. While flooding based routing suffers from looping of packet.

All greedy forwarding based protocols suffer from local maxima problem i.e. if a sending node unable to find forwarding neighbour in direction of destination, some recovery mechanism is needed to counter this problem. GLAR combines compass routing with restricted directional flooding and perform better than its ancestor LAR by reducing the forwarding region as well as geographical distance that a packet has to travel.

Table 2.5: Comaprision of location aided routing methods

Protocol→ Parameter ↓	MFR	NFP	Compass routing	DREAM	LAR	GLAR	Terminodes	GRID
Forwarding Type	Greedy	Greedy	Greedy	RDF	RDF	RDF	Hierarchical	Hierarchical
Scalability	No	No	No	No	No	No	Yes	Yes
Robustness	Average	Average	Average	High	High	High	Average	Average
Forwarding area	Complete network	Complete network	Complet e network	Restricted Triangular	Restricted Rectangular	Restricted Rectangular	Complete network	Complete network
Loop free	Yes	Yes	No	No	No	No	Yes	Yes
Path construction	Single path	Single path	Single path	Flooding	Flooding	Flooding	Single/Multi path	Single/Multi path
Local maxima problem	Yes	Yes	Yes	No	No	No	Yes	Yes

2.3 EXISTING QOS ENABLES ROUTING METHODS FOR WIRELESS AD HOC NETWORK

In recent time users are interested to share heavy data like media file in real time which requires certain Quality of Service [51] here the review of work done by researchers in this direction is presented,

Riedel et al. in [52] invented a ‘QoS cognisant handover procedure for mobile ad-hoc networks’ in the patent publication number US007693093B2. This invention identifies the entire path from source to destination along with associated QoS metric. Thereafter resources are pre-allocated along the candidate path proactively. The packets are routed through the path having best QoS suitability. This ensures the certain level of Quality of Service to suits the user need.

Shojafar et al In [53] proposed a dynamic resource provisioning scheduler for internet based virtualized data centres which have TCP/IP connection to mobile device with limited time and energy, and attempts to provide the Quality of Service and also minimize the energy consumption in processing and communication. Dynamic resource provisioning scheduler achieves dynamic load balancing through online job decomposition.

Hosseini Nazhad et al In [54] proposed a novel Hierarchical Clustering Algorithm (HCAL) and consequent hierarchical routing for Large scale Mobile Ad-Hoc Network (LMANET). The selection of cluster head in HCAL is based on node's weight value which is the function of node's degree and link expiration time to node's neighbours. The node with highest weight value is selected as cluster head. HCAL provides Quality of Service in routing as selected cluster heads forms stable clusters which are scalable to cover large number of nodes.

Suri et al In[55] proposes a Quality of Service enabled Table driven routing protocol which is adaptable to type of traffic need with low-delay, high-reliability, and high-throughput. In NQM every node in network proactively maintains a table which contains available bandwidth and delay of its neighbouring nodes. To provide the Quality of Service along the path from source to destination, appropriate forwarding node is selected which satisfied the required QoS parameter i.e. permissible delay and required bandwidth.

V. V. Mandhare et al in [56] proposed a distributed cache algorithm. In this algorithm source node made aware its neighbors about the erroneous route. Therefore nodes are able to omit the stale routes in advance and ensures the certain QoS parameters.

Seyed Hossein Hosseini Nazhad et al [57] proposed a novel hierarchical clustering mechanism (HCAL) in which link expiration and node degree is the consideration for ensuring the Quality of Service in packet routing and maintenance of cluster structure.

Usman A. D et al [58] proposes a throughput efficient ad-hoc on demand distance vector (TE-AODV) routing protocol to improve the Quality of Service and minimize the control overhead . In TE-AODV every node determine their *Instant Processing State* (IPS) by utilizing the knapsack heuristic. A predefined threshold value of IPS is set. Any node have IPS below that value will not participate in routing of data.

2.4 SUMMERY

Through extensive study of available literature and critical look at the various aspects, the following observations are figured out.

- In location aided routing, a given transmitting node takes the routing decision (selects the next forwarding) only on the basis of location of neighboring nodes with respect to destination node and overlooks the fitness of next forwarding node (QoS). Therefore the established path may contain vulnerable nodes, which leads to frequent link breaks.
- In restricted directional flooding, the *forwarding region* is determined only once by the sender node. Consequently in routing process, control and data packet flooded in same region again and again until the destination node encountered, which leads to higher routing overhead.
- For route maintenance, existing routing protocol follow the complete route reconstruction approach, even for single link failure. Hence effort done in former path construction gets wasted and maintenance cost increased.
- In cluster based routing, due to the involvement in every single communication, originated or destined to their cluster. cluster head becomes overburdened and their battery drain rapidly.
- The ability of selecting the healthy cluster head, weight based cluster-head election process induces a huge calculation overhead on every node in the network, irrespective to their competence of winning the election.
- In the process of cluster formation, standard weight based clustering algorithms distributes the load among the cluster-heads unfairly i.e. some healthy cluster-heads may have assigned the number of members under the capacity and some weak cluster-heads may have assigned over the capacity.

In next chapter, a novel location based routing algorithm QoS Enabled Improved Location Aided routing (QEILA) is presented.

CHAPTER 3

QOS ENABLED IMPROVED LOCATION AIDED ROUTING (QEILA)

3.1 INTRODUCTION

A mobile Ad-hoc Network (MANET) is a collection of mobile nodes which cooperate with each other without any fixed infrastructure for transmission of data from one node of the network to any other node. Every node in MANET has equal hardware and software capabilities therefore, they all can act as sender, receiver or router. Mobile ad-hoc network has applications in various areas like military, battle field, disaster management, rescue operations and scenarios, where the infrastructure setup is not feasible. In addition to these applications, Mobile Ad-hoc network is also applicable in interactive conference lecture and social network in a limited geographic region, where the participants with mobile device like laptop, mobile phone, PDA need to share heavy data like video, image, and voice in real time scenario. For Routing of real time traffic in mobile Ad-hoc network, a reliable path from source to destination with legitimate Quality of Service in terms of bandwidth and battery backup is imperative. As the mobile ad-hoc network consists of battery driven, wireless nodes the topology of network is dynamic and the routing of data with QoS in such situation has to be addressed.

Routing protocols in MANET are classified in to three categories namely *Proactive* (table driven), *Reactive* (on demand) and *Hybrid*. In proactive routing every node maintains route to every other node in network all the time. The advantage of proactive protocol is that the path to every node is available all the time which leads to less delay. Disadvantage of this routing protocol is excessive control and processing overhead. The application of such protocol can be seen in Destination Sequenced Distance Vector Protocol (DSDV) [14]. On the other hand, reactive protocol discovers the path to the destination node whenever necessary, resulting in the reduction of control overhead considerably. The disadvantage to this protocol is transmission delay because sender has to wait while construction of the path to destination node.The application of Reactive protocol can be seen in the Dynamic Source Routing Protocol (DSR) [19][20] and the Ad-hoc On-demand Distance Vector Routing

Protocol (AODV) [21]. Hybrid routing protocols [59][60][61] takes the advantage of both proactive and reactive protocols. A suitable example of Hybrid routing protocol is Zone Routing Protocol (ZRP) [59].

With the availability of tiny GPS sensors, another reactive routing methodology called location aided routing has evolved in recent years [62][63][64]. In location aided routing every node in network is able to determine the geographical position of all the nodes in network and routing decision is based on the location of transmitting and destination node. Location aided routing attempts to minimise the control over head at network level, as only some selected number of nodes participate in route construction and data transmission process rather than entire network.

The main objective of this proposed work is to utilize the location aided routing for constructing the path from source to destination by considering healthy nodes in terms of battery life and bandwidth only, so that the Quality of Service can be achieved.

3.2 BACKGROUND

In Ad-hoc networks low control overhead and Quality of Service are the two desirable considerations. It can be understood from the literature review that the location based routing methodology attempts to minimize the control overhead by limiting the packet forwarding region but overlook the intermediary nodes health while constructing the path.

Transmission of heavy and real time data over wireless links with Quality of Service in Mobile ad-hoc network is a very challenging task. Bandwidth and battery backup parameters are considered to be check for Quality of Service. To provide the Quality of Service in mobile ad-hoc network, aliveness of a transmitting node is a great concern. Location aided routing methods for mobile ad-hoc network have gained popularity because of network’s energy saving. It keeps minimal number of participating nodes for path construction between source and destination, but they do not consider the health of participating nodes that constitute a path. To address this issue QoS Enabled Improved Location Aided routing (QEILA) has been proposed that harness the location aided routing with required QoS parameter check and builds a protocol which provides Quality of Service in terms of required battery life and available bandwidth.

The proposed QoS Enabled Improved Location Aided routing (QEILA) protocol utilizes the location based routing to minimize the control over head on the one hand and enables it with certain QoS parameter check while selecting the next forwarding node to provide the Quality of Service on the other hand. The proposed work takes the Improved Location Aided Routing (ILAR) as a reference protocol and equips it with Quality of Service to determine an efficient way to route the media and real-time traffic. In QEILA transmitting node initializes a table called *Candidate Next Node table* (CNN) (as shown in Table 3.1) and utilizes it while constructing the path and route maintenance during the communication. Candidate next node table contains best *k* nodes, nearer to reference line with Quality of Service detail (available bandwidth and battery life).

Table 3.1 Candidate next node table (CNN)

SN	Node Add	ADIST	TDIST	ABW	ABL
1	-	-	-	-	-

Where :
Node add: IP address of a mobile node
ABW: Available bandwidth
ABL: Available battery life
ADIST and TDIST described in later section

3.3 TERMINOLOGY USED IN QEILA

Expected region: Let a source node be S that wants to communicate to a destination node D, It is also assumed that S is aware about the position of Node D i.e. D’s coordinates ($X_{d,}Y_d$) at time t_0 and t_1. If node S knows the velocity of node D then S can predict the radius of zone in which D can move by $v(t_1 - t_0)$ .The predicted area is called excepted region. Example of an expected region is shown in Fig. 3.1

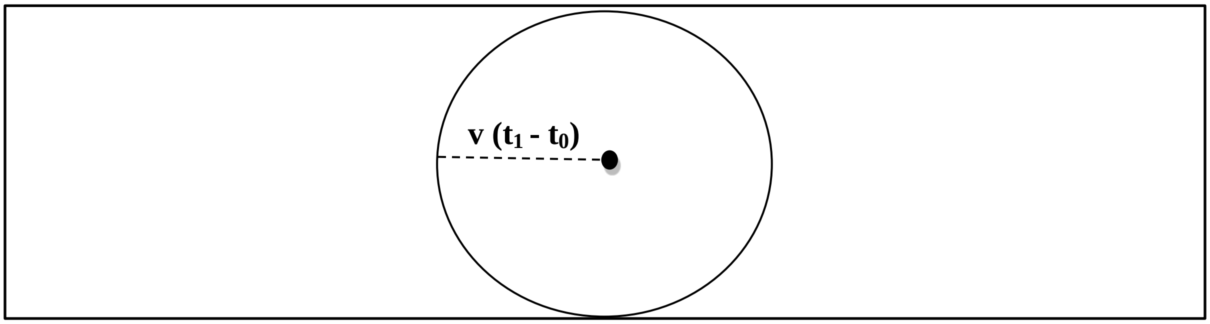


Figure 3.1: Illustration of Expected Region.

***Forwarding region**:* Forwarding region is an area where the request packets are flooded to find a path from source to destination. Let us assume a Source node S(x_s,y_s) is aware about the position of destination node D(x_d,y_d) at time t_0 and S wants to communicate to D and starts path finding routine to connect with D at time t_1. If S is aware about the velocity of node D then node S can anticipate the radius of expected region of D by $r = v(t_1 - t_0)$ where D(x_d,y_d) is positioned at centre. The protocol determines that rectangular forwarding region contains source node S and expected region. Based on expected region, S can determine the coordinates of four corners of forwarding region as shown in Fig. 3.2.

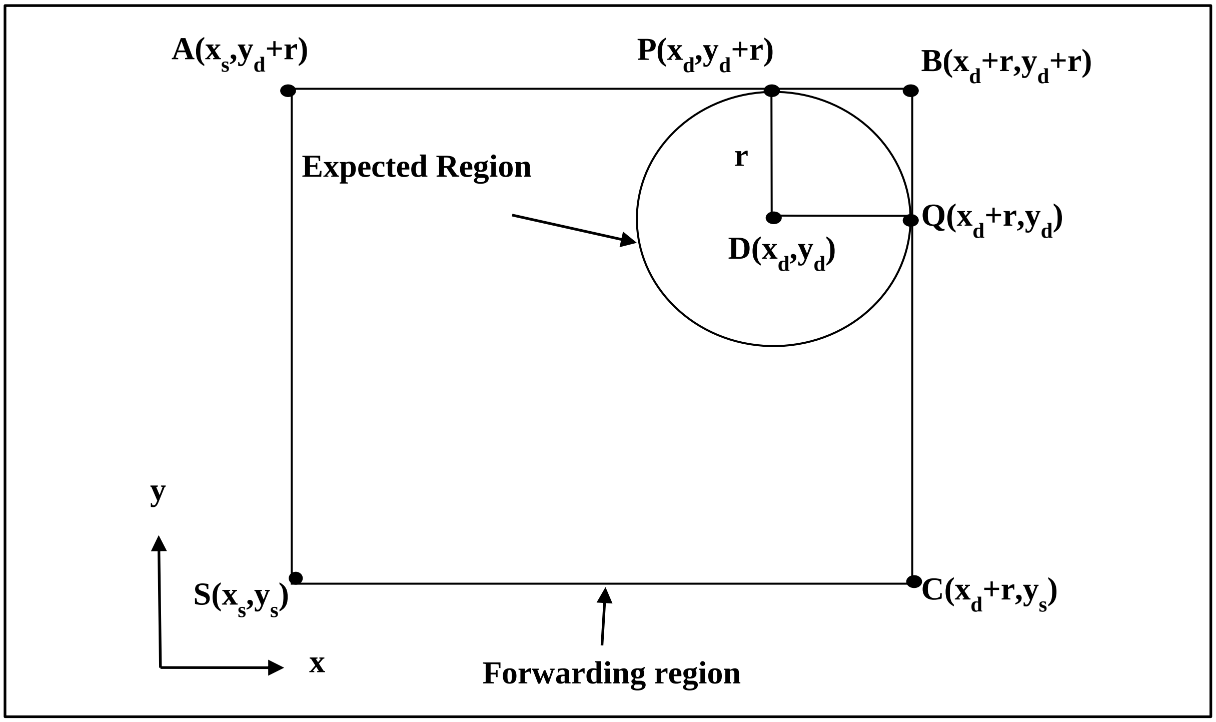


Figure 3.2: Illustration of Forwarding Region

3.4 THE PROCESS

In QEILA the RERQ packet is flooded in forwarding region. Based on the reference line that connects source node S and destination node D the next intermediate node with required QoS parameters will be selected as close as possible to reference line. As shown in Fig. 3.3. It has been assumed that the source node is *S* (x_s, y_s) and that the destination node is *D* (x_d, y_d). So the reference line L connecting these two points can be determined by Equation 3.1

$$(X_d - X_s)(Y-Y_s) - (Y_d - Y_s)(X-X_s) = 0 \qquad (3.1)$$

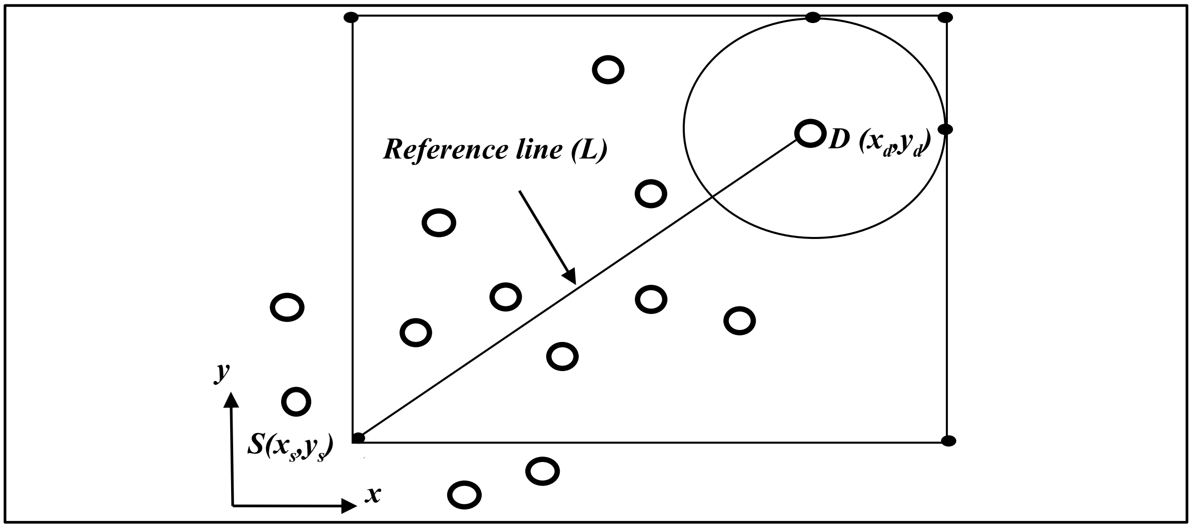


Figure 3.3: Illustration of Reference line L

3.4.1 Path finding

Whenever a node T needs to transmit data to another node for which it has no route entry in its routing table, a path finding routine is invoked. The source node starts the path finding routine by broadcasting a route request (RREQ) packet to all nodes in its neighborhood. Packet format for RREQ is shown in Fig. 3.4.

1	Type	TTL	RB	RBL
2	Source Address			
3	Destination Address			
4	Broadcast ID			
5	Destination sequence number			
6	Source sequence number			
7	Forwarding region information			
8	Source location			
9	Destination location			

RB: Required bandwidth

RBL: Required battery life

TTL: time to live

Figure 3.4: RREQ packet format in QEILA

Upon receiving RERQ, nodes present in T's neighbourhood check whether it belongs to forwarding region or not? If it is not in forwarding region then it discards RERQ, if it belongs, then it replies with revise route request R_RREQ to the transmitting node T. Packet format for R_RREQ is shown in Fig. 3.5. Transmitting node processes R_RREQ from all nodes in its neighbourhood and makes entry in the candidate next node table CNN as shown in Table 3.1 and sort it on ADIST field. Then it selects suitable next forwarding node M with required QoS parameter i.e. ABW>=RB and ABL>=RBL. M reserves the bandwidth for communication by ABW=ABW-RB and performs the broadcasting action. If there is no such node (with required QoS parameter) exists then path finding process stops. Otherwise this process is continued until the RERQ packet arrives at destination node. After receiving the RERQ, destination node ensures the QoS parameters i.e. ABW>=RB and ABL>=RBL. If true then it sends a RREP packet on reverse path and reserves bandwidth for communication by ABW=ABW-RB. Then each node back in the path receives the RREP packet to write the entry into the current routing table. The RREP packet format is shown in Fig. 3.6. Source node then starts sending data. If destination node dose not fulfil the QoS parameter, it discards the RERQ. In addition TDIST (the distance between the transmitting node and node N) value of every CNN entry larger than that of recent transmitting node ensures that the next forwarding node will always be nearer to destination node than transmitting node.

1	TYPE
2	NODEID
3	ADIST
4	TDIST
5	ABW
6	ABL

Figure 3.5: R_RREQ Frame format

1	Type	TTL
2	Destination ID	
3	Destination sequence number	
4	Source ID	
5	Source sequence number	

Figure 3.6: RREP packet format.

Where:

ABW: Available bandwidth
ABL: Available battery life
TDIST$_N$: the distance between the transmitting node and node N. and
ADIST$_N$: the Absolute distance from node N to the Reference line.

ADIST can be calculated by equation 3.2 and shown in Fig. 3.7. If the line passes through two points $P_1=(x_1,y_1)$ and $P_2=(x_2,y_2)$ then the distance of (x_0,y_0) from the line is:

$$\mathrm{ADIST}_N=\frac{|(Y_2-Y_1)X_0-(X_2-X_1)Y_0+X_2Y_1-Y_2X_1|}{\sqrt{(Y_2-Y_1)^2+(X_2-X_1)^2}} \qquad (3.2)$$

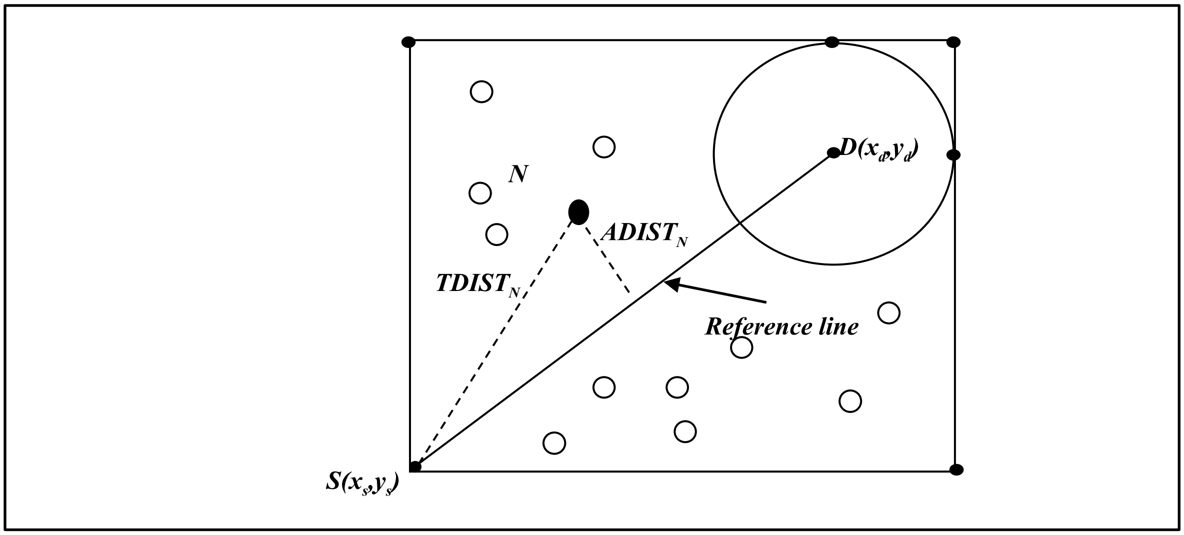


Figure 3.7: Illustration of TDIST and ADIST

The route request RREQ can be spotted uniquely with the combination of <Source ID, Broadcast ID>.Whenever the transmitting node broadcasts a new RREQ, the broadcast ID gets changed by incrementing it. When RREQ arrives at intermediate node it ensures that this RERQ is new with new Source ID and new Broadcast ID. If not that means node has already been processed this RERQ with same (Source ID, Broadcast ID) and discards it.

While path finding it may be the case that RERQ reaches to a node that already have path (route entry in routing table) from itself to destination node. In that case node ensures the freshness of path with the help of destination sequence number. It compares destination sequence number in RERQ with destination sequence number in its own table. If RERQ’s destination sequence number is less than node’s destination sequence no and if node has not processed this RERQ, then it replies with RREP back to the transmitting node from which it received the RERQ, otherwise it rebroadcast the RERQ. Suppose source node S wants to communicate with destination node D, S broadcast a RERQ in its neighbourhood and finds that node in neighbourhood such as A,B,C are in forwarding region. On receiving the RERQ node A,B,C reply with R_RREQ to the source node. The transmitting node then processes these all R_RREQ and find a next transmitting node A (with the help of CNN table) which is the nearest node to Reference line that fulfils the QoS criteria i.e. RB, RBL. Similarly, node A continues the broadcasting RERQ and follow the same process and finds node G. This process continues till the route to destination has been discovered.

The algorithm for path finding is presented with necessary assumptions in Fig. 3.8

```
Algorithm : PATH FINDING ROUTINE
Let total number of nodes in network is n Assume that the source $S_i$ wants to find a path to
destination $D_j$ and broadcasts the RERQ and $I_t$ received RERQ Where $1\leq i,j,t\leq n$ and $j\neq i$.
if (node $I_t$ is in the Request region)
{
    if (node $I_t$ is the destination node $D_j$ )
    {
           If(node  $I_t$ AB>=RB && RBL>=ABL)
             {
      (1) Node $I_t$ sends a RREP packet on reverse path.
      (2) Node $I_t$ Reserves  bandwidth for communication
           by ABW=ABW-RB
      (3) Each node receives the RREP packet to write the entry into the current routing table.
      (4) Node $S_i$starts to send data.
              }
            else
             Node $I_t$ discards the RREQ packet
}
else
{
(1) Node  $I_t$ forwards a R_RREQ packet to the transmitting node.
(2) The transmitting node process the all R_RREQ and perform following step .
     2.1 Made entry in CNN table with each R_RREQ.
    2.2 Sort the CNN table in ascending order of ADIST.
    2.3 M=findbest(CNN)
         If(M!=NULL)
            Node M Reserves bandwidth for communication AB=ABW-RB
           M perform broadcasting action
           else
          Node $I_t$ discards the RREQ packet
}
else
Node $I_t$ discards the RREQ packet
ALGO findbest(CNN)
{
     For(i=0 to k)  //k  is size of CNN table
     {
       If (CNN[i].AB>=RB&& CNN[i]>=RBL&&TDIST >= Distance of transmitting  node )
             return CNN[i]
     }
 return  NULL
}
```

Figure 3.8 : Algorithm for path finding in QEILA

3.4.2 Route Maintenance

In this work a novel approach for local route repair has been proposed in which node that finds broken link utilizes its CNN table which was created at the time of path finding. As mention earlier that at an intermediate node CNN table maintains the list of next candidate nodes with QoS parameter. Whenever an intermediate node finds broken link it deletes the entry of abandoned next node from its CNN table and searches again for suitable alternative node with QoS parameter in its CNN table (likelihood of presence of alternative node is very high). If such node is found in table then it can unicast the RERQ to that alternative node otherwise it rebroadcasts RERQ in its neighbourhood to repair the path which reduces control overhead considerably . Moreover for freshness of table, it updates the ABL field for all nodes by ABL=ABL-µ where µ is average power loss.

The algorithm for path repair with necessary assumptions is given in Fig. 3.9

```
ALGORITHM : PATH REPAIR

Assume that the source node S communicating with destination node D and intermediate
node I found a broken link .
if (I found broken link )
{
1.I deletes entry of abandoned node from CNN table ;
2.M=findbest(CNN)
        If(M!=NULL)
          {
            Unicast RERQ to M;
            }
Else
     rebroadcast RERQ in neighbourhood;
}
```

Figure 3.9: Algorithm for path repair QEILA

3.4.2.1 Illustration Route Maintenance

Let us assume that link between node G and H was broken as shown in Fig. 3.10. When node G finds that link between node H and itself is broken it consults its CNN table and finds an alternate node I (in sorted order) suitable with QoS and repairs the path locally as shown by dashed line in Fig 3.10

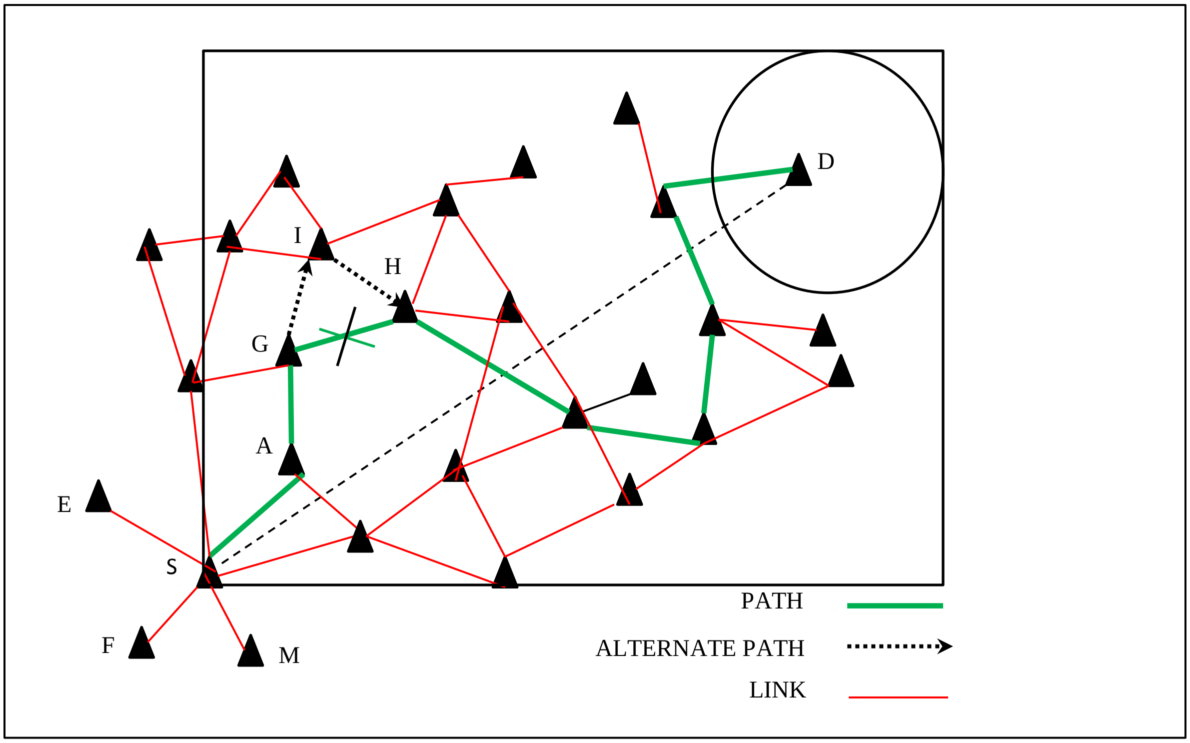


Figure 3.10: Illustration route maintenance.

3.5 RESULT AND DISCUSSION

The existing Improved Location Aided Routing (ILAR) and proposed protocol QoS Enabled Improved Location Aided routing (QEILA) have been simulated using Matlab 7.8.0. 20 mobile nodes were deployed and labelled from 1 to 20. Simulation area has been taken 200 X200 m^2. Available bandwidth at every node varies from 0 to 10 Mbps and required bandwidth and radio range has been taken as 5 mbps and 40 m respectively. Available battery life at every node varies from 0 to 100 % and required battery life has been taken as 50%.

Path construction in both the protocols is simulated for identical network topology and same source-destination pair i.e. source node is 4 and destination node is 17. Simulation stages for path construction of both the protocols ILAR and QEILA has been recorded and presented in Fig. 3.11 (a-h) and Fig. 3.12 (a-h) respectively. Simulation findings for ILAR and QEILA are tabularized with parameter values and comprehended in Table 3.2 and Table 3.3 respectively.

Simulation stages of path construction progression in ILAR, are shown in fig. 3.11(a-h), where source node 4 seeks the path to destination node is 17.

Stage 1: An arbitrary network topology with source node 4 and 17 as destination, as shown in Fig. 3.11 (a)

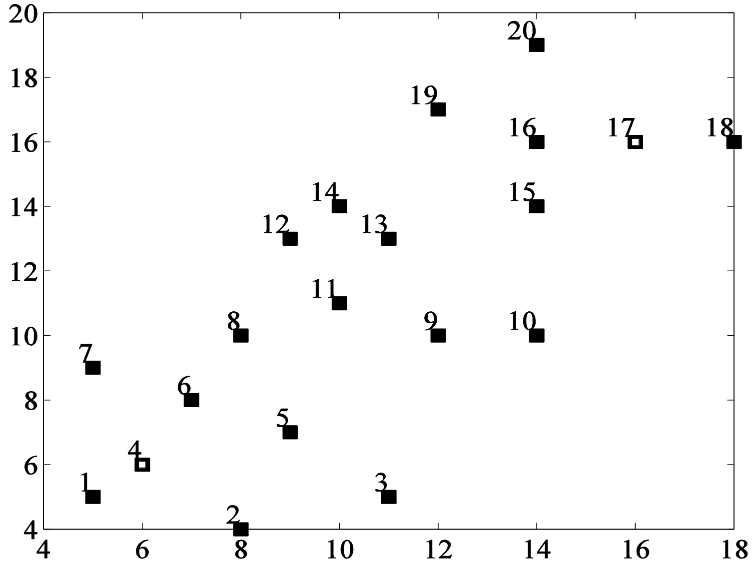


Figure 3.11(a): Stage 1 of ILAR

Stage 2: Source 4 node determines the rectangular forwarding region, as shown in Fig. 3.11(b)

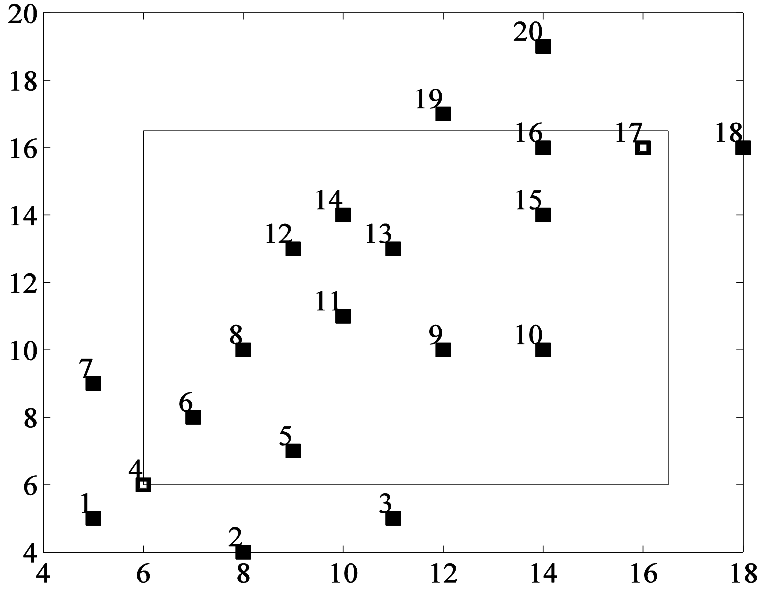


Figure 3.11 (b): Stage 2 of ILAR

Stage 3: Source node 4 determines the reference line to destination node 17 and locates the next forwarding node 6 nearest to the reference line as shown in Fig. 3.11(c)

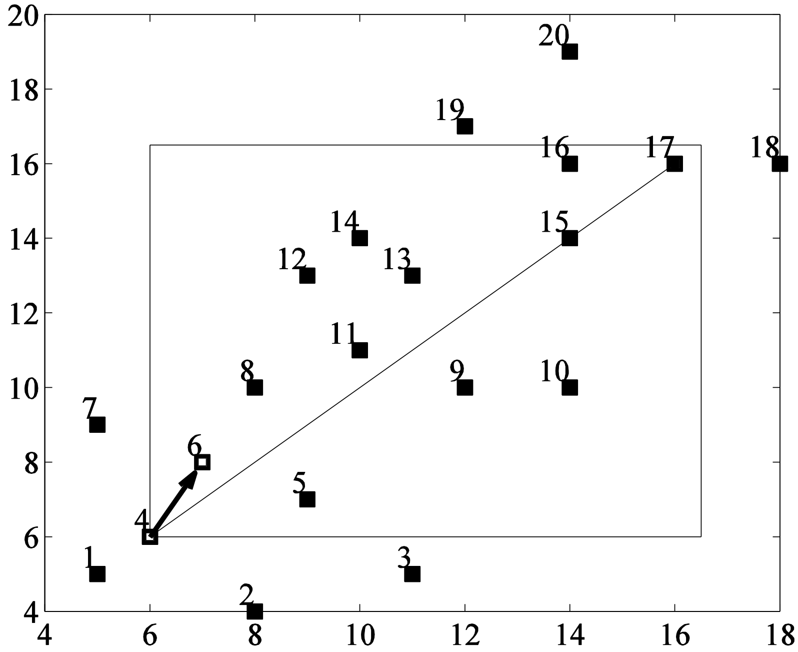


Figure 3.11 (c): Stage 3 of ILAR

Stage 4 : Transmitting node 6 locates the next forwarding node 8 nearest to reference line as shown in Fig. 3.11(d)

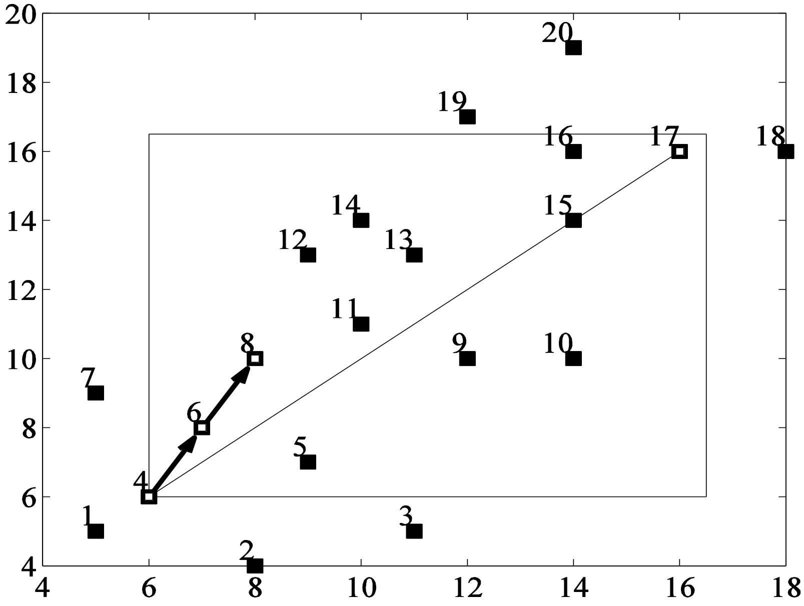


Figure 3.11(d): Stage 4 of ILAR

Stage 5: Transmitting node 8 locates the next forwarding node 11 nearest to reference line as shown in Fig. 3.11(f)

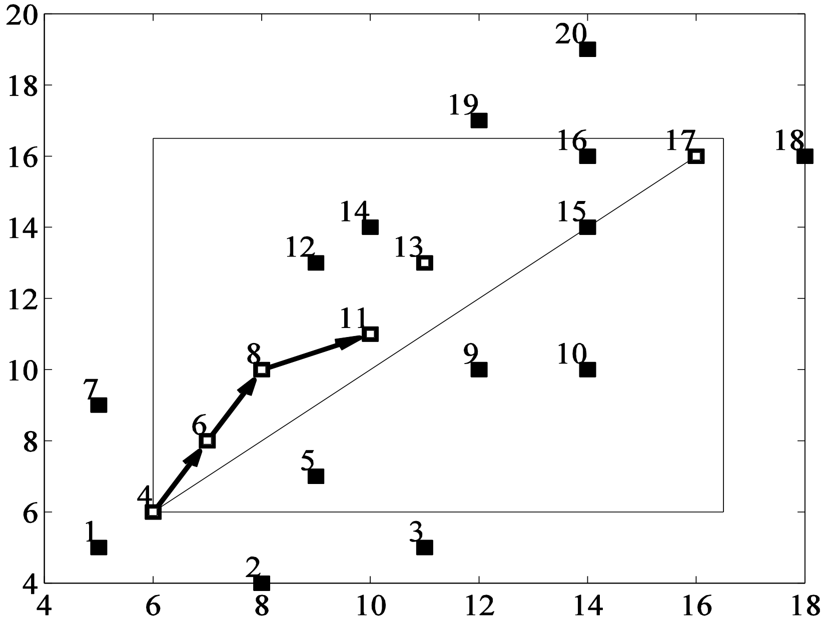


Figure 3.11(e): Stage 5 of ILAR

Stage 6: Transmitting node 11 locates the next forwarding node 13 nearest to reference line as shown in Fig. 3.11(f)

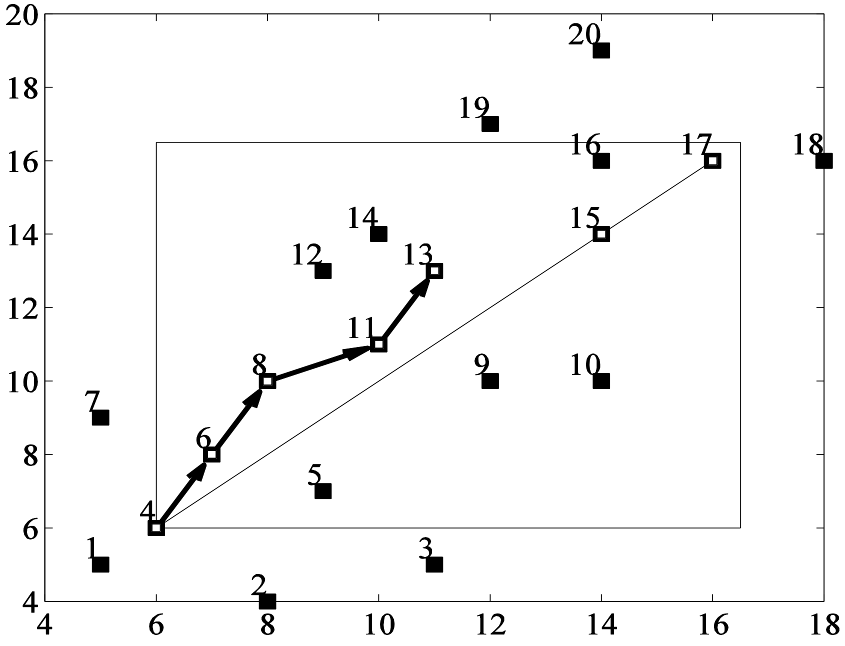


Figure 3.11 (f): Stage 6 of ILAR

Stage 7: Transmitting node 13 locates the next forwarding node 15 nearest to reference line as shown in Fig. 3.11(g)

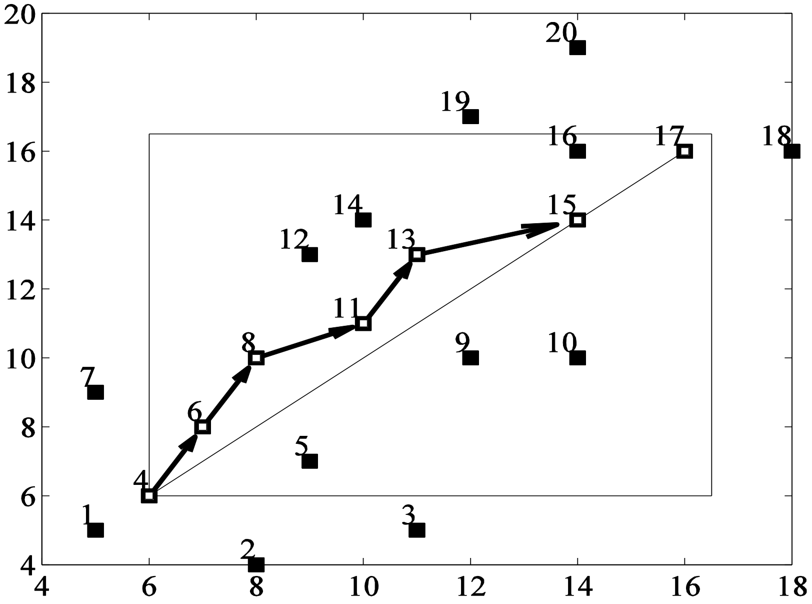


Figure 3.11 (g): Stage 7 of ILAR

Stage 8: Transmitting node 15 locates the destination node 17 then, node 17 respond with RREP packet that traverse revers path, and path gets established, as shown in Fig. 3.11(h)

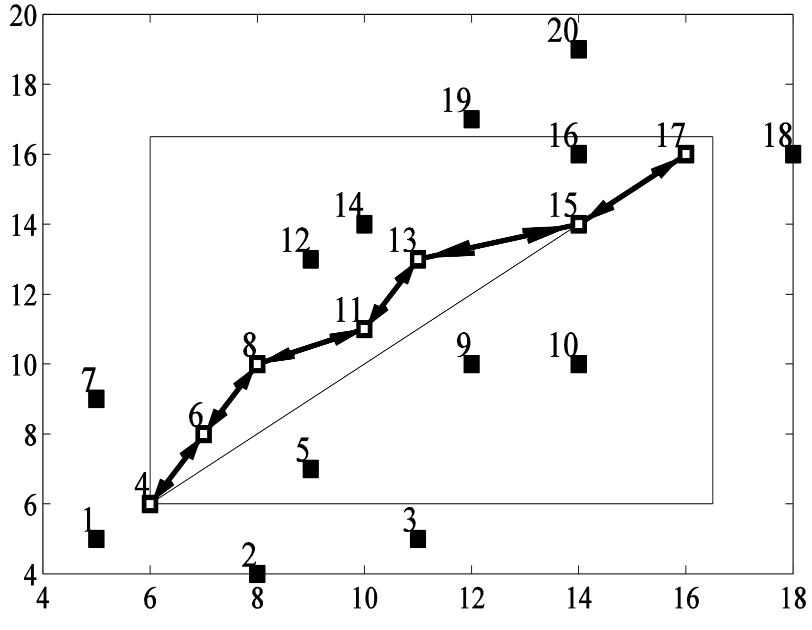


Figure 3.11 (h): Stage 8 of ILAR

Simulation findings of path construction stages in ILAR with various concerning parameters are recorded and tabulated in Table 3.2

Table 3.2: Simulation findings of path construction using ILAR

T.N.	T.N._ Neighbours's _Parameters						N.F.	NF_parameter	
4	T's Neighbours	5	6				6	NF_energy_	45
	Neighbours 's _energy_In_%	60	45						
	Neighbours 's _Bandwidth_In_Mbps	5.9	4.0					NF_Bandwidth	4.0
	Neighbours 's _Adist_unitmeter	1.4	0.7					NF_ Adist	0.7
6	T's Neighbours	5	8				8	NF_energy	80
	Neighbours 's _energy_In_%	60	80					NF_Bandwidth	4.0
	Neighbours 's _Bandwidth_In_Mbps	5.9	4.0					NF_ Adist	1.4
	Neighbours 's _Adist_unitmeter	1.4	1.4						
8	T's Neighbours	5	6	11	12		11	NF_energy	56
	Neighbours 's _energy_In_%	60	45	56	52			NF_Bandwidth	5.4
	Neighbours 's _Bandwidth_In_Mbps	5.9	5.6	5.4	5.0			NF_ Adist	0.7
	Neighbours 's _Adist_unitmeter	1.4	0.7	0.7	2.8				
11	T's Neighbours	8	9	12	13	14	13	NF_energy	76
	Neighbours 's _energy_In_%	80	55	52	76	51		NF_Bandwidth	5.9
	Neighbours 's _Bandwidth_In_Mbps	4.0	5.4	5.0	5.9	5.4		NF_ Adist	1.4
	Neighbours 's _Adist_unitmeter	1.4	1.4	2.8	1.4	2.8			
13	T's Neighbours	9	11	12	14	15	15	NF_energy	42
	Neighbours 's _energy_In_%	55	56	52	51	42		NF_Bandwidth	3.9
	Neighbours 's _Bandwidth_In_Mbps	5.4	5.4	5.0	5.4	3.9		NF_ Adist	0
	Neighbours 's _Adist_unitmeter	1.4	0.7	2.8	2.8	0			
15	T's Neighbours	13	16	17			17	NF_energy	90
	Neighbours 's _energy_In_%	76	51	90				NF_Bandwidth	7.8
	Neighbours 's _Bandwidth_In_Mbps	5.9	6.1	7.8				NF_ Adist	0
	Neighbours 's _Adist_unitmeter	1.4	1.4	0					
17	-	-	-	-	-		-	-	-

T.N.: Transmitting Node N.F.: Next Forwarding Node

Simulation stages of path construction progression in QEILA are shown in fig. 3.12(a-h), where source node 4 seeks the path to destination node 17.

Stage 1: An arbitrary topology of network with source node 4 and destination node 17 as shown in Fig. 3.12(a)

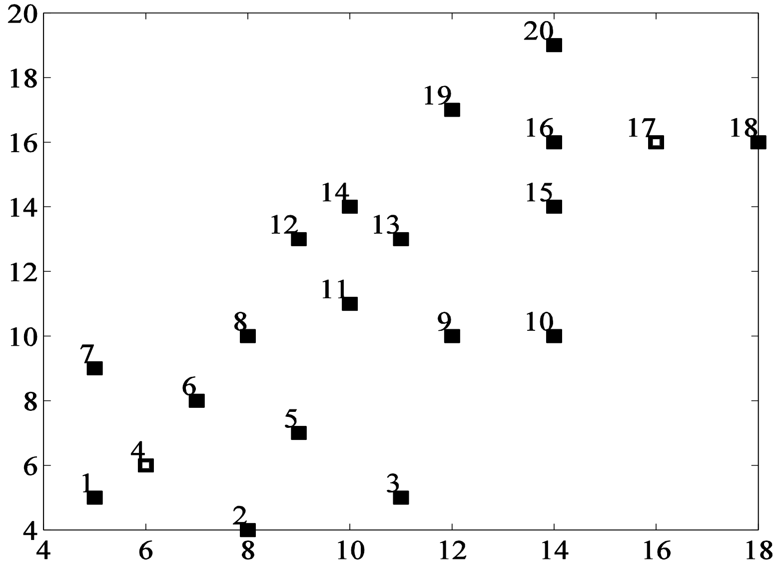


Figure 3.12 (a): Stage 1 of QEILA

Stage 2: Source node determines the rectangular forwarding region as shown in Fig. 3.12(b)

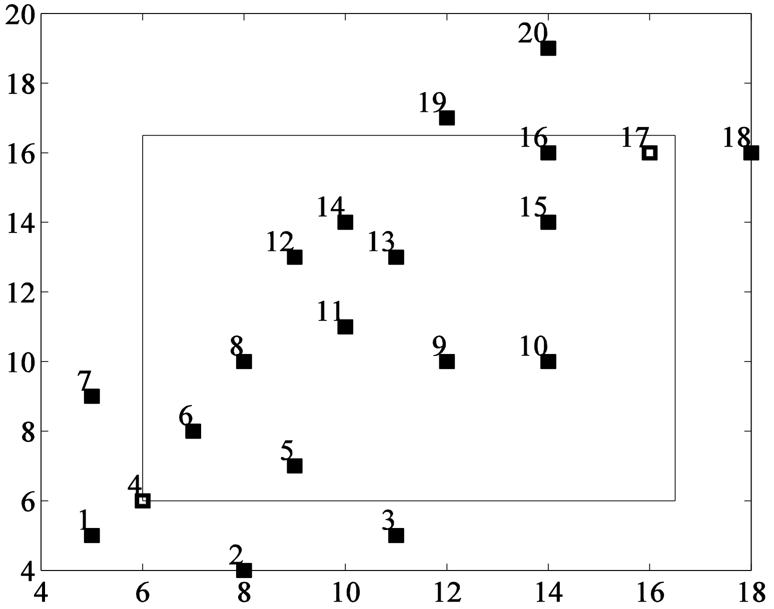


Figure 3.12 (b): Stage 2 of QEILA

Stage 3: Source node 4 determines the reference line to destination node 17 as shown in Fig. 3.12(c)

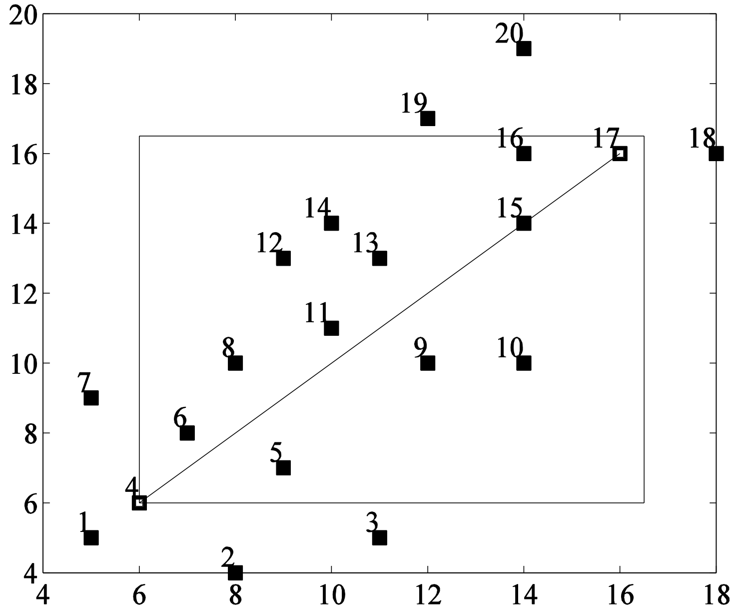


Figure 3.12 (c): Stage 3 of QEILA

Stage 4: Transmitting node 4 locates the next forwarding node 5 satisfying the QoS parameter as shown in Fig. 3.12(d).

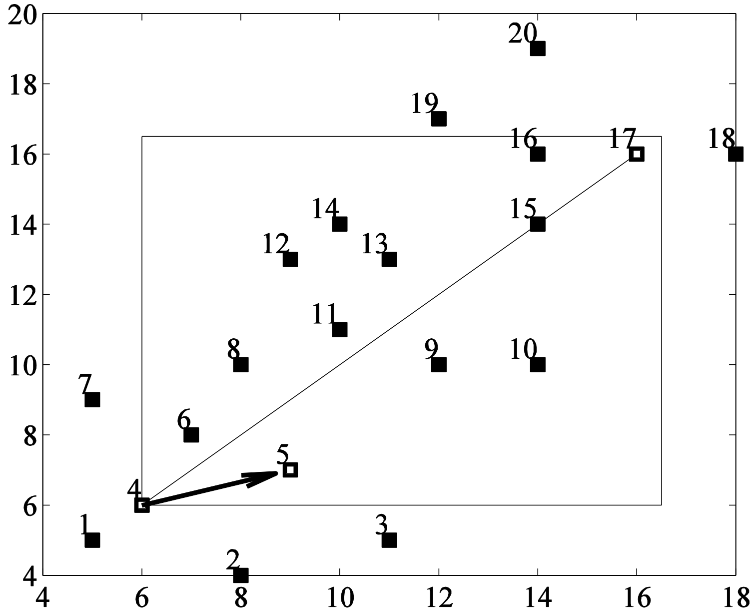


Figure 3.12 (d): Stage 4 of QEILA

Stage 5: Transmitting node 5 locates the next forwarding node 11satisfying the QoS parameter then as shown in Fig. 3.12(e).

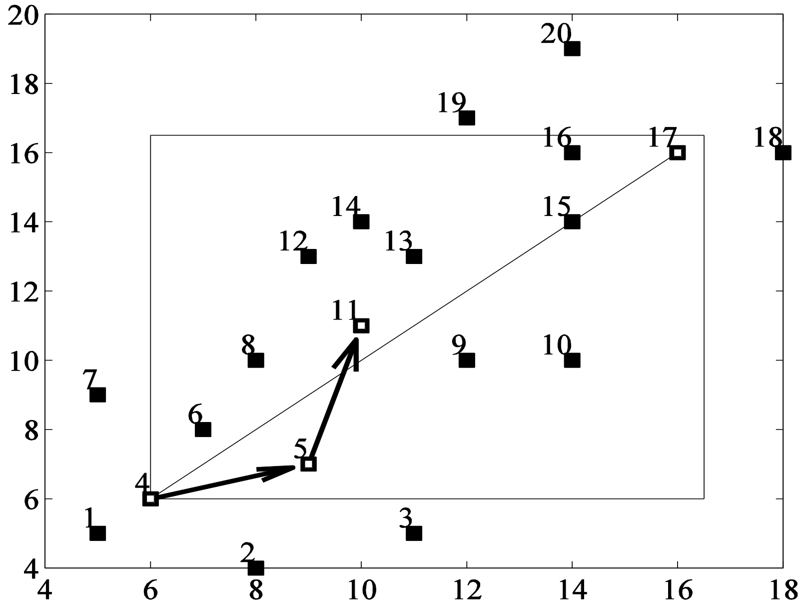


Figure 3.12 (e): Stage 5 of QEILA

Stage 6: Transmitting node 11 locate the next forwarding node 13 satisfying the QoS parameter as shown in Fig. 3.12(f).

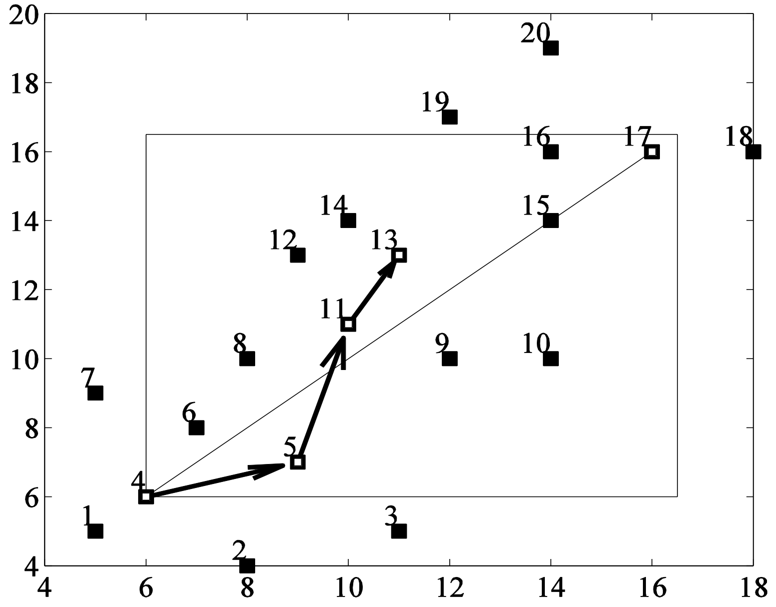


Figure 3.12 (f): Stage 6 of QEILA

Stage 7, Transmitting node 13 locates the next forwarding node 16 satisfying the QoS parameter as shown in Fig.3.12(g).

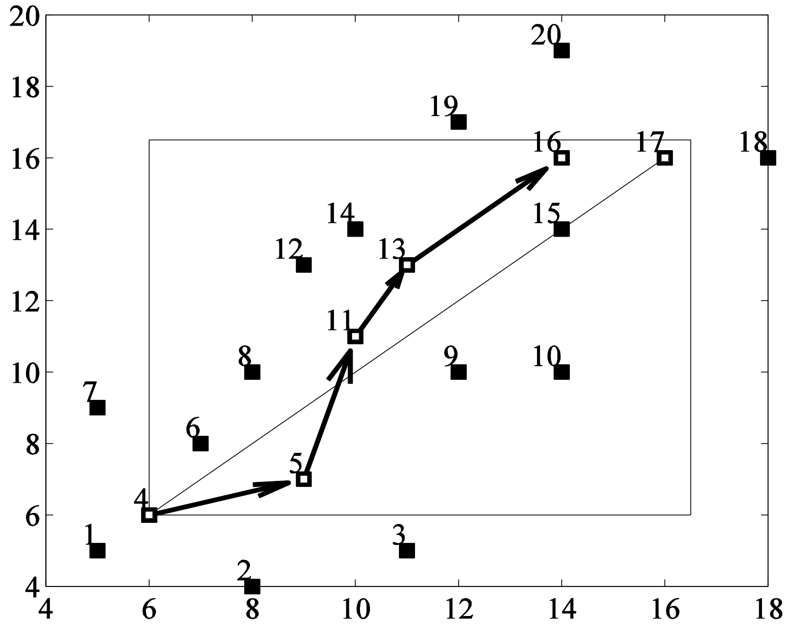


Figure 3.12 (g): Stage 7 of QEILA

Stage 8, Transmitting node 16 locates the destination node 17 then node 17 respond with RREP which travels the reverse path back to the sender 4 and path gets established as shown in Fig.3.12 (h)

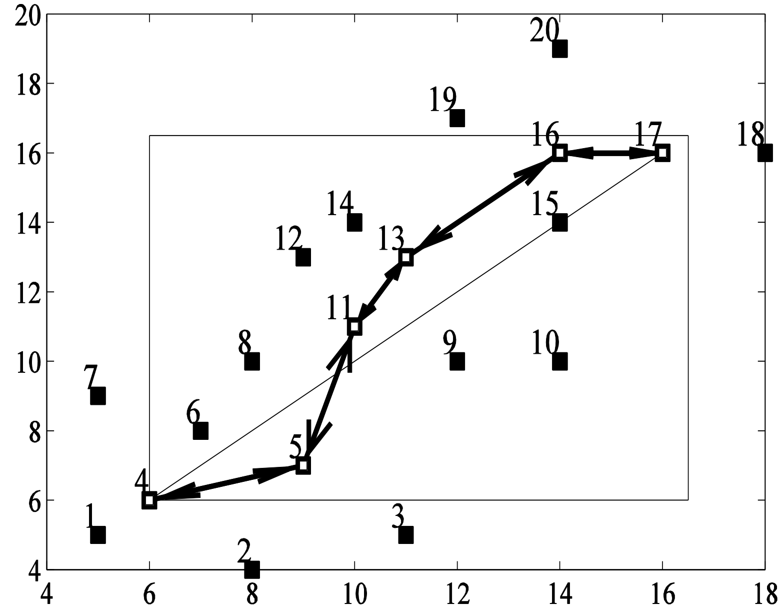


Figure 3.12 (h): Stage 8 of QEILA

Simulation findings of path construction stages in QEILA with various concerning parameters are recorded and tabulated in table 3.3

Table 3.3: Simulation findings of path construction using QEILA

<table>
<tr><th>T.N.</th><th colspan="9">T.N._ Neighbours’s _Parameters</th><th>N.F</th><th colspan="2">N.F._parameter</th></tr>
<tr><td rowspan="4">4</td><td>T’s Neighbours</td><td colspan="4">5</td><td colspan="4">6</td><td rowspan="4">5</td><td rowspan="2">NF_energy_</td><td rowspan="2">60</td></tr>
<tr><td>Neighbours ‘s _energy_In_%</td><td colspan="4">60</td><td colspan="4">45</td></tr>
<tr><td>Neighbours ‘s _Bandwidth_In_Mbps</td><td colspan="4">5.9</td><td colspan="4">4.0</td><td>NF_Bandwidth</td><td>5.9</td></tr>
<tr><td>Neighbours ‘s _Adist_unitmeter</td><td colspan="4">1.4</td><td colspan="4">0.7</td><td>NF_ Adist</td><td>1.4</td></tr>
<tr><td rowspan="4">5</td><td>T’s Neighbours</td><td colspan="3">6</td><td colspan="2">8</td><td colspan="2">9</td><td>11</td><td rowspan="4">11</td><td>NF_energy</td><td>56</td></tr>
<tr><td>Neighbours ‘s _energy_In_%</td><td colspan="3">45</td><td colspan="2">80</td><td colspan="2">55</td><td>56</td><td>NF_Bandwidth_</td><td>5.4</td></tr>
<tr><td>Neighbours ‘s _Bandwidth_In_Mbps</td><td colspan="3">4.0</td><td colspan="2">7.8</td><td colspan="2">5.4</td><td>5.4</td><td rowspan="2">NF_ Adist</td><td rowspan="2">0.7</td></tr>
<tr><td>Neighbours ‘s _Adist_unitmeter</td><td colspan="3">0.7</td><td colspan="2">1.4</td><td colspan="2">1.4</td><td>0.7</td></tr>
<tr><td rowspan="4">11</td><td>T’s Neighbours</td><td>5</td><td>6</td><td>8</td><td>9</td><td>10</td><td>12</td><td>13</td><td>14</td><td rowspan="4">13</td><td>NF_energy</td><td>76</td></tr>
<tr><td>Neighbours ‘s</td><td>60</td><td>45</td><td>80</td><td>55</td><td>78</td><td>52</td><td>76</td><td>51</td><td>NF_Bandwidth</td><td>5.9</td></tr>
<tr><td>Neighbours ‘s _Bandwidth_In_Mbps</td><td>5.9</td><td>4.0</td><td>7.8</td><td>5.4</td><td>6.9</td><td>5.0</td><td>5.9</td><td>5.4</td><td rowspan="2">NF_ Adist</td><td rowspan="2">1.4</td></tr>
<tr><td>Neighbours ‘s _Adist_unitmeter</td><td>1.4</td><td>0.7</td><td>1.4</td><td>1.4</td><td>2.8</td><td>2.8</td><td>1.4</td><td>2.8</td></tr>
<tr><td rowspan="4">13</td><td>T’s Neighbours</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>14</td><td>15</td><td>16</td><td rowspan="4">16</td><td>NF_energy</td><td>51</td></tr>
<tr><td>Neighbours ‘s _energy_In_%</td><td>80</td><td>55</td><td>78</td><td>56</td><td>52</td><td>51</td><td>42</td><td>51</td><td>NF_Bandwidth</td><td>6.1</td></tr>
<tr><td>Neighbours ‘s _Bandwidth_In_Mbps</td><td>7.8</td><td>5.4</td><td>6.9</td><td>5.4</td><td>5.0</td><td>5.4</td><td>3.9</td><td>6.1</td><td rowspan="2">NF_ Adist</td><td rowspan="2">1.4</td></tr>
<tr><td>Neighbours ‘s _Adist_unitmeter</td><td>1.4</td><td>1.4</td><td>2.8</td><td>0.7</td><td>2.8</td><td>2.8</td><td>0</td><td>1.4</td></tr>
<tr><td rowspan="4">16</td><td>T’s Neighbours</td><td colspan="3">13</td><td colspan="2">15</td><td colspan="3">17</td><td rowspan="4">17</td><td>NF_energy</td><td>90</td></tr>
<tr><td>Neighbours ‘s _energy_In_%</td><td colspan="3">76</td><td colspan="2">42</td><td colspan="3">90</td><td>NF_Bandwidth</td><td>7.8</td></tr>
<tr><td>Neighbours ‘s _Bandwidth_In_Mbps</td><td colspan="3">5.9</td><td colspan="2">3.9</td><td colspan="3">7.8</td><td rowspan="2">NF_ Adist</td><td rowspan="2">0</td></tr>
<tr><td>Neighbours ‘s _Adist_unitmeter</td><td colspan="3">1.4</td><td colspan="2">0</td><td colspan="3">0</td></tr>
<tr><td>17</td><td>-</td><td colspan="3">-</td><td colspan="2">-</td><td colspan="3">-</td><td>-</td><td>-</td><td>-</td></tr>
</table>

T.N.: Transmitting Node N.F.: Next forwarding Node

Simulation results of proposed protocol QEILA are compared with simulation results of ILAR and comprehended in Table 3.4. Results show that in case of ILAR, some intermediate node for path from source to destination are vulnerable in terms of battery life and available bandwidth as ILAR selects the next forwarding node on the basis of its nearness to reference line only. Whereas in case of QEILA, selected intermediate nodes for path from source to destination node are more reliable and stable in terms of battery life and bandwidth as QEILA selects the next forwarding node which is closest to reference line and fulfils the required Quality of Service parameters i.e. required battery life and available bandwidth.

Table 3.4: Comparison of QEILA and ILAR

ILAR			**QEILA**		
NODE NO	**ABL(%)**	**ABW(mbps)**	**NODE NO**	**ABL(%)**	**ABW(mbps)**
4(S)	51	5.0500	4(S)	51	5.0500
6	45	5.6000	5	60	5.9000
8	80	4.000	11	56	5.4300
11	56	5.4300	13	76	5.9860
13	76	5.9860	16	51	6.1000
15	42	3.9020	17(D)	90	7.8900
17(D)	90	7.8900	-	-	-

Simulation results also reveals that in case of ILAR as shown in Fig. 3.11, path from source node 4 to destination node 17 is 4→6→8→11→13→15→17 in which node 6 and 15 are not reliable because node 6 has available battery life ABL value 45 which is less than required battery life (RBL), node 8 has available bandwidth ABW value 4.0 which is less than require bandwidth (RB), node 15 has available battery life ABL value 42 which is less than required battery life (RBL) and available bandwidth ABW value 3.9020 which is less than require bandwidth (RB). Whereas in case of QEILA path from source node 4 to destination node 17 is 4→5→11→13→16→17 and all intermediate nodes are reliable and stable i.e. all nodes in path have available battery life (ABL) and available bandwidth (ABW) more than require battery life (RBL) and require bandwidth (RB) respectively as shown in Table 3.4 which ensures the Quality of Service.

Based on the results obtained a path reliability factor PRF can be calculated with help of following formula:

$$PRF = \frac{\text{number of reliable nodes}}{\text{total number of nodes in path}}$$

Where PRF: Path reliability factor

Reliable Node: node having ABL and ABW more than or equal to RBL and RB respectively

By putting the obtained values we get:

$$PRF_{QEILA} = \frac{6}{6} = 1$$

$$PRF_{ILAR} = \frac{5}{7} = 0.714$$

A plot of PRF for both the protocols is shown in Fig 3.13

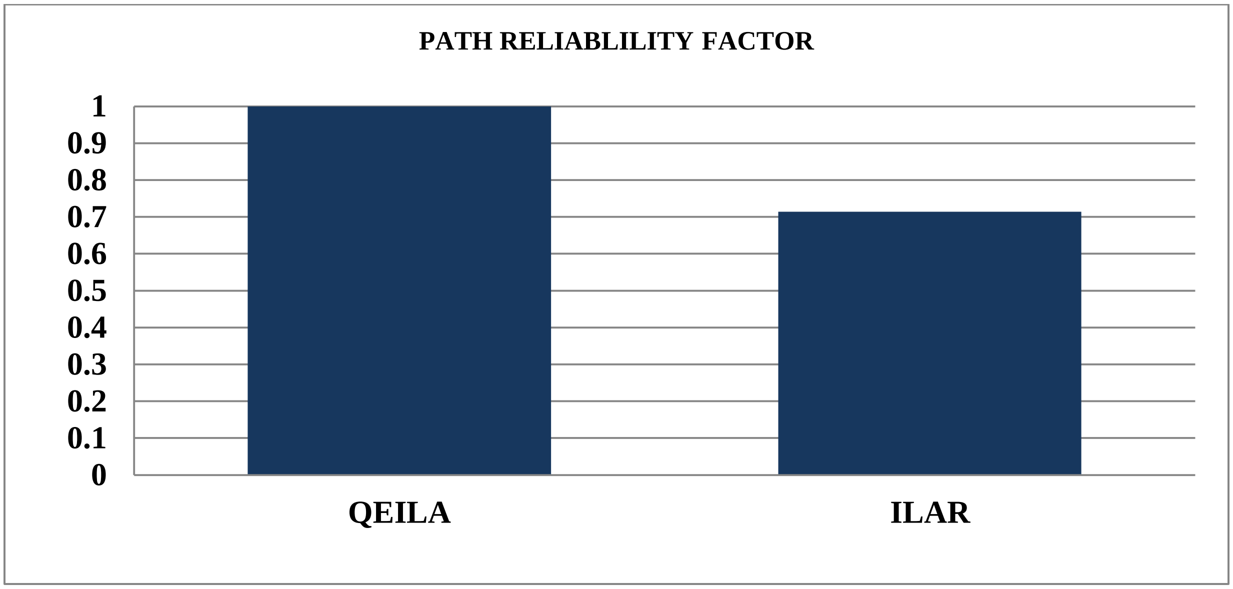


Figure 3.13: Path reliability comparison of QEILA And ILAR

In Fig. 3.13, it can be seen that QEILA constructs more reliable path from source to destination as compared to ILAR for same network topology.

3.6 SUMMARY

In this work QoS Enabled Improved Location Aided routing (QEILA) has been proposed and developed which is an improvement over ILAR. QEILA is also allocation aided routing which provides QoS for media and real-time traffic. QEILA utilizes the candidate next node table CNN which contains the best intermediate node closer to reference line that reduces the forwarding region effectively with consideration of QoS. A partial route repair method has also been proposed which repairs the broken path locally with the help of CNN table and reduces the control overhead considerably.

In next chapter a novel location based routing protocol, Location Information Based Destination Converging Routing Method (LIBDCR) is presented.

CHAPTER 4

LOCATION INFORMATION BASED DESTINATION CONVERGING ROUTING METHOD (LIBDCR)

4.1 INTRODUCTION

In a Mobile ad-hoc network [65], flat routing i.e. *table driven*, and *on demand* [66][67] are generally used for routing the data and control traffic. But it has been observed that the flat routing protocols like DSDV [14] or AODV [21] impose considerable routing overhead because control packets (Table_UPADTE, RREQ, RREP etc.) proliferate in the entire network in uncontrolled manner irrespective of communicating node's (source and destination) position. In this respect, another reactive routing methodology has been evolved in recent years i.e. "*Location Aware Routing*". Location aware routing attempts to control the proliferation of packets pertaining to the position of source and destination node. Many researchers have developed routing protocols in this class, in which source node determines the boundary for packet dissemination with respect to the position of destination and itself in the very beginning of path construction routine. Therefore information packet roams only in the predetermined geographical boundary during the path construction and data transmission process. This saves considerable amount of network resources.

Examples in this category are *Location Aided Routing* (LAR) [46], *Location Aware Routing Protocol with Dynamic Adaptation of Request Zone for Mobile Ad hoc Networks* (LARDAR) [68] and *Location Based Power Aware Routing* is also a novel reactive routing protocol LBPAR [69]. of these protocols, in LAR source node determines a rectangular packet forwarding boundary, while in LARDAR and LBPAR source node determines a triangular boundary. LARDAR is a reactive routing protocol which reduces the forwarding region to triangular by calculating the angle α and β of tangent drawn from source to expected region (as described in chapter 3). LBPAR improve the efficiency of LARDAR. LBPAR also uses triangular forwarding region but instead of angular calculation in LARDAR i.e. α, β LBPAR determines the maximum and minimum slop of tangent drawn from source node to expected region.

4.2 BACKGROUND

In this research, a novel routing scheme Location Information Based Destination Converging Routing Method (LIBDCR) has been proposed which takes LARDAR [68] and LBPAR [69] as reference protocols and improve their efficiency. Unlike LRDAR, LIBDCR calculates pair of tangents T_U and T_L of circular expected region (by equation 4.1), to determine the forwarding region, not only at source node but also at every intermediate node. This leads to convergence of forwarding region to destination node. LIBDCR also ensure the end-to-end QoS by reserving the require band width along the path from sending node to destination node. LIBDCR has also proposed an efficient route maintenance method in which transmitting node repairs broken link locally without involvement of source node, which in turn reduces the control overhead considerably.

$$y = mx \pm r\sqrt{1+m^2} \qquad (4.1)$$

Path construction in LIBDCR is depicted in Fig. 4.1, where at every intermediate node (shown by red circles) forwarding region has been determined.

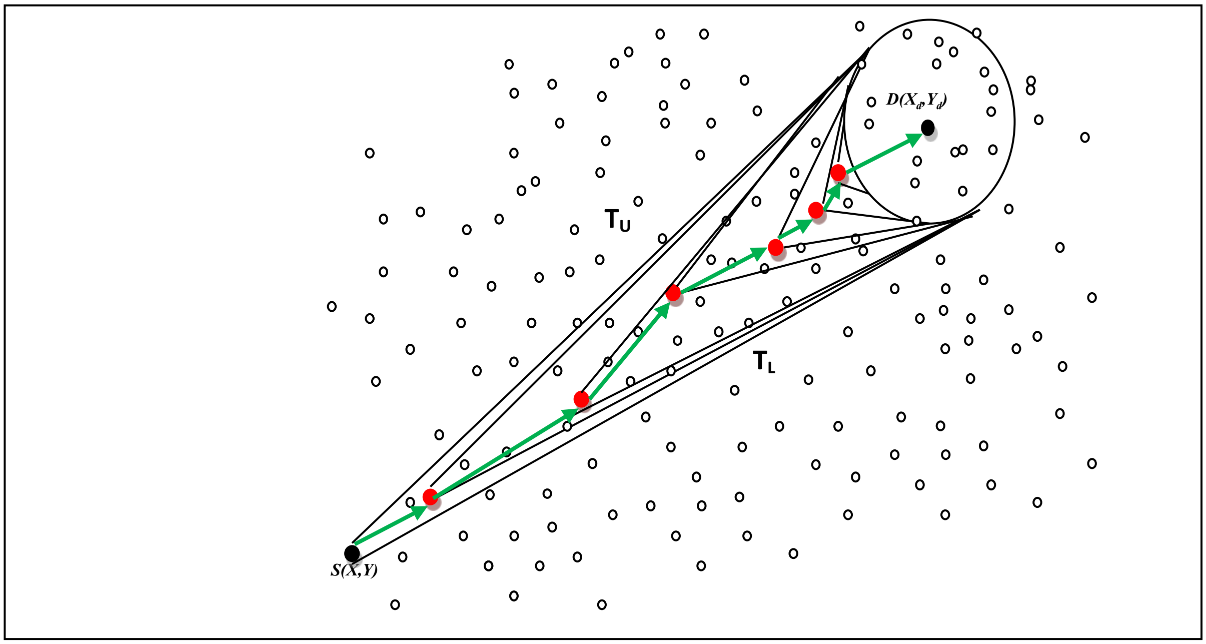


Figure 4.1: Illustration Path construction in LIBDCR

4.3 THE PROCESS

The process of path construction and maintenance in LIBDCR is as follows.

4.3.1 Path construction

Let us assume source node S wants to establish a path with QoS to a destination node D. It calculates the pair of tangents T_U and T_L drawn from itself to expected region. Node S broadcasts a RERQ packet containing T_U , T_L and QoS parameters in its neighbourhood. Packet format for RERQ is shown in Fig. 4.2. After receiving RERQ, neighbouring node checks whether they belong to forwarding region or not by satisfying the equation 4.2. If it is not satisfying the equation then it discards the RERQ. If this is also satisfies and if the node is destination node, it ensure the QoS parameters i.e. ABW>=RB and ABL>=RBL. If yes then it sends a RREP packet on reverse path and reserve the bandwidth for communication by ABW=ABW-RB and each node then receives the RREP packet to write the entry into the current routing table. Source node then starts sending data. Otherwise if destination node dose not fulfil the QoS parameters, it discards the RERQ. If the receiver of RERQ is an intermediate node, it replies with revise route request R_RREQ to the transmitting node. Packet format for R_RREQ is shown in Fig. 4.3. The transmitting node processes such R_RREQ from all nodes in its neighbourhood and makes entry in a Next Forwarding Node table (NFN) (shown in Table 4.1) and sorts the table on ABL field. Then it selects best node M with required QoS parameter. In the next step M reserves the bandwidth for communication by ABW=ABW-RB and attach its own address to record of traversed node and calculates T_U and T_L with its own coordinates (X_N,Y_N). In the meanwhile if M happened to know the new location of destination node D, it updates destination location field in RERQ, calculates T_U and T_L accordingly and broadcasts RERQ packet further with new T_U and T_L . If no such node i.e. M (with required QoS) is found it reports error message ERR.

$$\boxed{(\mathbf{y}-\mathbf{y_1})-\mathbf{m_1}(\boldsymbol{x}-\mathbf{x_1}) \geq \mathbf{0} \;\&\&\; (\mathbf{y}-\mathbf{y_1})-\mathbf{m_2}(\boldsymbol{x}-\mathbf{x_1}) \leq \mathbf{0} \qquad (4.2)}$$

Where $\mathbf{m_1}$ and $\mathbf{m_2}$ are slopes corresponding to tangents $\mathbf{T_U}$ and $\mathbf{T_L}$ respectively, $\boldsymbol{x_1}$ and $\mathbf{y}_1$ are the coordinates of source node $\boldsymbol{x}$ and $\boldsymbol{y}$ are the coordinates of the point which is testing the belongingness to forwarding region

1	TYPE	TTL	Tu	T_L	RB	RBL
2	Source IP Address					
3	Destination IP Address					
4	Broadcast ID					
5	Destination sequence number					
6	Source sequence number					
7	Source location					
8	Destination location					
9	R_O_T					

Figure 4.2: Packet format RERQ in LIBDCR

Where
RB :Required bandwidth
RBL: Required battery life
TTL: Time to live
R_O_T: Record of traversed node

Packet format for reply to route request R_RERQ is shown below.

1	Type
2	Node ID
3	ABW
4	ABL

Figure 4.3: Packet format R_RERQ in LIBDCR

Next forwarding node table (NFN) maintained by transmitting is shown below.

Table 4.1: Next forwarding node table (NFN)

SN	NODE_ID	ABW	ABL
-	-	-	-

Where
ABL- Available Battery Life
ABW- Available Band Width

The algorithm for path construction with necessary assumptions is shown in Fig. 4.4

Let total number of nodes in network is n And A is set of mobile node in network. S,D,N belongs to A. Assume that the source node S wants to find a path to D the destination node , and node N received the routing information

```
PATH CUNSTUCTION(RREQ Packet)
If ( N  in forwarding region && count(address in R_O_T<=TTL)
{
    if (node N is the destination node D)
       {
          If(node  N ‘s  AB>=RB && RBL>=ABL)
                Node N Reserves  bandwidth for communication by ABW=ABW-RB
                Node N sends a RREP packet on reverse path..
                Each node receives the RREP packet to write the entry into the current routing table.
                Node S starts to send data.
          else
             Discard RERQ.
       }
    else
       {
             Node N forwards a R_RREQ packet to the transmitting node.
             The transmitting node process the all received R_RREQ and perform following step .
                 Made entry in NFN table with each R_RREQ.
                 Sort the NFN table in descending  order of ABL.
                 M=SELCTBEST (NFN)
                 If(M!=NULL)
                     {
                     Node M Reserves  bandwidth for communication
                     AB=ABW-RB
                     Attach own address record of traversed node;
                     Calculate $T_U$, $T_L$ with M’s Coordinates.
                     Update $T_U$, $T_L$in RERQ packet with recently calculated $T_U$, $T_L$.
                     Broadcast RREQ packet;
                     }
                 else
                     Report No suitable node found ;
 }
 else
       Discard  RREQ packet;
}
ALGO SELCTBEST (NFN)
{
       For(i=0 to k)  //k  is size of NFN table
         {
          If (NFN[i].AB>=RB&&NFN[i].ABL>=RBL)
          return NFN[i];
         }
     return  NULL
}
```

Figure 4.4: Algorithm for path construction in LIBDCR

4.3.2 Route Maintenance

An efficient method for partial route repair has been proposed in which intermediate node that observes broken link consult its NFN table which is created at the time of path construction. As mention earlier, that at an intermediate node NFN table maintains the list of next forwarding node with QoS parameters. Whenever an intermediate node finds broken link it deletes the entry of abandoned next node from its NFN table and searches again for suitable alternative node with QoS parameter in its NFN table (likelihood of presence of alternative node is very high in NFN), if such node found in the table then it can unicast the RERQ to that alternative node, otherwise it rebroadcast RERQ in it neighbourhood to repair the path. As the information of broken link does not have to propagate back to the sender and repaired locally it reduces control overhead considerably. Algorithm for path repair with necessary assumptions is given below in Fig. 4.5.

```
   ALGO PATH  REPAIR
   Let total number of nodes in network is n and A is set of mobile node in network.
   S,D,I belongs to A
   Assume that the source node S communicating with destination node D and   intermediate
node P found a broken link .
   if (Pfound broken link )
   {
        1.P deletes entry of abandoned node from NFN table ;
        2.M=findbest(NFN)
            If(M!=NULL)
              {
                Unicast RERQ to M;
                }
        else
         rebroadcast RERQ in vicinity;
}
```

Figure 4.5: Algorithm for path maintenance with necessary assumptions

Illustration of route maintenance is shown in Fig. 4.6 in which node P realizes the broken link from P to B, it selects an alternate node C to connect with Q and repair the path locally as shown by dashed line

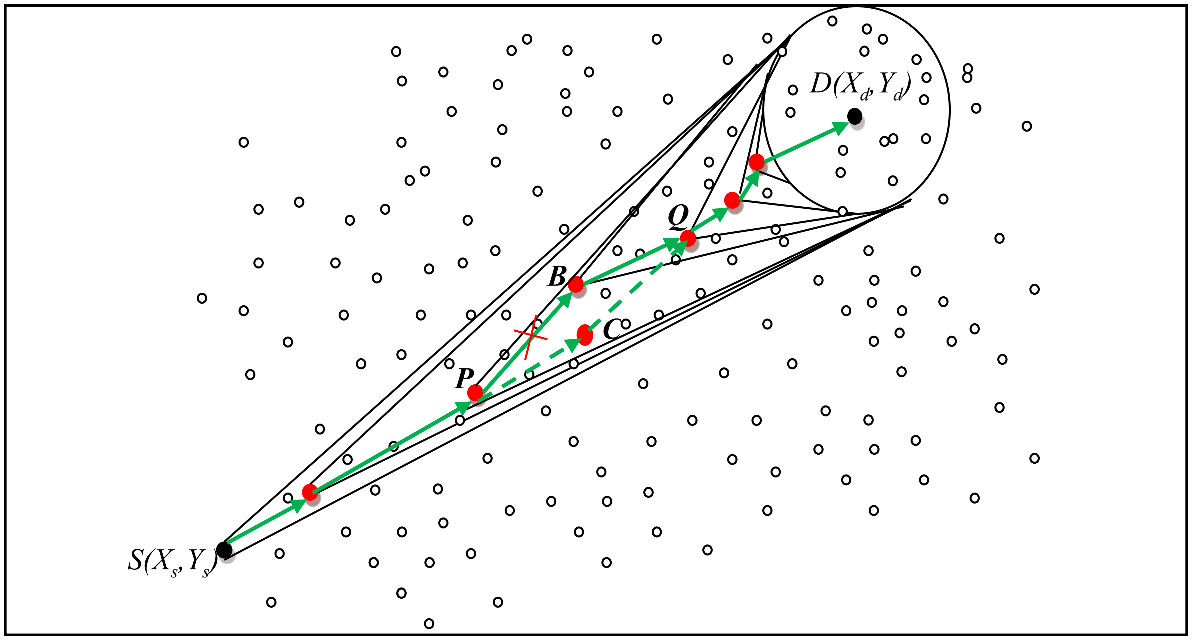


Figure 4.6: Illustration of route maintenance in LIBDCR

4.4 RESULTS AND DISCUSSION

Location Aware Routing protocol relies on geometrical relationships amongst mobile node's physical location, provided that geographical coordinates of every node is known to every other node. A mathematical tool was needed for simulation, which provides Mathematical support for its visualization. The proposed protocol Location Information Based Destination Converging Routing Method (LIBDCR) and corresponding protocol Location Based Power Aware Routing (LBPAR) has been simulated in the same network topology using MATLAB 7.8.0. For simulation purpose, 57 randomly generated nodes have been plotted in 300 X 300 meter^2 area. Available battery life of nodes has been taken between 0-100 % and available bandwidth has been taken between 0-10 mbps. The findings during the path construction of both protocols have been recorded and are depicted in Fig. 4.7(a-n) and 4.8 (a-l). The findings during the simulation have been tabulated in table 4.2 and 4.3. The results of both the protocols are compared in Table 4.4. Results show that in case of LIBDCR the route request region is smaller as compared to LBPAR. LIBDCR decreases the route request region at each intermediate node. This results in less number of nodes participating in route construction process which in turn leads to reduction in the route construction cost, and saves network energy considerably. LIBDCR also controls the quality of path generated.

Simulation stages of path construction in case of LBPAR are shown in Fig 4.7(a-n). Where source node is 1 and destination node is 32.

Stage 1: Arbitrary topology of an adhoc network where source node 1 seeks the path to destination 32 as shown in Fig. 4.7(a).

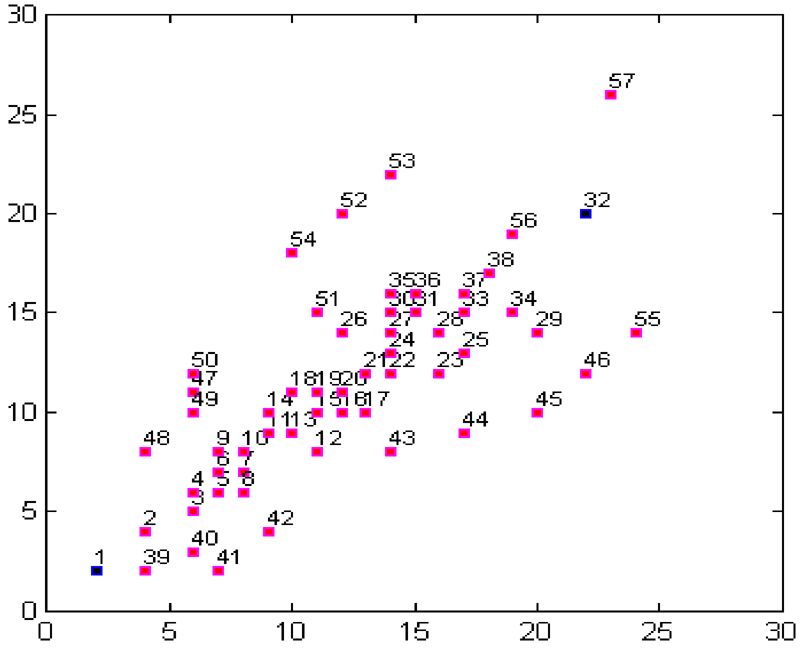


Figure 4.7 (a): Stage 1 of LBPAR

Stage 2: Source node 1 determines the expected region as shown in Fig. 4.7(b).

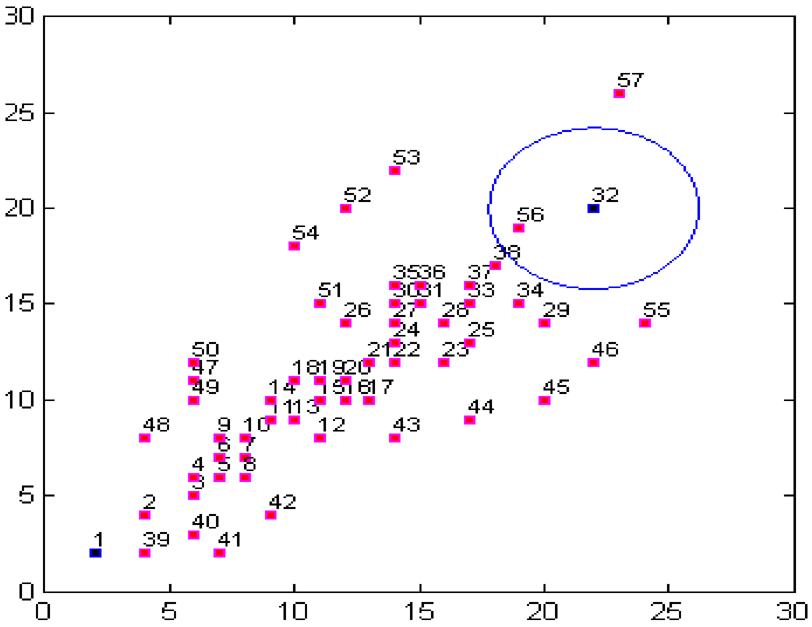


Figure 4.7 (b): Stage 2 of LBPAR

Stage 3: Source node 1 determines the forwarding region, as shown in Fig. 4.7(c).

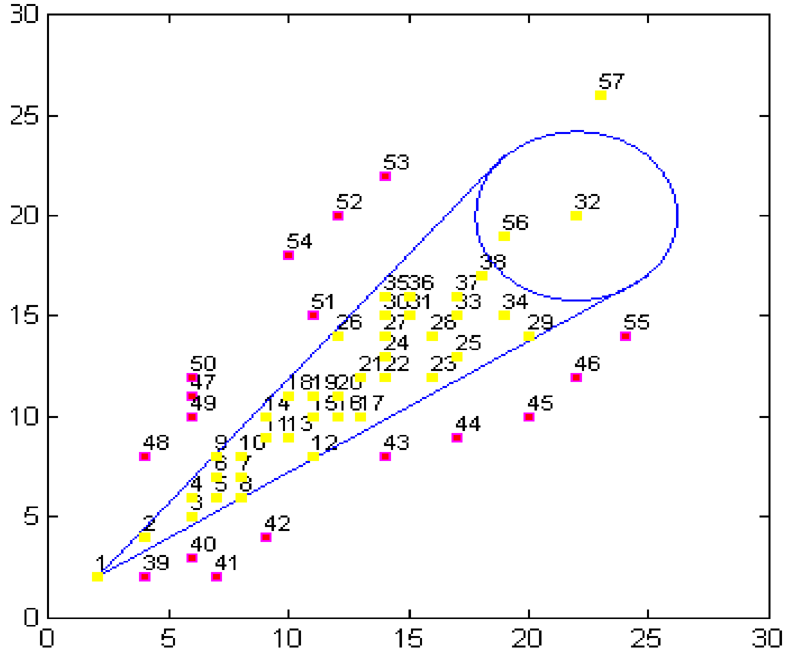


Figure 4.7 (c): Stage 3 of LBPAR

Stage 4: Node 1 forwards the RERQ in its vicinity, as shown in Fig. 4.7(d).

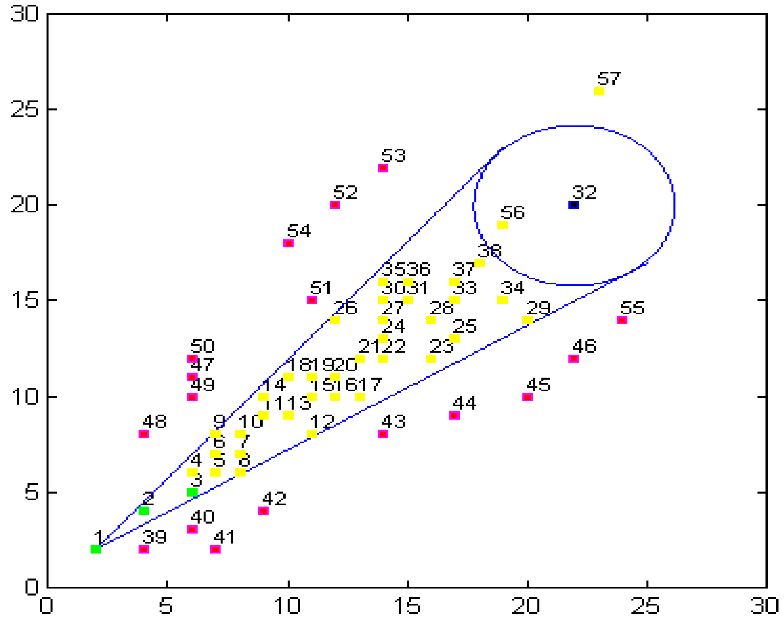


Figure 4.7 (d): Stage 4 of LBPAR

Stage 5: Node 1 selects next forwarding node 2 as shown in Fig. 4.7(e).

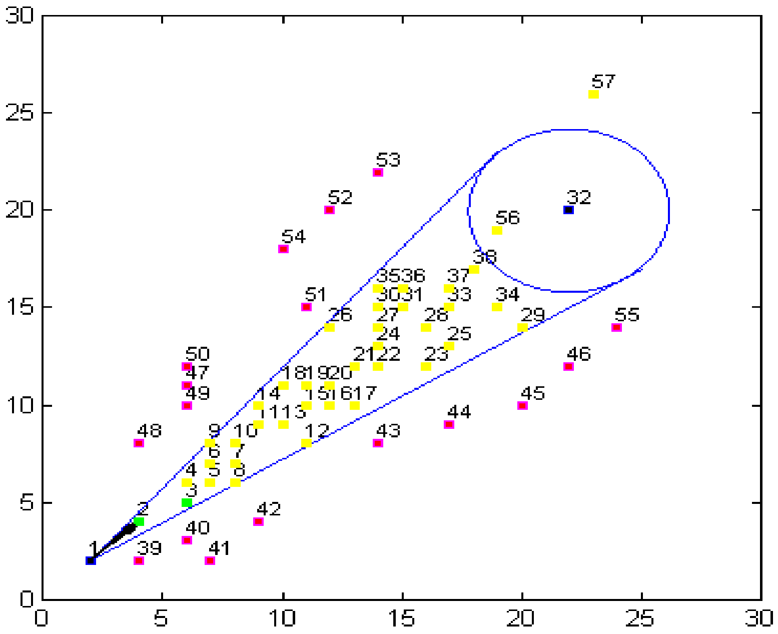


Figure 4.7 (e): Stage 5 of LBPAR

Stage 6: Node 2 forwards the RERQ in its vicinity and selects node 3 as next forwarding node. as shown in Fig. 4.7(f).

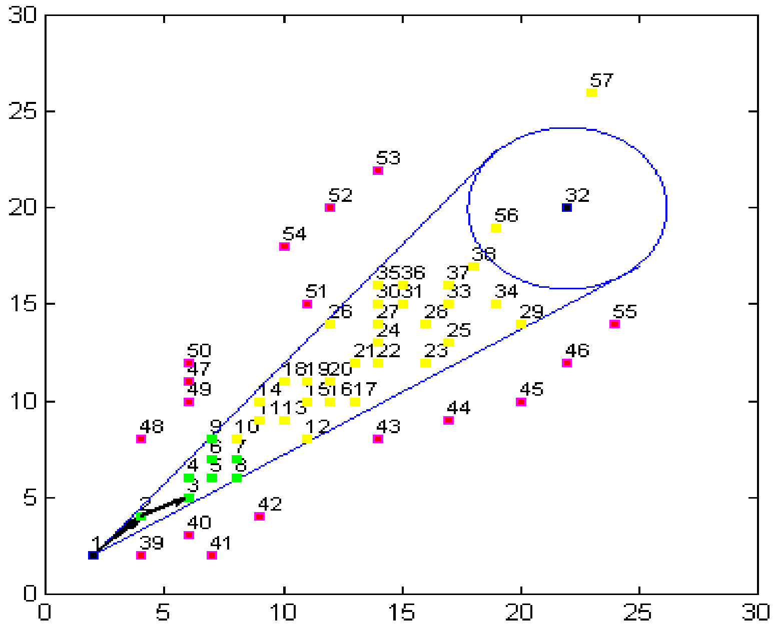


Figure 4.7 (f): Stage 6 of LBPAR

Stage 7: Node 3 forwards the RERQ in its vicinity and selects node 8 as next forwarding node as shown in Fig. 4.7(g).

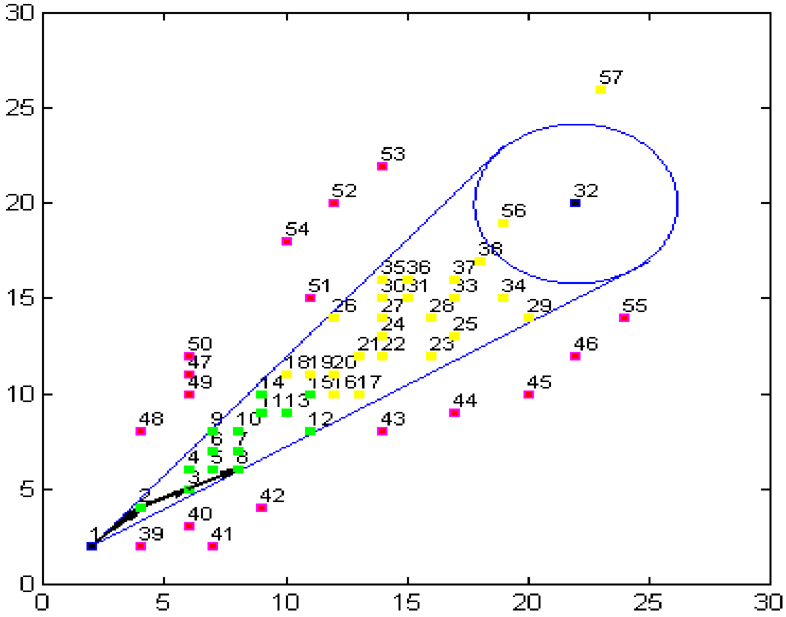


Figure 4.7 (g): Stage 7 of LBPAR

Stage 8: Node 8 forwards the RERQ in its vicinity and selects node 14 as next forwarding node as shown in Fig. 4.7(h).

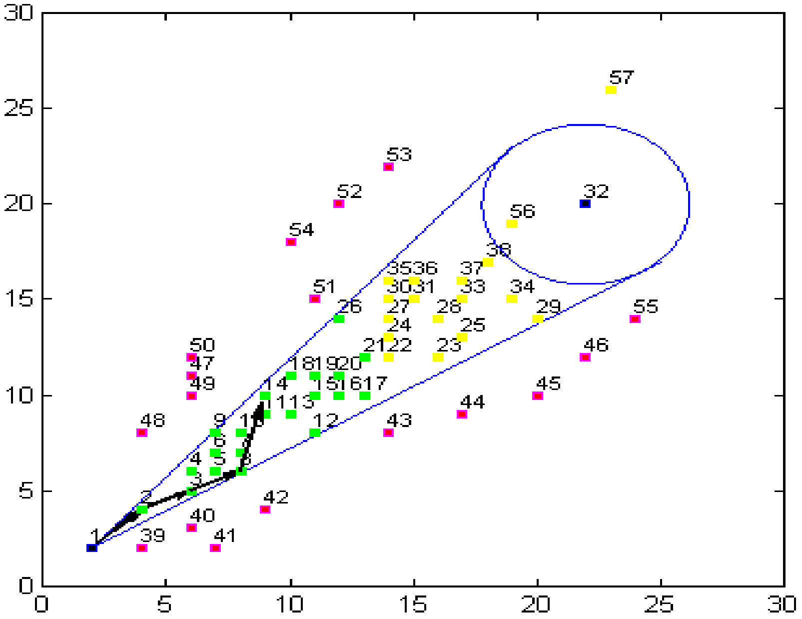


Figure 4.7 (h): Stage 8 of LBPAR

Stage 9: Node 14 forwards the RERQ in its vicinity and selects node 17 as next forwarding node as shown in Fig. 4.7(i).

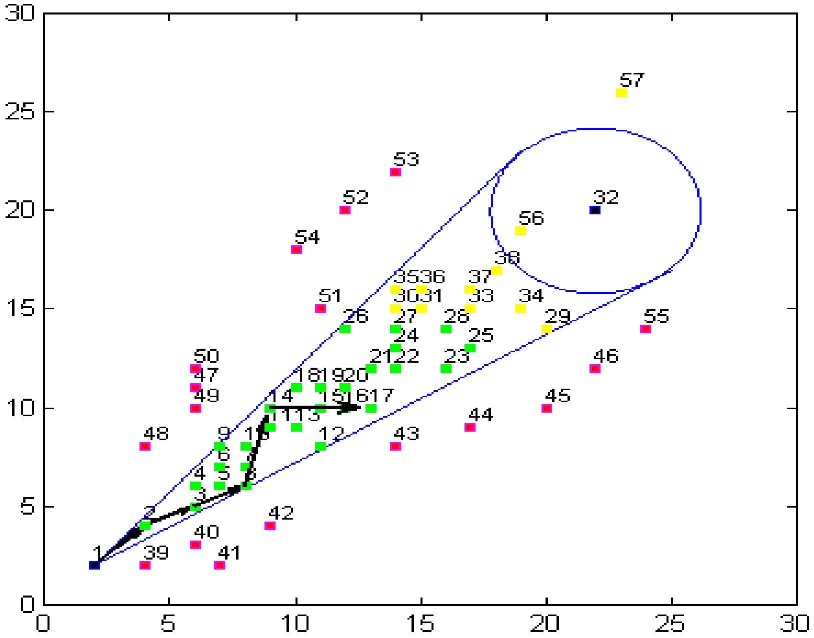


Figure 4.7 (i): Stage 9 of LBPAR

Stage 10: Node 17 forwards the RERQ in its vicinity and selects node 23 as next forwarding node as shown in Fig. 4.7(j).

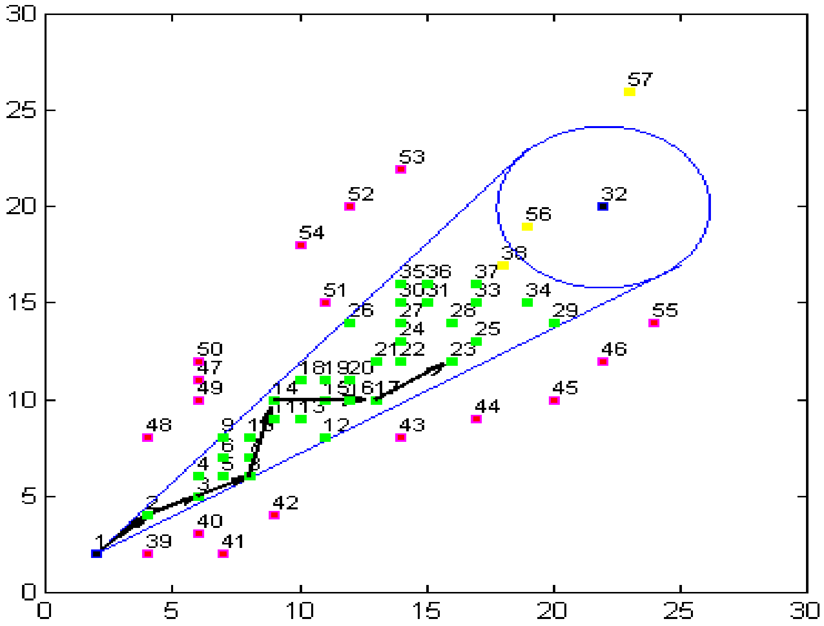


Figure 4.7 (j): Stage 10 of LBPAR

Stage 11: Node 23 forwards the RERQ in its vicinity and selects node 31 as next forwarding node, as shown in Fig.4.7(k).

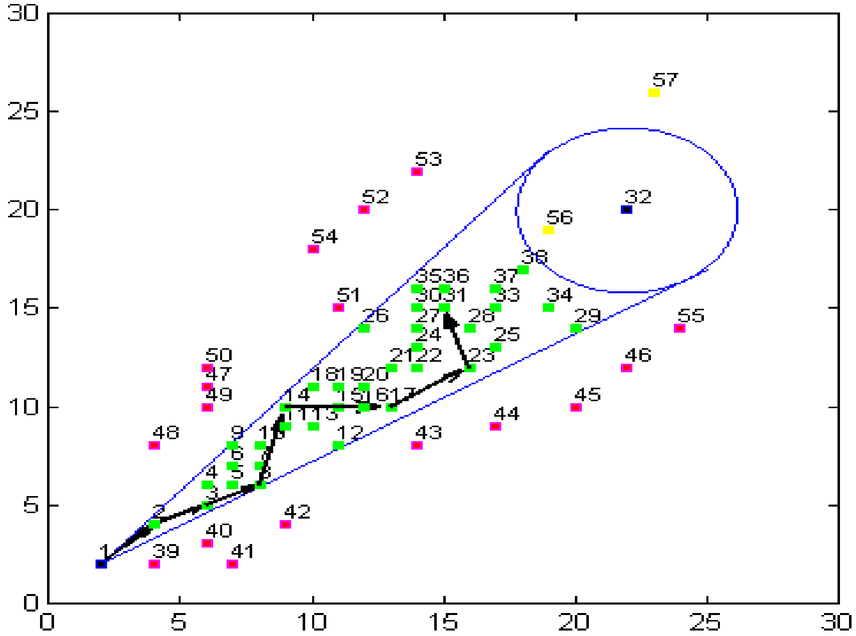


Figure 4.7 (k): Stage 11 of LBPAR

Stage 12: Node 31 forwards the RERQ in its vicinity and selects node 38 as next forwarding node as shown in Fig. 4.7(l).

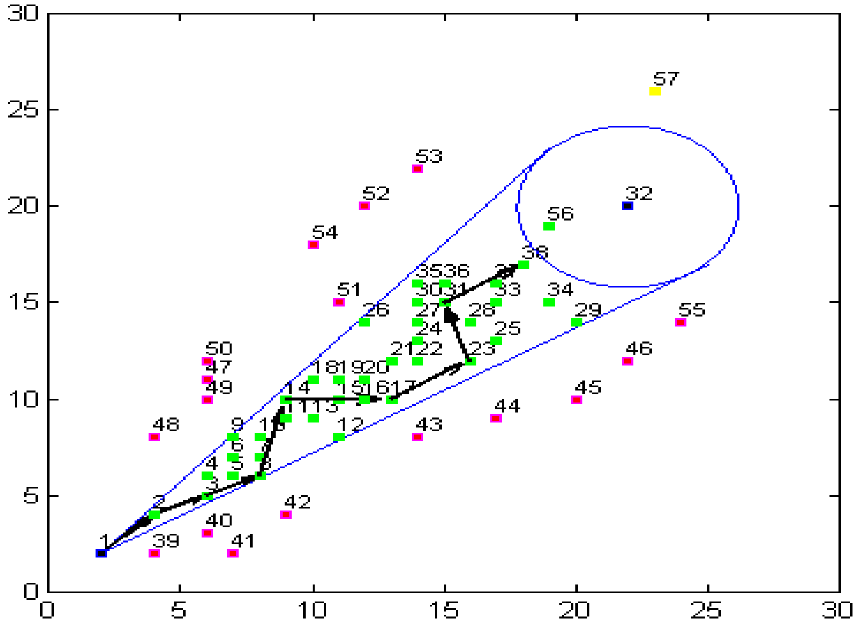


Figure 4.7 (l): Stage 12 of LBPAR

Stage 13: Node 38 forwards the RERQ in its vicinity and found the destination node 32, as shown in Fig. 4.7(m).

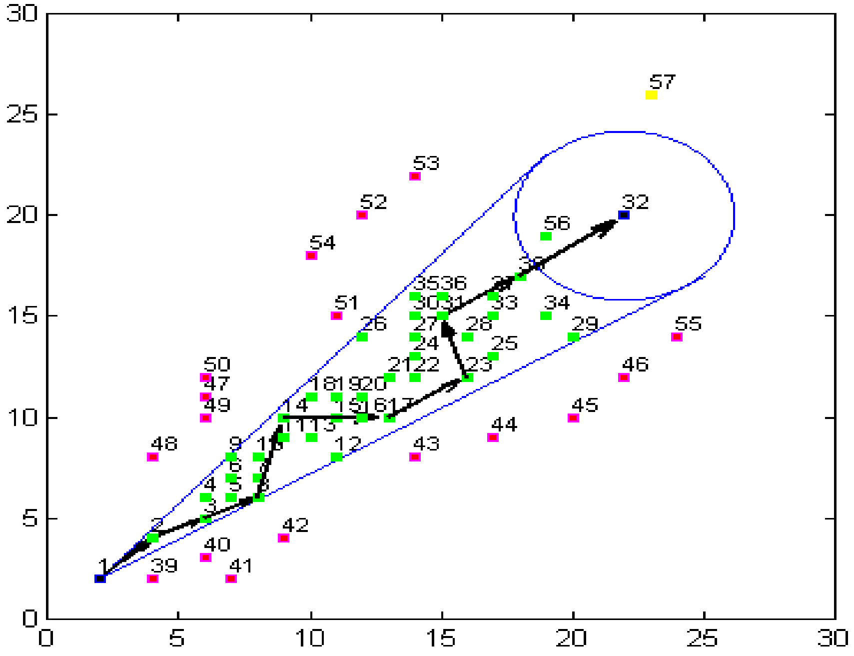


Figure 4.7 (m): Stage 13 of LBPAR

Stage 14: destination node 32 respond with RREP that travels along the reverse path back to the source node 1 and path get established as shown in Fig. 4.7(n).

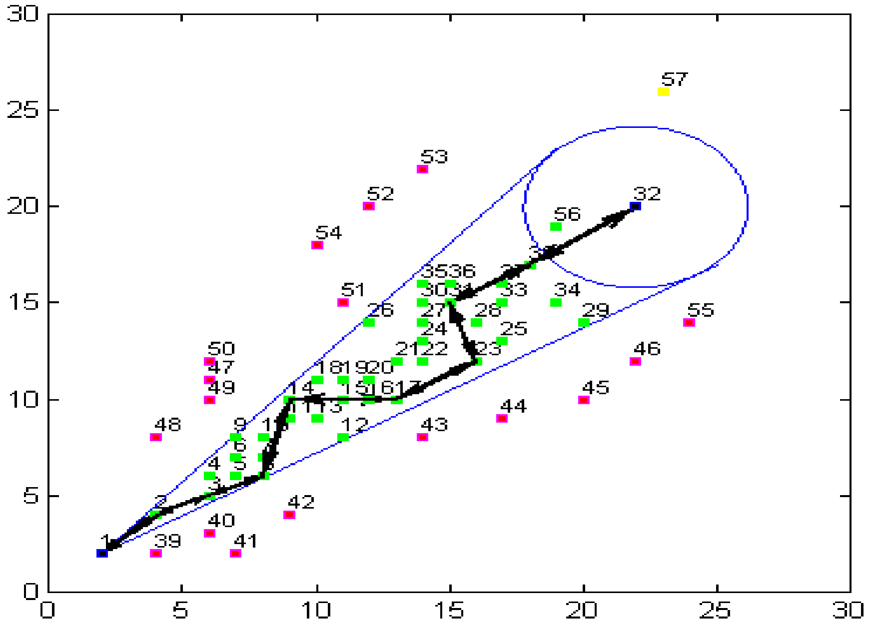


Figure 4.7 (n): Stage 14 of LBPAR

Simulation findings of path construction stages in LBPAR are shown in table 4.2.

Table 4.2: Finding of path construction in LBPAR

Node #	Findings in path construction in LBPAR		Total
1(S)	**Nodes in FR**	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37 38,56,57	**40**
	Nods in Vicinity & FR	2,3	**2**
	ABL(%)	55, 51	
	ABW(Mbps)	5.4, 5.1	
2	**Nodes in FR**	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37 38,56,57	**40**
	Nods in Vicinity & FR	1,3,4,5,6,7,8,9	**8**
	ABL(%)	51,51,52,60,45,78,46,55	
	ABW(Mbps)	5.0,5.1,5.0,5.9,4.0,5.9,4.8,5.4	
8	**Nodes in FR**	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,56,57	**40**
	Nods in Vicinity & FR	2,3,4,5,6,7,9,10,11,12,13,14,15	**13**
	ABL(%)	55,51 ,52,60,45,78,55,78 ,56,52 ,46,51,62	
	ABW(Mbps)	5.4,5.12,5.0, 5.9,4.0,5.9,5.4,6.9,5.4,5.0,5.9,5.0,7.0	
14	**Nodes in FR**	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37 38,56,57	**40**
	Nods in Vicinity & FR	4,5,6 ,7 ,8 ,9,10 ,11,12 ,13 ,15,16,17,18 ,19 ,20,21,26	**18**
	ABL(%)	52,60,45,78,46,55,78,56,52,46,62,51,40,51,51,58,51,45	
	ABW(Mbps)	5.0,5.9,4.0,5.9,4.8,5.4,6.9,5.4,5.0,5.9,7.0,6.1,3.8,6.7,7.8,6.1,5.0,4.0	
17	**Nodes in FR**	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37 38,56,57	**40**
	Nods in Vicinity & FR	11,12,13,14,15,16,18,19,20,21,22,23,24,25,26,27,28	**17**
	ABL(%)	56,52,46,51,62,51,51,51,58,51,75,54,52,60,45,70,76	
	ABW(Mbps)	5.4,5.0,5.9,5.0,7.0,6.1,6.7,7.8,6.1,5.0,5.4,5.1,5.0,5.9,4.0,5.0,7.8	
23	**Nodes in FR**	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37 38,56,57	**40**
	Nods in Vicinity & FR	16,17,20,21,22,24,25,26,27,28,29,30,31,33,34,35,36,37	**18**
	ABL(%)	51,40,58,51,75,52,60,45,70,76,55,78,56,76,41,42 51,90	
	ABW(Mbps)	6.1,3.8,6.1,5.0,5.4,5.0,5.9,4.0,5.0,7.8,5.4,6.9,4.0,5.9,7.4,3.9,6.1,7.8	
31	**Nodes in FR**	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37 38,56,57	**40**
	Nods in Vicinity & FR	20,21,22,23,24,25,26,27,28,30,33,34,35,36,37,38	**16**
	ABL(%)	58,51,75,54,52,60,45,70,76,78,76,41,42,51,90,53	
	ABW(Mbps)	6.1, 5.0,5.4,5.1,5.0,5.7,4.0,5.0,7.8,6.6,5.5,7.4,3.6 6.1,7.4,6.7	
38	**Nodes in FR**	1,2, 3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37 38,56,57	**40**
	Nods in Vicinity & FR	25,27,28,29,30,31,32,33,34,35,36,37,56	**13**
	ABL(%)	60,70,76,55,78,56,67,76,41,42,51,90,48	
	ABW(Mbps)	5.9,5.0,7.8,5.4,6.9,4.0,6.6,5.9,7.4,3.9,6.1,7.8,6.1000	
32(D)	---	---	**---**

Simulation stages of path construction in case of LIBDCR are shown in fig 4.8 (a-l). Where source node is 1 and destination node is 32.

Stage 1: Arbitrary topology of network where source node 1 seeks the path to destination node 32, as shown in Fig. 4.8(a).

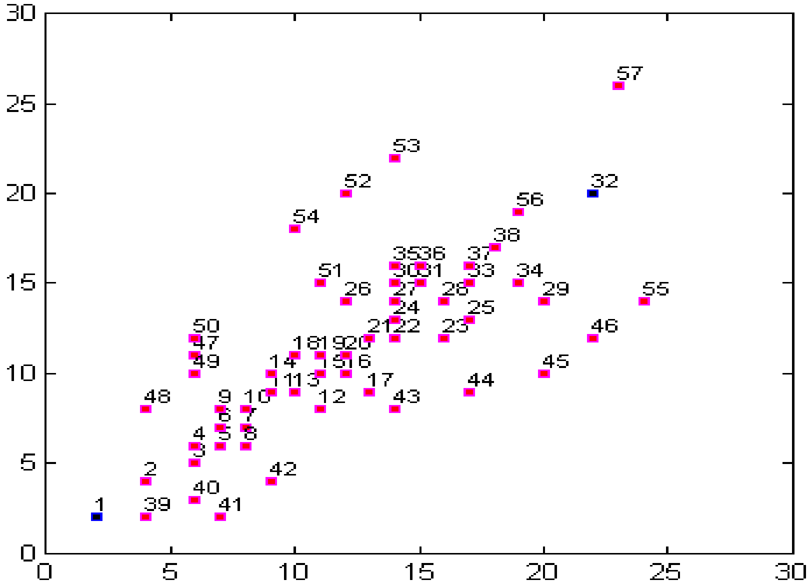


Figure 4.8(a): Stage 1 of LIBDCR

Stage 2: Source node 1 determines the expected region, as shown in Fig.4.8(b).

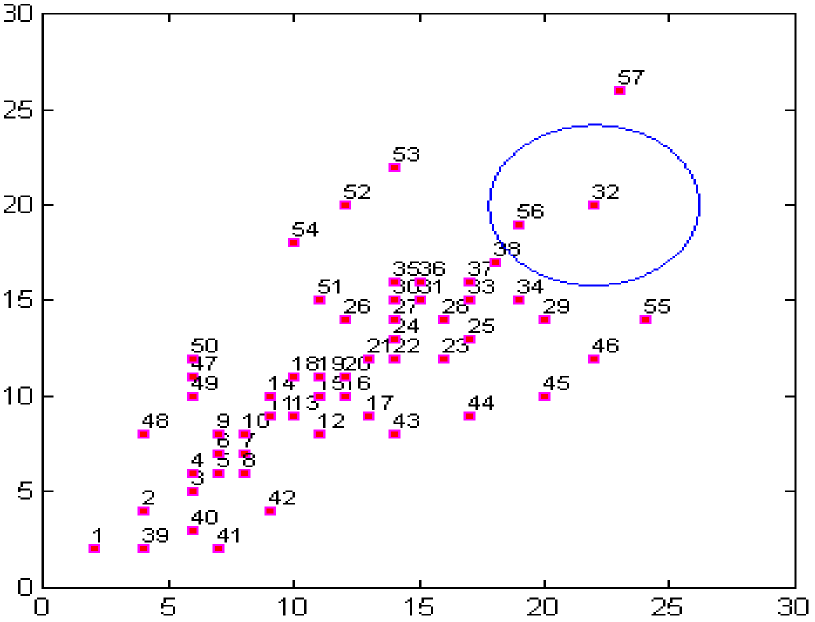


Figure 4.8 (b): Stage 2 of LIBDCR

Stage 3: Node 1 determines the forwarding region, as shown in Fig. 4.8(c).

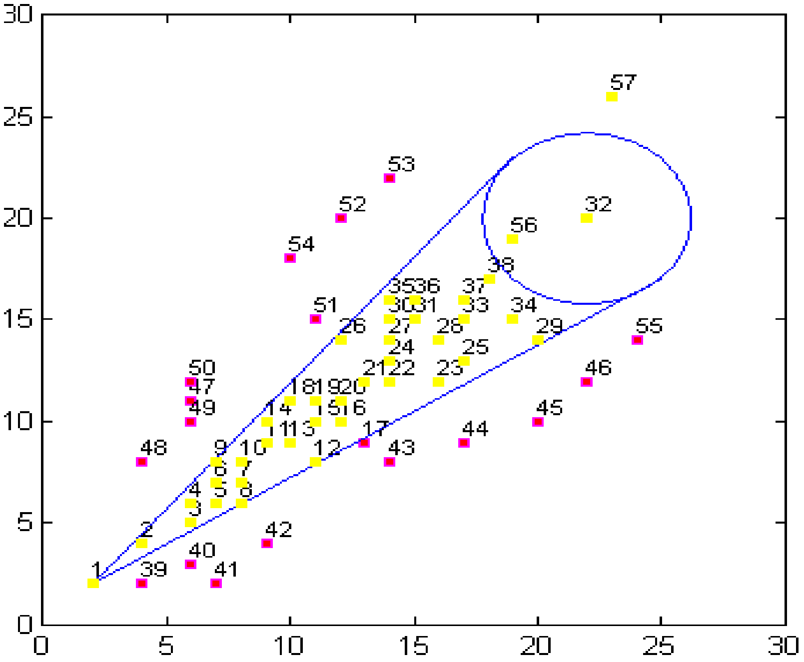


Figure 4.8 (c): Stage 3 of LIBDCR

Stage 4: Node 1 broadcast the RERQ in its vicinity and selects the next forwarding node 2, the node 2 again determines the forwarding region, and carries the process further, as shown in Fig. 4.8(d).

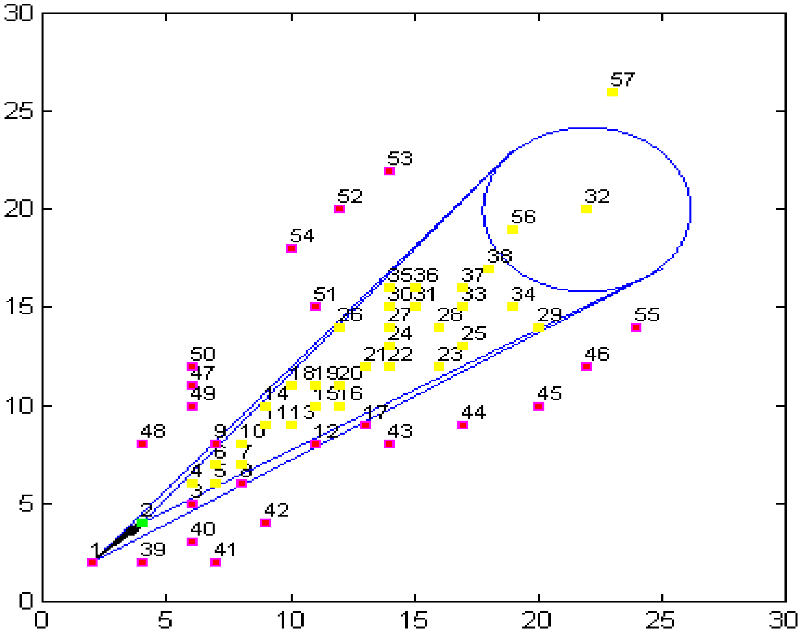


Figure 4.8 (d): Stage 4 of LIBDCR

Stage 5: Node 2 broadcasts the RERQ in its vicinity and selects the next forwarding node 7, and the node 7 again determines the forwarding region, and carries the process further, as shown in Fig. 4.8(e).

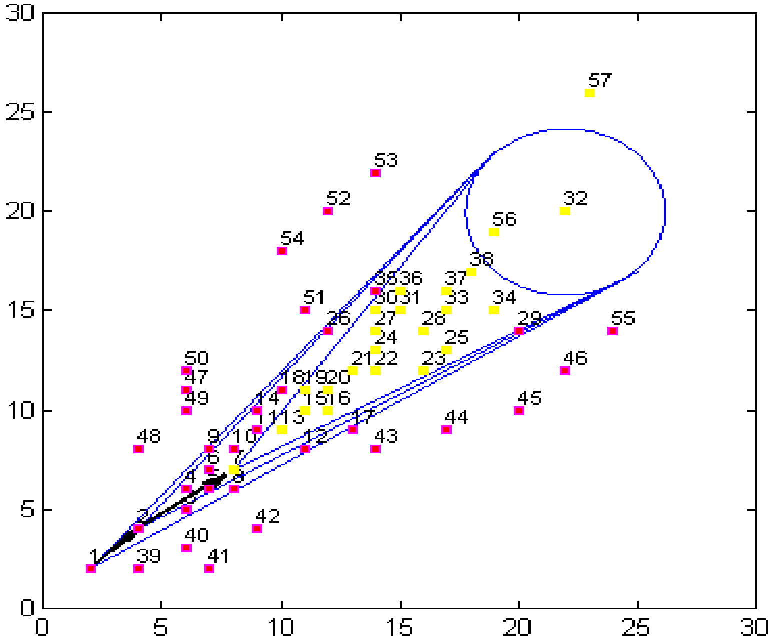


Figure 4.8 (e): Stage 5 of LIBDCR

Stage 6: Node 7 broadcasts the RERQ in its vicinity and selects the next forwarding node 15. The node 15again determines the forwarding region, and carries the process further, as shown in Fig. 4.8(f).

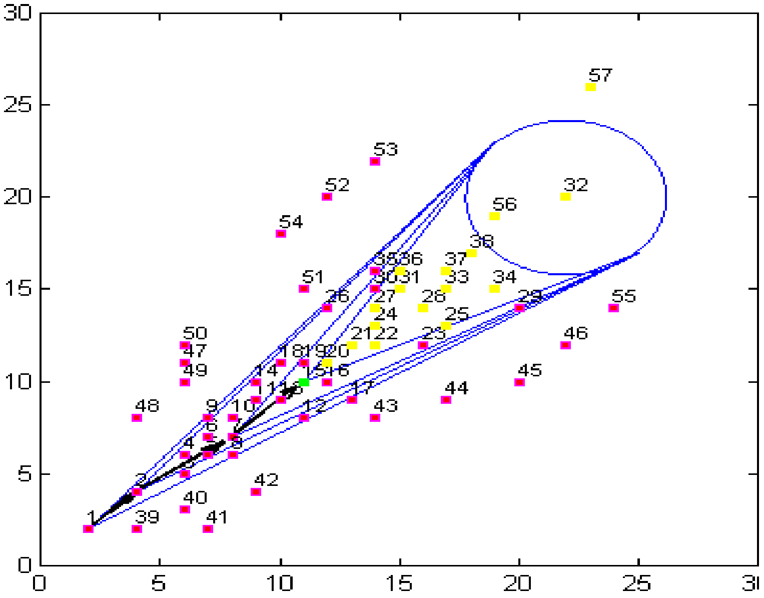


Figure 4.8 (f): Stage 6 of LIBDCR

Stage 7: Node 15 broadcasts the RERQ in its vicinity and selects the next forwarding node 22. The node 22 again determines the forwarding region, and carries the process further, as shown in Fig. 4.8(g).

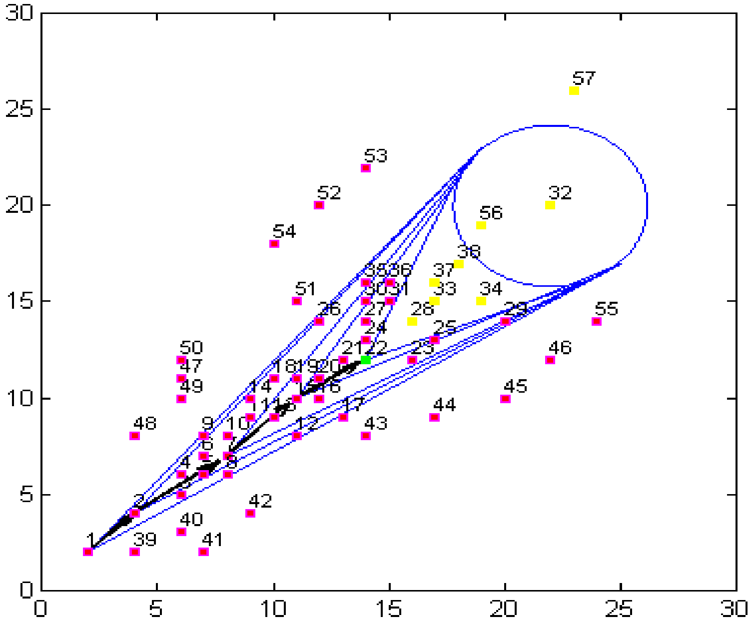


Figure 4.8 (g): Stage 7 of LIBDCR

Stage 8: Node 22 broadcasts the RERQ in its vicinity and selects the next forwarding node 37. The node 37 again determines the forwarding region, and carries the process further, as shown in Fig.4.8(h).

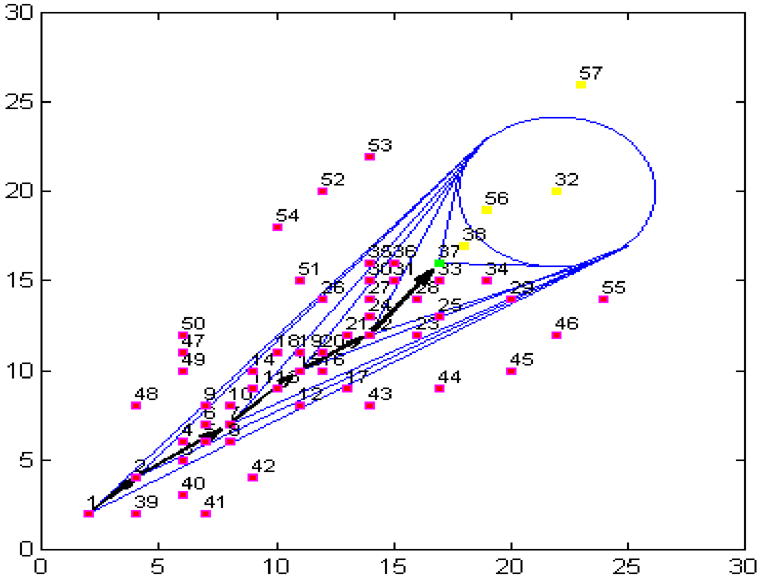


Figure 4.8 (h): Stage 8 of LIBDCR

Stage 9: Node 37 broadcasts the RERQ in its vicinity and selects the next forwarding node 38. The node 38 again determines the forwarding region, and carries the process further, as shown in Fig. 4.8(i).

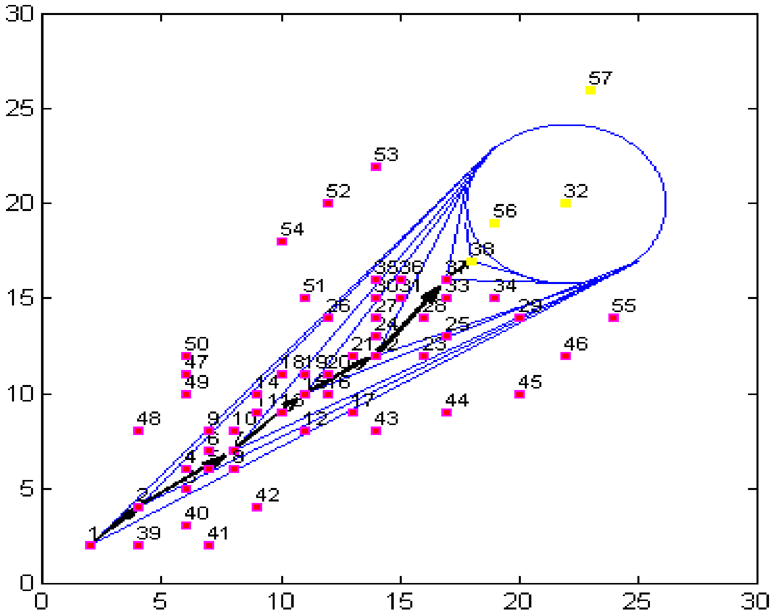


Figure 4.8 (i): Stage 9 of LIBDCR

Stage 10: Node 38 broadcast the RERQ in its vicinity, as shown in Fig. 4.8(j).

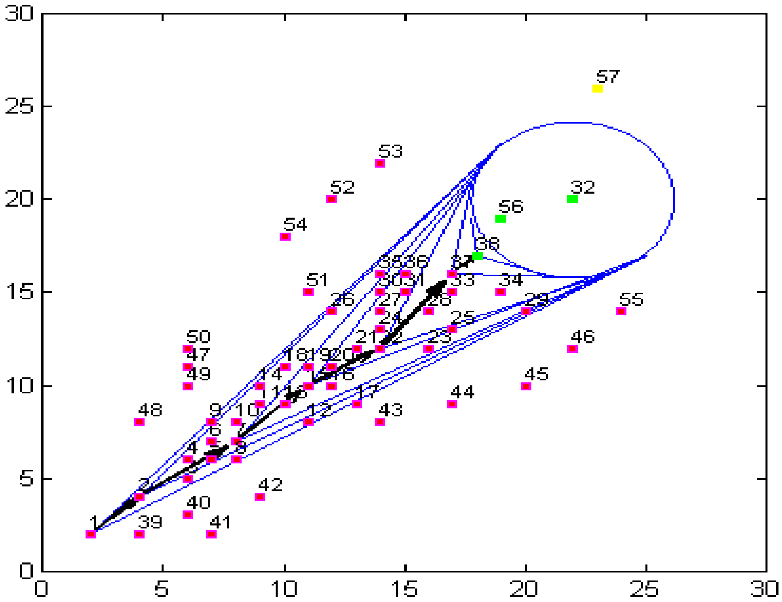


Figure 4.8 (j) : Stage 10 of LIBDCR

Stage 11: Node 38 determines the presence of destination node 32 and establishes a link to 32, as shown in Fig. 4.8(k).

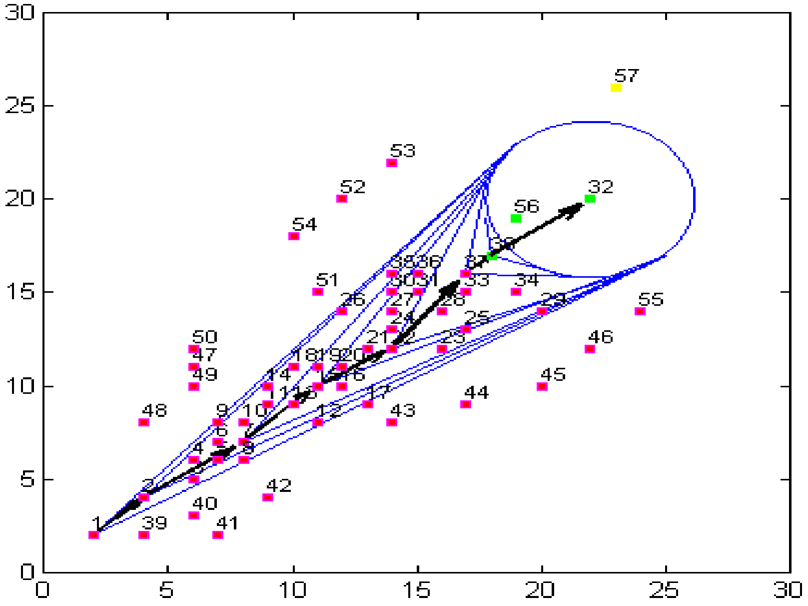


Figure 4.8 (k): Stage 11 of LIBDCR

Stage 12: Destination node 32 respond with RREP that travels along the reverse path back to the source node 1 and path get established,   as shown in Fig. 4.8(l).

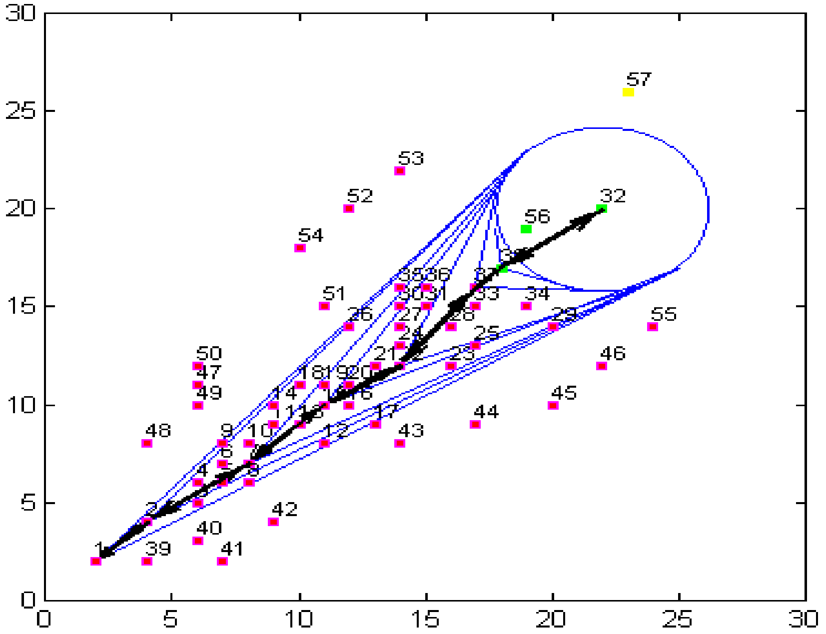


Figure 4.8 (l) : Stage 12 of LIBDCR

Simulation findings of path construction stages in LIBDCR are shown in Table 4.3

Table 4.3: Simulation findings of path construction in LIBDCR

Node No.	Findings in path construction in LIBDCR						Total
1(S)	**Nodes in FR**	1,2, 3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37 38,56,57					**40**
	Nods in Vicinity & FR	2	3				**2**
	ABL(%)	55	51				
	ABW(Mbps)	5.4000	5.1200				
2	**Nodes in FR**	2 ,4,5,6,7,10,11,13,14,15,16,17,18,19,20,21,22,23,24 25,26,27,28,29,30,31,32,33,34,35,36,37,38,56,57					**35**
	Nods in Vicinity & FR	4	5	6	7		**4**
	ABL(%)	52	60	45	78		
	ABW(Mbps)	5.1	5.9000	4.0000	5.9610		
7	**Nodes in FR**	7,13 ,15,16,17,19,20,21,22,23,24,25,27,28,30,31 32,33,34, 36,37,38,56,57					**24**
	Nods in Vicinity & FR	13	15	16	19		**4**
	ABL(%)	46	62	51	51		
	ABW(Mbps)	5.1	7.0620	6.1000	7.8910		
15	**Nodes in FR**	15,20,21,22,24,25,27,28,31,32,33,34,36,37,38,56,57					**17**
	Nods in Vicinity & FR	20	21	22	24	27	**5**
	ABL(%)	58	51	75	52	70	
	ABW(Mbps)	6.0	5.0500	5.4000	5.0920	5.0610	
22	**Nodes in FR**	22,28,32,33,34,37,38,56,57					**9**
	Nods in Vicinity & FR	28	33	37			**3**
	ABL(%)	76	76	90			
	ABW(Mbps)	7.8300	5.9860	7.8900			
37	**Nodes in FR**	32 ,37,38,56,57					**5**
	Nods in Vicinity & FR	38	56				**2**
	ABL(%)	53	51				
	ABW(Mbps)	6.7100	6.1000				
38	**Nodes in FR**	32,38,56,57					**4**
	Nods in Vicinity & FR	32	56				**2**
	ABL(%)	67	51				
	ABW(Mbps)	6.6982	6.1000				
32(D	**---**	---	---				**---**

It may be observed from the Table 4.4 that path constructed by LIBDCR is more stable and reliable in terms of path life and bandwidth i.e. each node in the path has good battery life and bandwidth.

Table 4.4: Comparison of findings of path formation in LIBDCR and LBPAR

SN	LBPAR (Existing)					Path construction in LIBDCR(Proposed)				
	Node no	No of Nodes in FR	Nods in Vicinity & FR	ABL (%)	ABW (Mbps)	Node no	No of Nodes in FR	Nods in Vicinity & FR	ABL (%)	ABW (Mbps)
1	1(S)	40	2	51	5.05	1(S)	40	2	51	5.05
2	2	40	8	55	5.400	2	35	4	55	5.400
3	8	40	13	46	4.8	7	24	4	78	5.961
4	14	40	18	51	5.0	15	17	5	62	7.0620
5	17	40	17	40	3.8	22	9	3	75	5.400
6	23	40	18	54	5.1	37	5	2	90	7.8900
7	31	40	16	56	4.0	38	4	2	53	6.7100
8	38	40	13	53	6.7100	32(D)	-	-	67	6.6982
9	32(D)	-	-	67	6.6982	-	-	-	-	-

In mobile ad-hoc network, number of participating nodes in route discovery process has cost implications in terms of network energy, bandwidth and time. So the major goal of any location aware routing protocol is to keep less number of participating nodes while constructing the path, so that the route request proliferation should be as minimum as possible. Simulation results in Table 4.4 show that in case of Location Based Power Aware Routing the path from source to destination contains 1 →2 →8 →14 →17 →23 →31 →38 →32 nodes, and at every node the number of participating nodes in route discovery process remain the same i.e. 40, this implies that whether the RERQ transmitting node is closer or farther to destination ,all nodes in forwarding region has to process the RERQ. Whereas in case of Location Information Based Destination Converging Routing Method (LIBDCR) the path from source node 1 to destination node 32 is 1 →2 →7 →15 →22 →37 →38 →32. In this path the forwarding region gets narrowed down at every successive intermediate node, Table 4.4 also shows that, at source node 1 the intermediate nodes are 2,7,15,22,37,38 and the number of participating nodes in route discovery are 40,35,24,9,5,4 respectively. It can be seen that at every step, number of participating nodes are getting lesser in number. This helps in saving the network energy, bandwidth and time. Furthermore it also ensures that the path discovery process moves in forward direction only i.e. nodes that are behind the transmitting node, will not participate in path discovery process again. LIBDCR also ensures the quality of transmission path, by selecting the reliable nodes which have certain level of battery life and bandwidth for path construction. In Fig. 4.9 Nodes involvement in path construction process for both protocols is shown as graph.

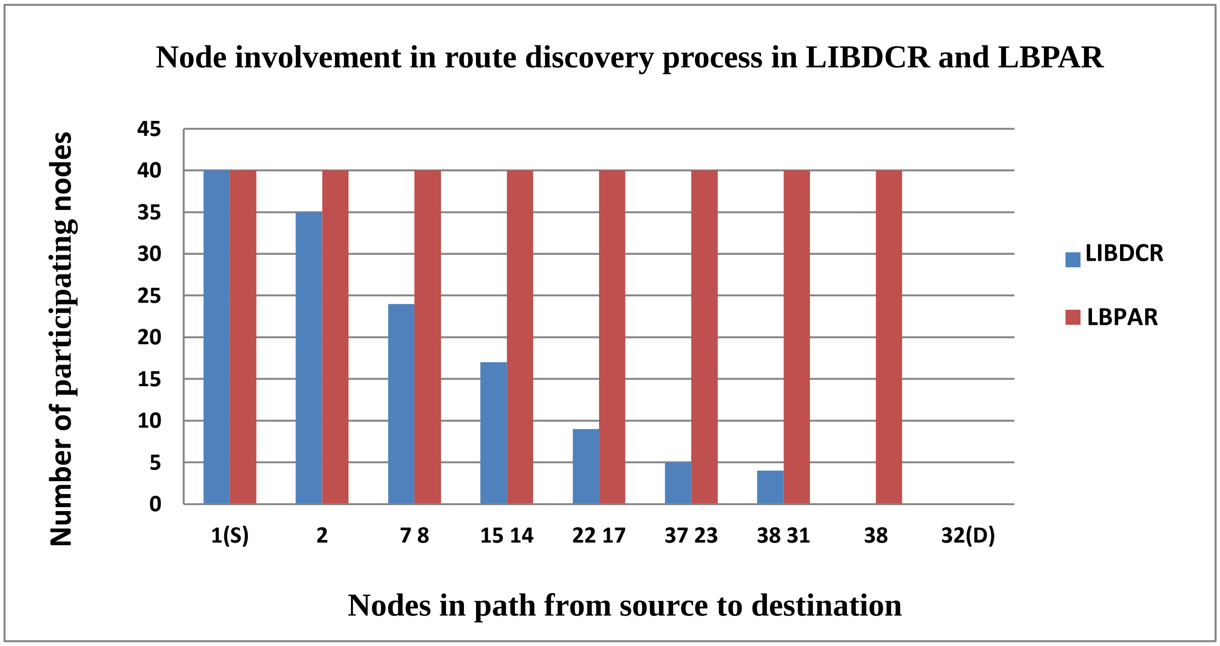


Figure 4.9 : Nodes involvement in path construction process for LIBDCR and LBPAR

The count of number of participating nodes for both protocols is shown in Table 4.5

Table 4.5: Count of participating nodes for path construction

Protocol	**Total Number of participating node**
LIBDCR	40+35+24+17+9+5+4 = **134**
LBPAR	40+40+40+40+40+40+40+40=**320**

So the reduction in control overhead by using LIBDCR is = $\frac{320-134}{320}$ x $100 = 58.125$ %.

It may be observed from Table 4.4 and Fig 4.9, the path construction process LIBDCR engage less number of mobile nodes as compared to LBPAR for the same network topology and source-destination pair.

4.5 SUMMARY

In this work, Location Information Based Destination Converging Routing Method (LIBDCR) has been proposed which is an improvement over LARDAR and LBPAR. LIBDCR reduces the control overhead by narrowing the forwarding region which results in lesser number of participating nodes in route discovery process. LIBDCR is adaptable to destination node's mobility as intermediate node updates the RERQ with fresh location information of destination node. LIBDCR also Provides Quality of Service for real-time traffic like voice and video by selecting the reliable nodes in path having better battery life and band width using the next forwarding node table NFN. A partial route repair method is also introduced which repairs the broken path locally with the help of NFN table and reduces control overhead considerably.

In the next chapter "A Light Weight Efficient Cluster based routing Model for Mobile Ad-hoc Networks (LWECM)" is presented.

CHAPTER 5

A LIGHT WEIGHT EFFICIENT CLUSTER BASED ROUTING MODEL FOR MOBILE AD-HOC NETWORKS (LWECM)

5.1 INTRODUCTION

In an infrastructure less ad-hoc network [70] the flat routings impose great control overhead over the network especially when number of nodes is outsized [71][72][73] and dense. Therefore, flat routing schemes are not scalable. To deal with such problems, hierarchical routing is used [74][75][76]. In hierarchical routing, network is divided into group of neighbouring nodes called clusters [24]. Partitioning the network into clusters reduces the control overhead because a node merely requires knowledge of routing information in its cluster and not of the complete network. Hence cluster based routing makes the large network appear smaller and the topology which is dynamic appear less dynamic. In cluster based routing, every cluster has a node called cluster-head which acts as coordinator and is responsible for intra cluster and inter cluster transmission [25][26][27]. A cluster-head selection policy is used to select a cluster-head while setting up cluster structure. The main objective of a clustering algorithm is to select a healthy and stable cluster-head to elongate the network life and have efficient routing [28]. Cluster members who listen to more than one cluster-head or a member of nearby cluster, act as a gateway node [29][30] and get utilized for inter cluster communication. Rest of the members act as ordinary nodes. An ordinary node is the member of a cluster which is neither the cluster-head nor the gateway node.

5.2 GAPS IN CLUSTER BASED ROUTING MODEL

In mobile ad-hoc networks, clustering based routing methods have shown their importance to perform better network management. Nevertheless clustering based routing has their own side effects and weaknesses due to additional cost in creating and maintaining a cluster structure. Main source of cluster maintenance cost is re-clustering. This is due to the expiration or movement of cluster-head. In cluster based routing, some specific nodes (cluster heads) get overburdened in comparison to other nodes available in the network due to the fact that they have to bear additional responsibility without any additional hardware or software capability. A cluster head has to coordinate among its cluster members and all the traffic (inside the cluster or outside the cluster) passes through the cluster head. Due to this reason, battery power of cluster head drains more rapidly than other nodes in the network and therefore cluster-head becomes the single point of failure [77]. Another source of maintenance overhead is movement of cluster-head from its cluster. When a cluster head leaves its cluster, re-clustering becomes imperative and it may trigger the clustering in the entire network.[78][79][80][81].

5.3 PROPOSED APPROACH

It is clear from the above discussion that the cluster-head plays crucial role in the cluster based routing but is overloaded because of extra responsibilities. The burden of cluster-head can be minimized by letting off the cluster-head from some responsibilities which can be handled by other non-cluster-head members. The proposed work focuses on these problems by introducing “A Light Weight Efficient Cluster based routing Model for Mobile Ad-hoc Networks (LWECM)” along with a novel cluster maintenance procedure. A new type of node is introduced called refugee node in addition to the cluster-head, gateway and ordinary nodes. Refugee nodes come into existence during the cluster maintenance. When a cluster head dies or leaves from its cluster, its member nodes try to affiliate themselves to nearby clusters. When the orphan node is not able to hear any nearby cluster-head but is able to hear a member of a cluster then this orphan node attaches itself to network silently as refugee node through the member node. The idea of refugee node is based on the concept that the re-affiliation is less costly in comparison to re-clustering.

5.3.1 Cluster formation

In cluster formation process, one-hop, non-overlapped clusters are formed. Initially all nodes broadcast AC (acquaint message) containing their unique ID and residual power to their direct neighbours. Packet format for AC messages is shown in Fig. 5.1. All nodes store AC messages received from other nodes. If a node does not receive any AC message with higher residual power than itself, it proclaims itself as a cluster head and announces its status to direct neighbours. If there is tie in nodes residual power then node with lowest ID will act as a cluster-head. Thereafter, all the uncovered neighbours affiliate themselves by setting their cluster-head a winner. Once a node affiliates itself, it will not participate further in clustering process so that non-overlapping of clusters can be formed and also re-affiliation cost can be minimized. This process is repeated until all the nodes in network become cluster-head or member of a cluster. Pseudo code for cluster formation is shown in Fig. 5.2

ID	RP

Figure 5.1: Packet format for acquaint message in LWECM

```
ALGORITHM : CLUSTER FORMATION
V is set of all node in network
$\varphi(v)$ is the set of neighbours of node v
For every $v \in V$
{
  If  $RPv > RPi$  where $i \in \varphi(v)$
      Set v as cluster-head
   else if ($RPv = RPi$ && $IDv < IDi$)
            Set v as cluster-head
         else set i as cluster-head
     For every  $x \in \varphi(v)$
            If (x is uncovered)
            Set  $CLUSTERHEADx = v$ ;
      End for
}End for
```

Figure 5.2: Pseudo code for cluster formation in LWECM

Communication rules of the proposed approach are given in next sub section.

5.3.2 Communication Rules: Proposed approach

In the proposed work, it is assumed that in one hop clustered network, source node S wants to communicate with destination node D. The proposed rules for communication are given below:

1. If destination node D is neighbour of source node S and S is an ordinary node then node S can directly communicate with destination node D without involvement of cluster-head CHS.

2. If the destination node D is not the neighbour of source nodes S and S is an ordinary node then S sends the packet destined to D and its cluster-head. If destination node D is the

member of same cluster then cluster-head may suggest an intermediate node I through which rest of the communication will take place. If no such node is available then communication will take place via cluster-head.

3. If destination node D is not the neighbour of node S and S is an ordinary node then S sends the packet destined to D, via its cluster-head and if node D does not belong to same cluster then cluster-head forward this rout request to all gateway nodes for further path discovery process.

4. If the destination node D belongs to some other cluster and neighbour of a current gateway node G then this gateway node can communicate with D without involvement of D's cluster-head 'CH_D' otherwise G will forward the route request to CHD.

5.3.3 Cluster maintenance

In a cluster based mobile ad-hoc networks a cluster maintenance procedure is required when any node leaves from or joins to a cluster or die due to the lack of resources like battery power. In most of the existing clustering schemes, movement of a single node may initiate the re-clustering in entire network which leads to high maintenance cost. In this work, a low cost maintenance procedure is being proposed with the following cases:

Case 1: When a cluster-head is no longer able to act as cluster-head due to lack of battery power.

In this case, a predefined threshold for battery level to serve as cluster-head is set. When a cluster-head is about to reach this threshold, it broadcasts an alarm message 'ALR' in its cluster and asks the health status of its cluster members. In response to ALR message all members having battery backup greater than threshold, reply with an ALR_RPL message containing their battery level. Packet format for ALR and ALR_RPL is shown in Fig. 5.3 and Fig. 5.4 respectively. Cluster-head then hands over its status to node with maximum battery life. If cluster head does not receive any reply from its members with in certain time limit, it assumes that no member is available to act as a cluster-head and broadcast an 'adjourn' message in its cluster. After receiving this adjourn message, all members including cluster-head try to join neighbouring cluster by affiliation algorithm.

ID	TYPE

Figure 5.3: Packet format for ALR message

ID	RP	TYPE

Figure 5.4 :Packet format for ALR_RPL message

Where ID: Node Id , RP: Residual power , TYPE: CH, Gateway, Member

Case 2: When cluster-head moves from its cluster.

Movement of cluster head does not trigger any routine explicitly in this case. If members of a particular cluster does not hear their cluster-head for certain time, it is assumed that cluster-head has left the cluster and orphan nodes execute affiliation routine for joining another nearby cluster. The affiliation algorithm is given in next section

5.3.4 Affiliation Algorithm for Orphan Nodes

For joining a cluster, a node N broadcasts an entreaty message ‘EN’ to its direct neighbours. On receiving a EN message, all neighbours calculate burden factor ‘BF’ according to equation 5.1 and replies with a ‘EN_reply’ message (packet format for EN and EN_reply are shown in Fig. 5.5 and Fig. 5.6 respectively). If node N receives ‘EN_reply’ from a cluster-head, it sends an ACK message back to that cluster-head. If N receives ‘EN_reply’ from more than one cluster-heads, it selects its cluster-head with lowest ‘BF’ value. If node Ni does not hear any cluster-head and receives ‘EN_reply’ from a member M of a cluster, it joins that cluster as a refugee node until it hears any nearby cluster-head. Ni will communicate with this cluster through node M. If N does not receive any ‘EN_reply’ with in certain time, it declares itself as a cluster-head. Pseudo code of affiliation algorithm is shown in Fig. 5.7.

$$BF=\alpha.(1/RP)+\beta.D \qquad (5.1)$$

Where : BF is burden factor , RP is residual power such that ($RP > 0$), D is degree of node,
α and β are coefficient to adjust the weight of , BF D is the sum of $\alpha + \beta = 1$

Let us assume node Ni wants to affiliate itself by a cluster, it perform following steps.

STEP 1: For joining a nearby cluster a mobile node Ni broadcast an entreaty message ‘ENi’ to its direct neighbours.

STEP 2: On receiving an ‘ENi’ message, all neighbours reply with an ‘EN_reply’ message.

STEP 3: If node Ni receives ‘EN_reply’ from a cluster-head, it sends a ACK message back to that cluster-head. If Ni received ‘EN_reply’ from more than one cluster-head it selects its cluster-head with lowest BF value. If node Ni receives EN_reply from a member M of a cluster, it joins that cluster as a refugee node. Ni will communicate with this cluster through node M. If Ni does not receive any EN_reply for a certain time it declares itself as a cluster-head.

ID	N_STATUS

Figure 5.5: Packet format for EN message

ID	N_STATUS	BF

Figure 5.6: Packet format for EN_RPL message

Where : ID : Node ID , BF : Burden factor, N_STATUS: Node status (011- for cluster-head, 001-ordinary node, 010-gateway node, 000-refugee node)

```
N :   Affiliation seeking node ENR: Array of EN_reply message at node $N_i$
SLC:  contains the ID selected cluster-head/member node LOW: Contains the lowest BF
Algorithm Affiliation ( )
{
  Broadcast EN message in Neighbourhood
  LOW=ENR [0].BF
  SLC=NULL
  For all EN_reply message received at N
        {
               If(ENR[i].status == 011)
               ENR[i].BF < LOW
               SLC=ENR[i].ID // ID of CH with smaller  BF
        }
           If (SLC!=NULL)
               Send ACK to SLC.
               Set status to 00;  Break; // set status to member node
        For all EN_reply message received at N
        {
                If (ENR[i].status==001|| ENR[i].status==010)
                ENR[i].BF < LOW
                SLC=ENR[i].ID      // ID of member node with smaller BF value
        }
           If (SLC!=NULL)
                Send ACK to SLC.
                Set status to 000 ; Break; // set status to refugee node

    Set status to 011; // set status to cluster-head
}
```

Figure 5.7: Pseudo code of affiliation algorithm

5.3.5 Illustration

Fig. 5.8 shows the cluster structure where nodes A, G and M represent cluster-heads, node E, K represent gateway nodes, node O represents refugee node and rest of the nodes are ordinary nodes. Radio links between nodes and the boundary of clusters are represented by dashed and dotted lines respectively.

Let us suppose node B wants to communicate with node P, then B can directly communicate with P without involvement of its cluster-head A as P is in radio range of node B. Similarly if node J wants to communicate with node L, it sends the packet to its cluster-head G, thereafter G sends this packet to its gateway nodes E and K as L is not the member of G. Now node K locates the L as L is in radio range of K and will act as a last hop from J to L for further communication, without the involvement of L's cluster head M .

On other hand, node O does not belong to any cluster, it is joined with network as refugee node through node F without disturbing the existing topology.

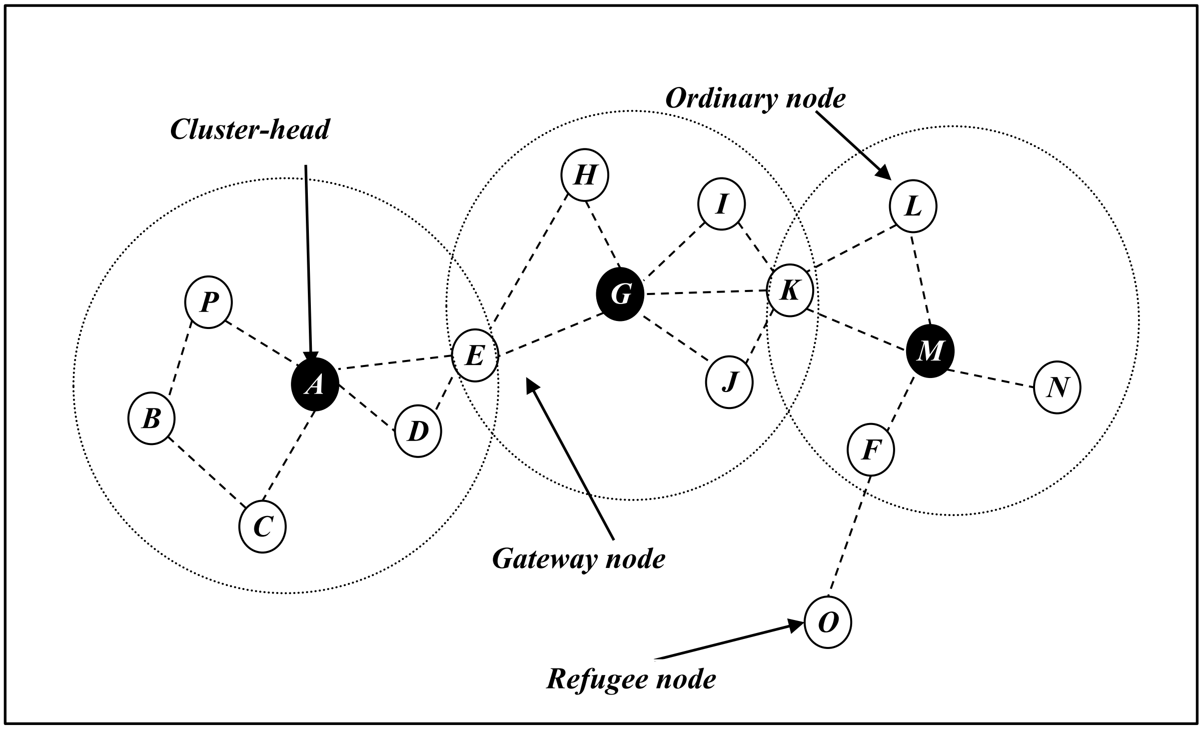


Figure 5.8: Cluster Structure in LWECM

5.4 SIMULATION SETUP AND RESULT ANALYSIS

In this section, path construction mechanism of existing standard Cluster Based Routing Protocol (CBRP)[31] and proposed approach “Light Weight Efficient clusters based routing Model for Mobile Ad-hoc Networks (LWECM)” have been simulated and compared using MATLAB 7.8.0.

The MATLAB has been selected as simulation environment for this work, due to its ability of algorithm development with numerical computation. Moreover the MATLAB has vast plotting functions library and allows the researches to develop their own graphical application and user interface with customized functionality.

For ad-hoc network routing protocol’s simulation, there are mainly two systems for assessment of certain protocol. First system tests same protocol over the different network topologies to assess the behavior of that protocol on different loads and scale. Whereas the second system performs simulation of different protocols over the same network topology to compare the behavior and efficiency of these protocols.

For simulation and assessment of this work, second approach has been followed. In this arrangement, 21 mobile nodes are deployed randomly in 200 X 200 meter square area. Every mobile node has a unique ID tagged with it and varies from 1 – 21. The radio range of every mobile node has been kept 20 meter. For simplicity, battery of each node is considered of same capacity (mAH) and outstanding battery power at every node varies from 1 to 100 percent.

It is assumed that every mobile node is able to determine the presence of its direct neighbors. For communication between two adjacent nodes a bidirectional link must exist. For comprehensive simulation, following four cases of path construction have been identified regarding all possible situations of the presence of source node and destination node in inter cluster communication and intra cluster communication.

CASE I: Path construction for intra-cluster communication where source and destination are in the each other’s radio range as shown in Fig. 5.9(a-f) for CBRP and Fig. 5.10(a) to 5.10(e) for LWECM.

CASE II: Path construction for intra-cluster communication where source and destination are not in the each other’s radio range as shown in Fig. 5.11(a) to 5.11(f) for CBRP and Fig. 5.12(a) to 5.12(g) for LWECM

CASE III: Path construction for inter-cluster communication where source and destination are not in the each other’s radio range as shown in Fig. 5.13(a) to 5.13(j) for CBRP and Fig. 5.14(a) to 5.14(j) for LWECM.

CASE IV: Path construction for inter-cluster communication where source and destination are in the each other’s radio range as shown in Fig. 5.15(a) to 5.15(g) for CBRP and Fig. 5.16(a) to 5.16(c) for LWECM.

Simulation findings from the path construction of CBRP and LWECM has been compared and comprehended in table 5.1. The aim of this work is to improve the life of cluster-heads and reduce path length between communicating parties.

This comparison is based on number of cluster-heads involved in a constructed path and hop count of path. For fair comparison of Cluster Based Routing Protocol (CBRP) and Light Weight Efficient clusters based routing Model for Mobile Ad-hoc Networks (LWECM), the network topology and scenario, have been kept identical. Moreover the source and destination pair have been kept same for simulation.

In the following figures, three cluster- heads 1, 7 and 13 are represented by bold black small circle, the cluster boundaries of these cluster-heads have been represented by big solid-lined-circle, gateway nodes have been represented by grey small circles, and ordinary nodes have been represented by white small circles. Flows of information packets have been represented by unidirectional black arrow and communication links between nodes have been represented by bidirectional black arrow.

CASE I: Path construction for intra-cluster communication where source and destination are in the each other’s radio range.

Figure 5.9 (a-f) show the simulation stages of path construction in CBRP for case I. The source node is 1 and destination node 3.

Stage1: Arbitrary topology of an mobile ad-hoc network, as shown in Fig. 5.9(a)

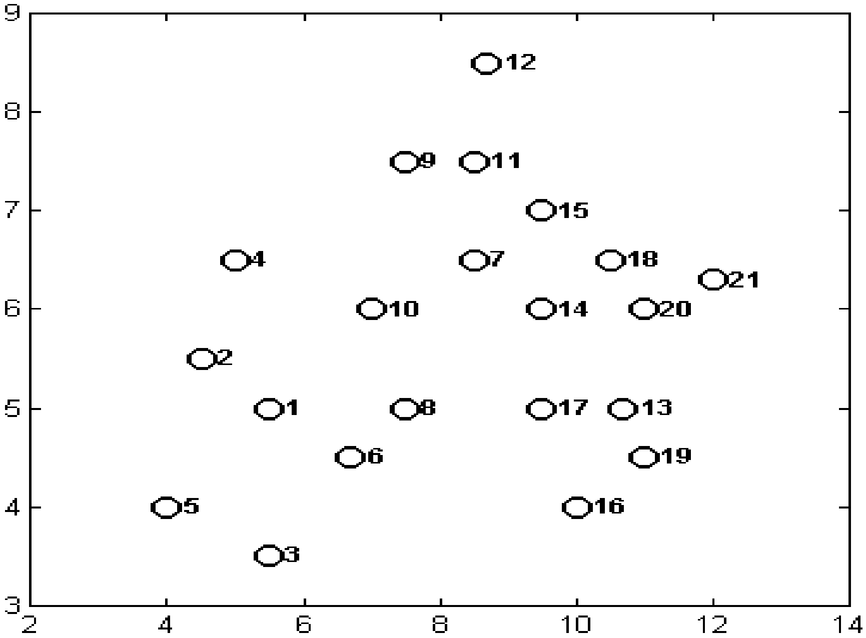


Figure 5.9.(a): Stage 1 of CBRP in case I

Stage 2: 3 well-formed clusters with cluster head 1,7,13, Nodes in grey color are gateway nodes, as shown in Fig. 5.9(b)

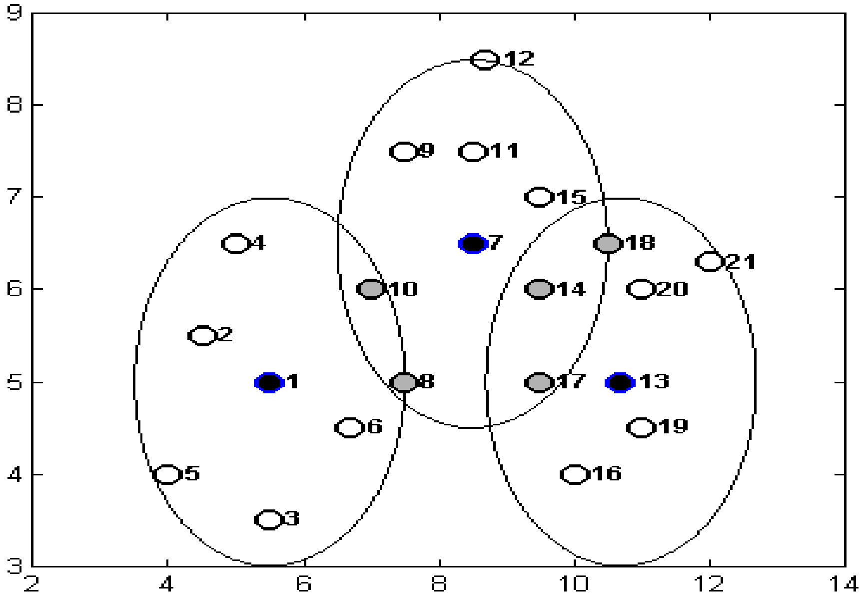


Figure 5.9.(b): Stage 2 Stage 1 of CBRP in case I

Stage 3 Source node 5 seeks the path to destination 3, as shown in Fig.5.9(c).

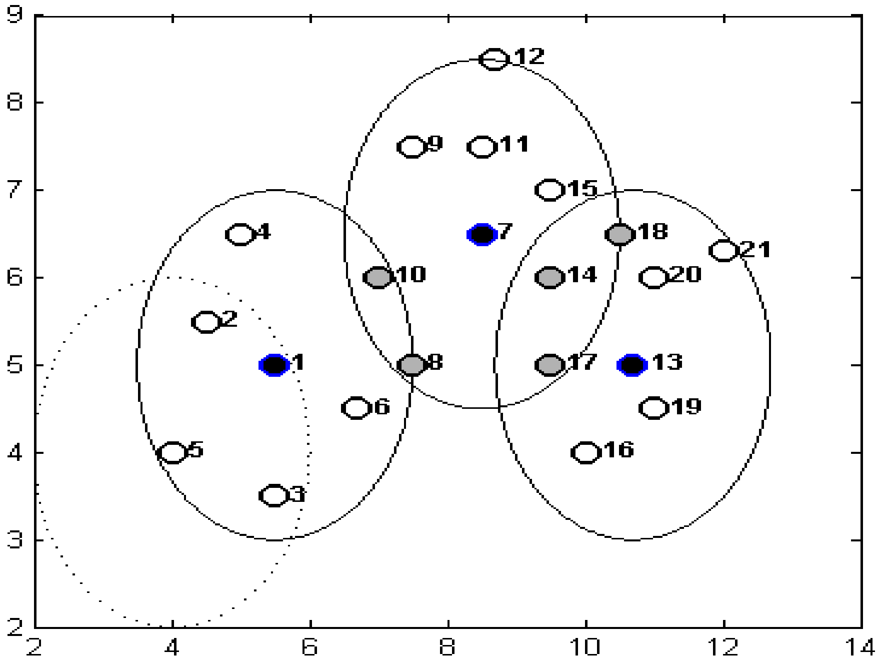


Figure 5.9(c): Stage 3 Stage 1 of CBRP in case I

Stage 4: Source node 5 sends request to its cluster head 1, as shown in Fig. 5.9(d).

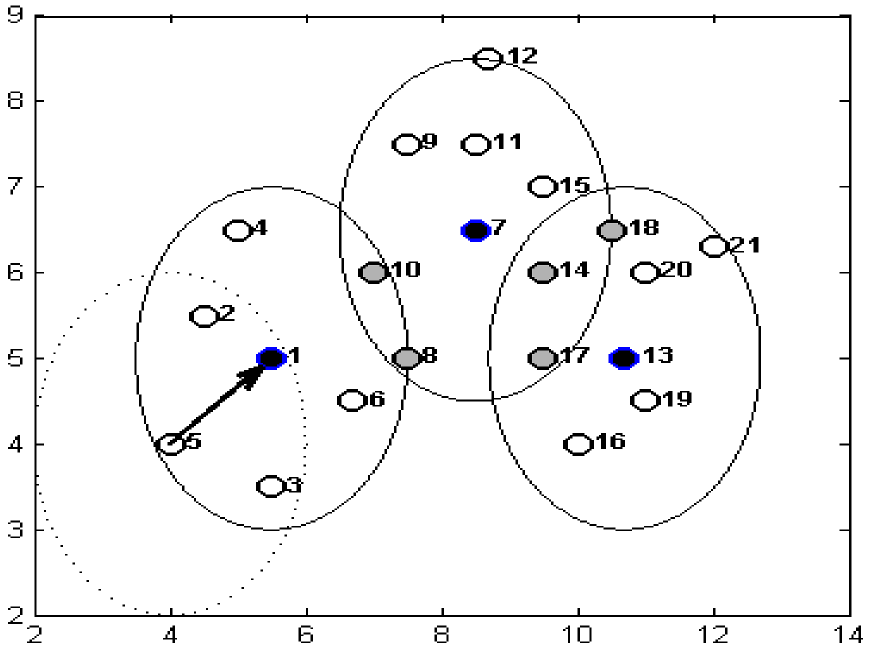


Figure 5.9(d): Stage 4 Stage 1 of CBRP in case I

Stage 5: Cluster head 1 locates the destination 3 in its own cluster and forward the request, as shown in Fig. 5.9(e).

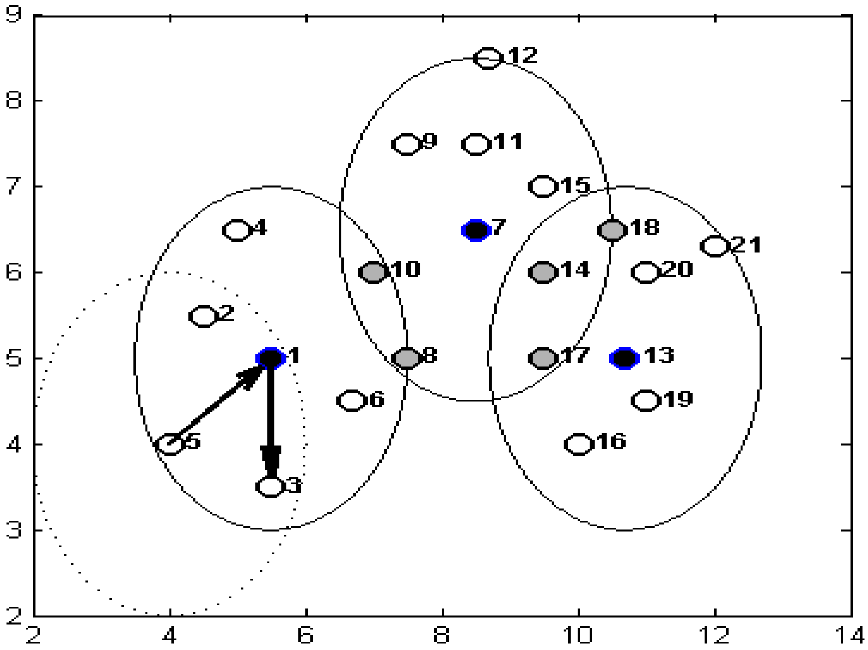


Figure 5.9(e): Stage 5 Stage 1 of CBRP in case I

Stage 6: Destination responds with acknowledgement which passes through reverse path and path is established, as shown in Fig. 5.9(f).

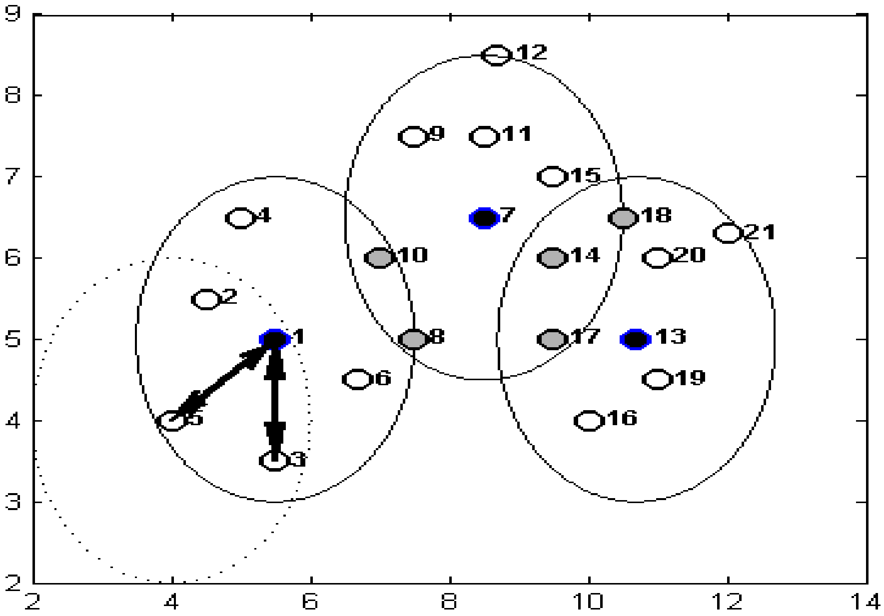


Figure 5.9.(f): Stage 6 Stage 1 of CBRP in case I

Fig. 5.10 (a-d) show the simulation stages of path construction in LWECM. For case I. The source node is 1 and destination node 3.

Stage 1: 3 well-formed clusters with three cluster head 1,7,13. Nodes in grey color are gateway nodes, as shown in Fig. 5.10(a).

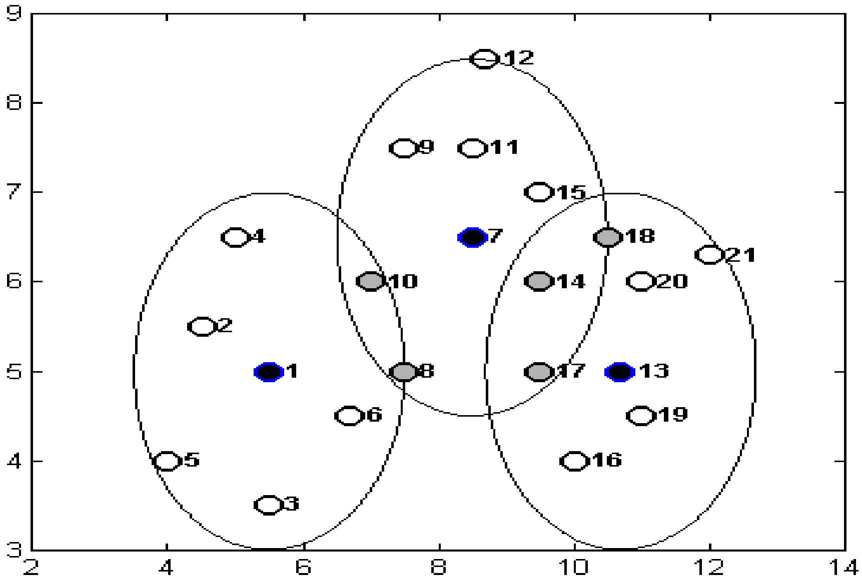


Figure 5.10(a): Stage 1 of LWECM in case I

Stage 2: Source node 5 seeks the path to destination Node 3, as shown in Fig. 5.10(b).

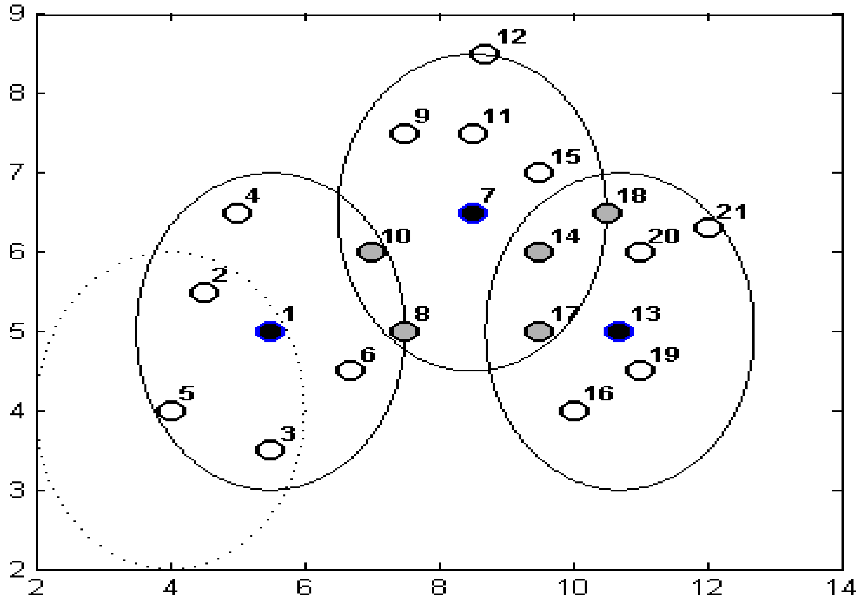


Figure 5.10(b): Stage 2 of LWECM in case I

Stage 3: As the destination 3 is in the radio range of source 5 node, it sends the request direct to destination without involvement of cluster head 1, as shown in Fig. 5.10(c).

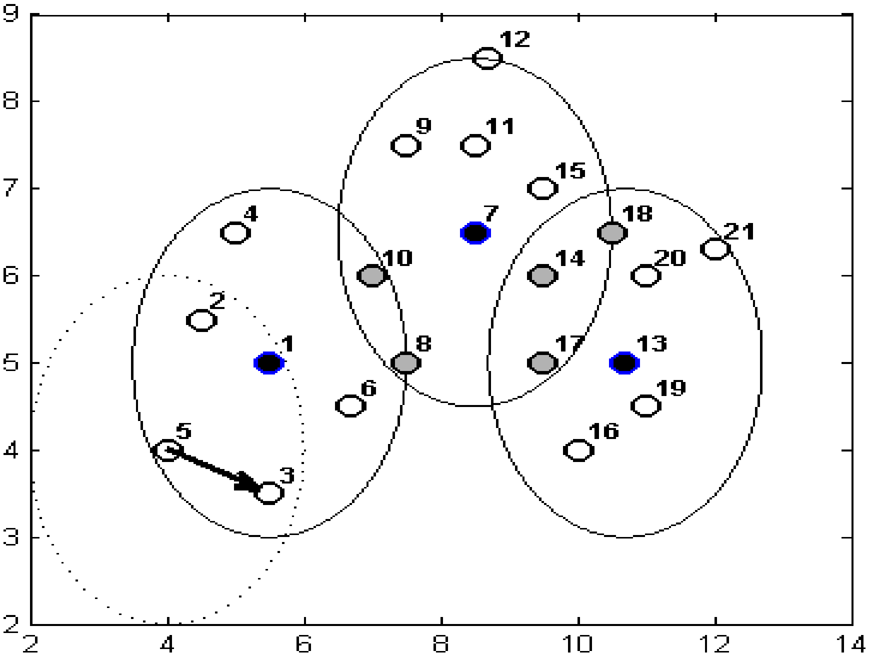


Figure 5.10(c): Stage 3 of LWECM in case I

Stage 4: Path is established with reverse acknowledgment between node 5 and 3 as shown in Fig. 5.10(d).

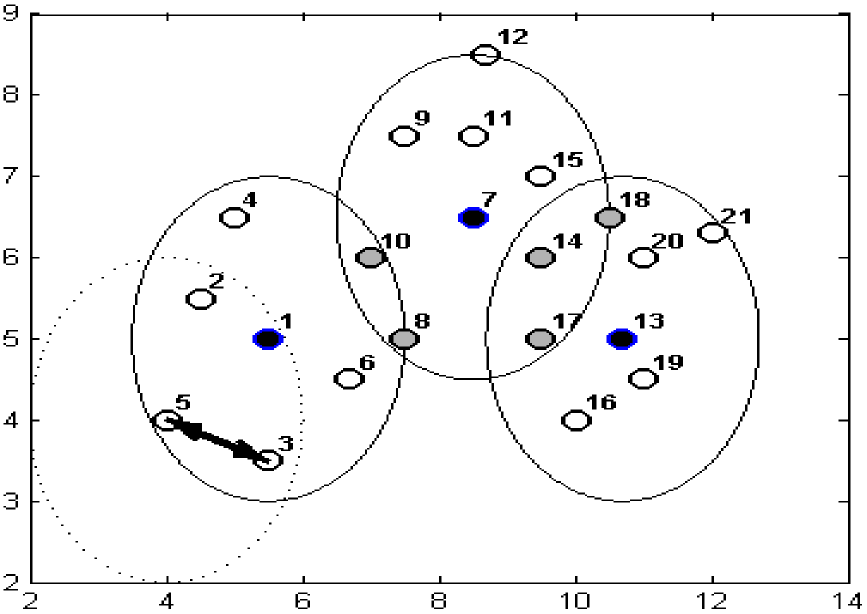


Figure 5.10(d): Stage 4 of LWECM in case I

CASE II: Path construction for intra-cluster communication where source and destination are not in the each other’s radio range.

Fig. 5.11 (a-e) show the simulation stages of path construction in CBRP for case II. The source node is 3 and destination node 4.

Stage 1: 3 well-formed clusters with cluster head 1,7,13. Nodes in grey color are gateway nodes as shown in Fig. 5.11(a)

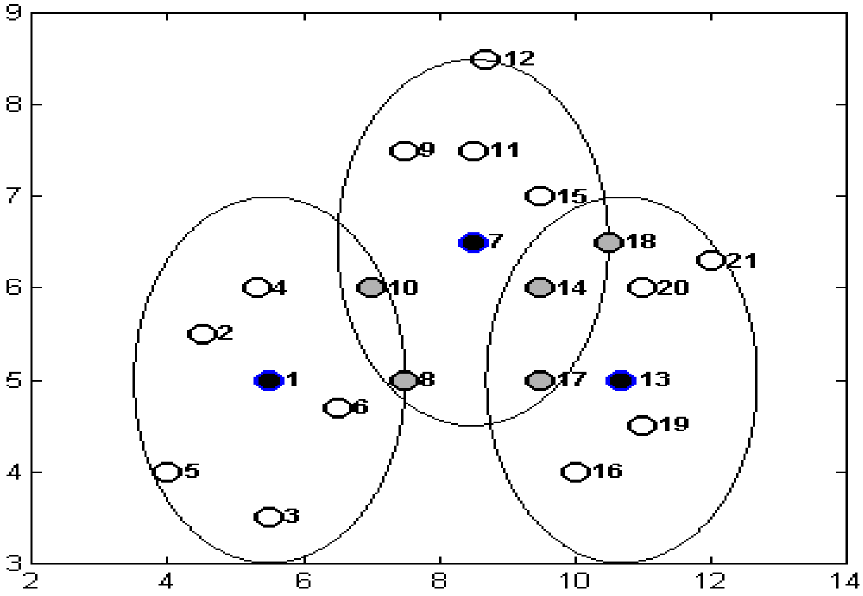


Figure 5.11(a): Stage 1 of CBRP in case II

Stage 2: source node 3 seeks the path to destination 4, as shown in Fig. 5.11(b)

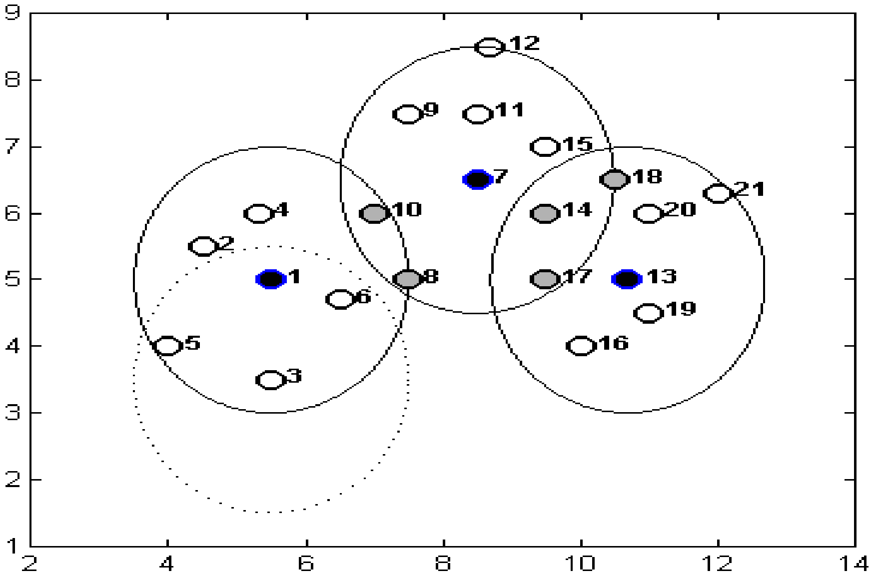


Figure 5.11(b): Stage 2 of CBRP in case II

Stage 3: Source node 3 sends request to cluster head 1, as shown in Fig. 5.11(c)

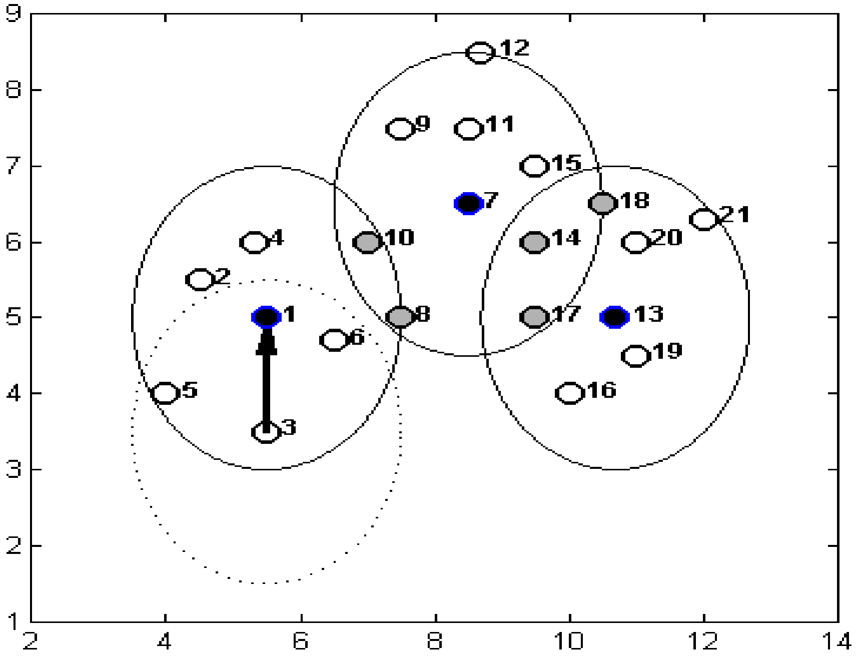


Figure 5.11(c): Stage 3 of CBRP in case II

Stage 4: Cluster head 1 locates the destination 4 in its own cluster and forward the request, as shown in Fig. 5.11(d)

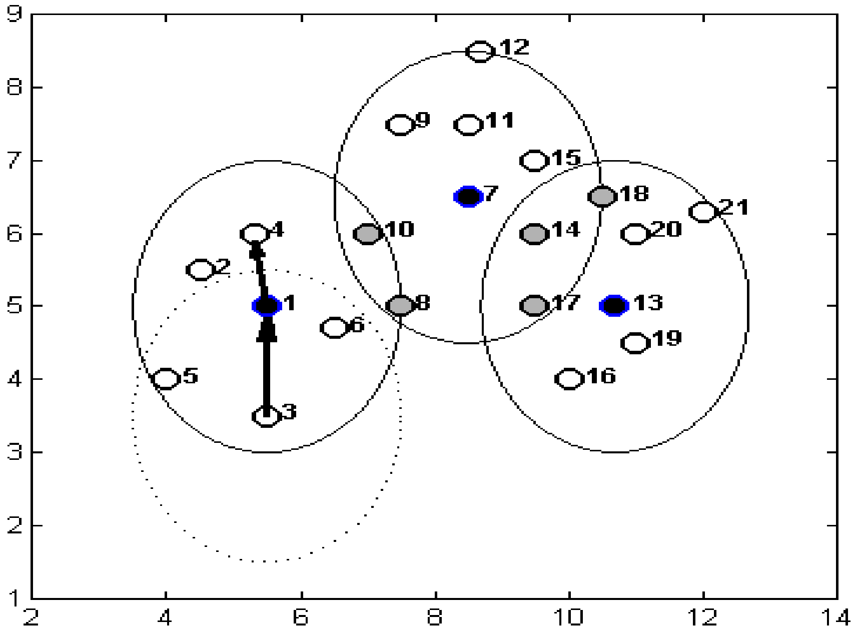


Figure 5.11(d): Stage 4 of CBRP in case II

Stage 5: Destination sends the acknowledgment on reverse path and path established, as shown in Fig. 5.11(e)

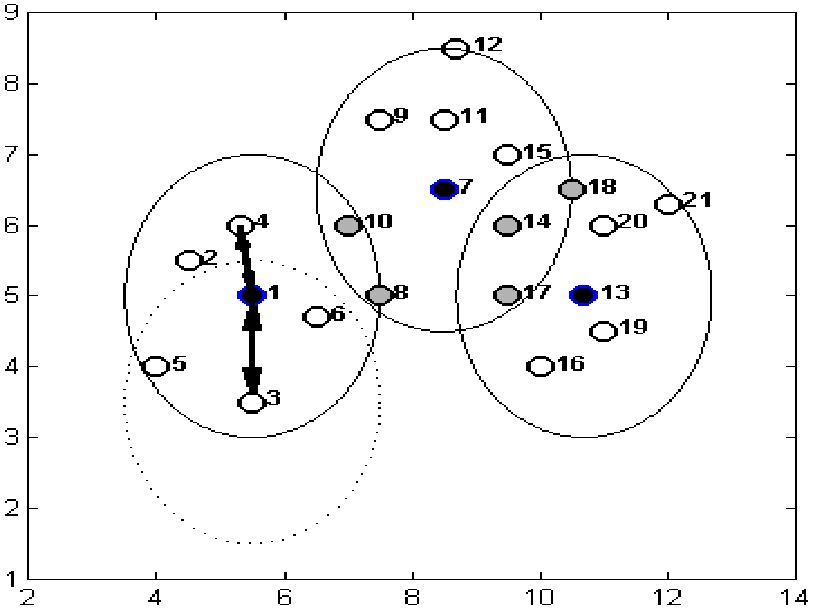


Figure 5.11(e): Stage 5 of CBRP in case II

Figure 5.12 (a-g) show the simulation stages of path construction in LWECM for case II the source node is 3 and destination node 4.

Stage1: 3 well-formed clusters with three cluster head 1, 7, 13 nodes in grey color are gateway nodes, as shown in Fig. 5.12(a).

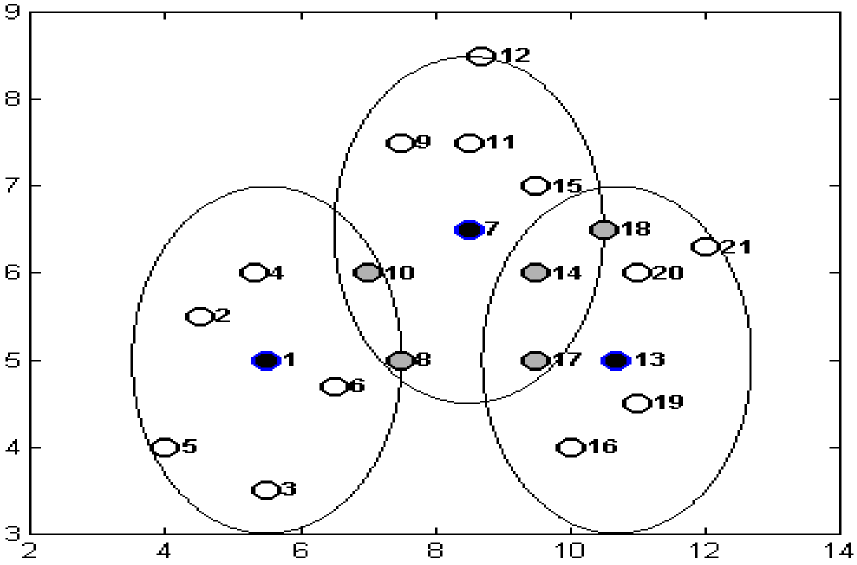


Figure 5.12(a): Stage 1 of LWECM in case II

Stage 2: Source node 3 seeks the path to destination 4, as shown Fig. 5.12(b)

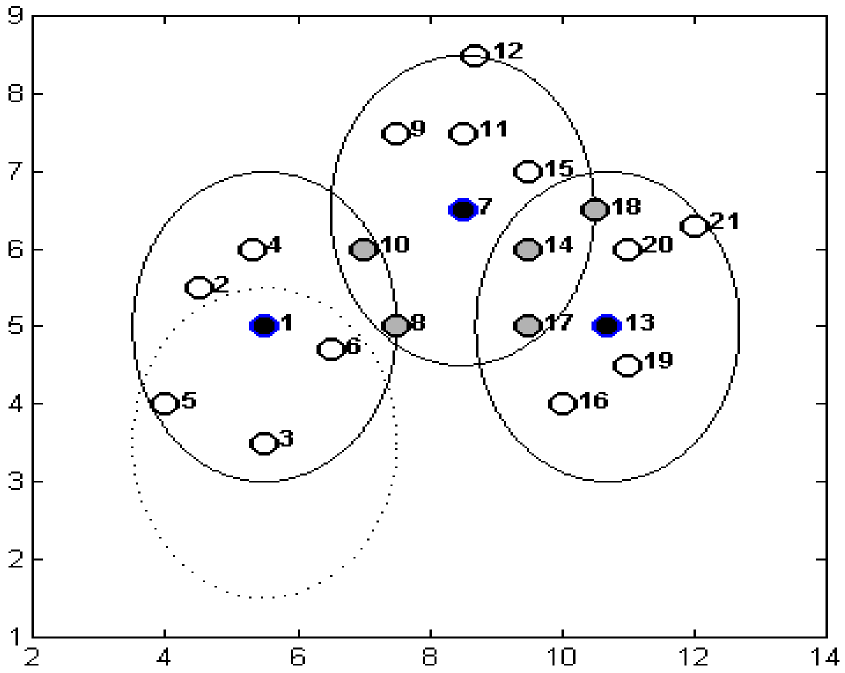


Figure 5.12(b): Stage 2 of LWECM in case II

Stage 3: Source node 3 sends request to cluster head 1, as shown Fig. 5.12(c)

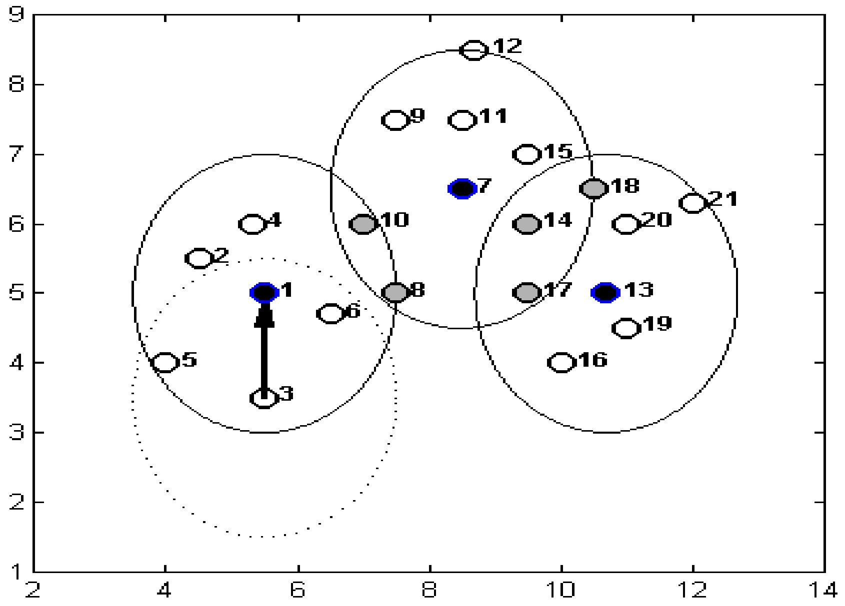


Figure 5.12(c): Stage 3 of LWECM in case II

Stage 4: Cluster head appoints an intermediate node 6 and send information back to the source 3 for path construction, as shown Fig. 5.12(d).

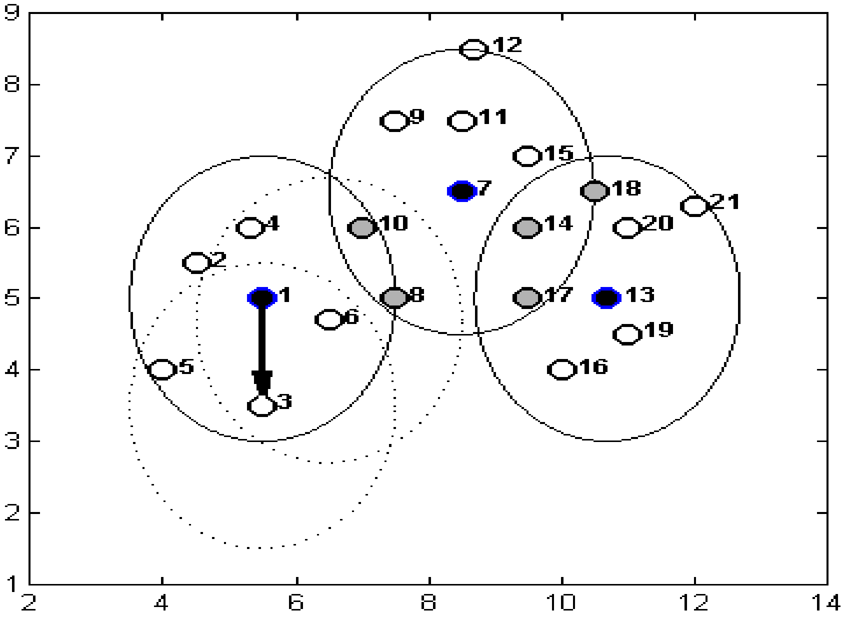


Figure 5.12(d): Stage 4 of LWECM in case II

Stage 5 Source then sends the request to appointed intermediate node 6, as shown Fig. 5.12(e).

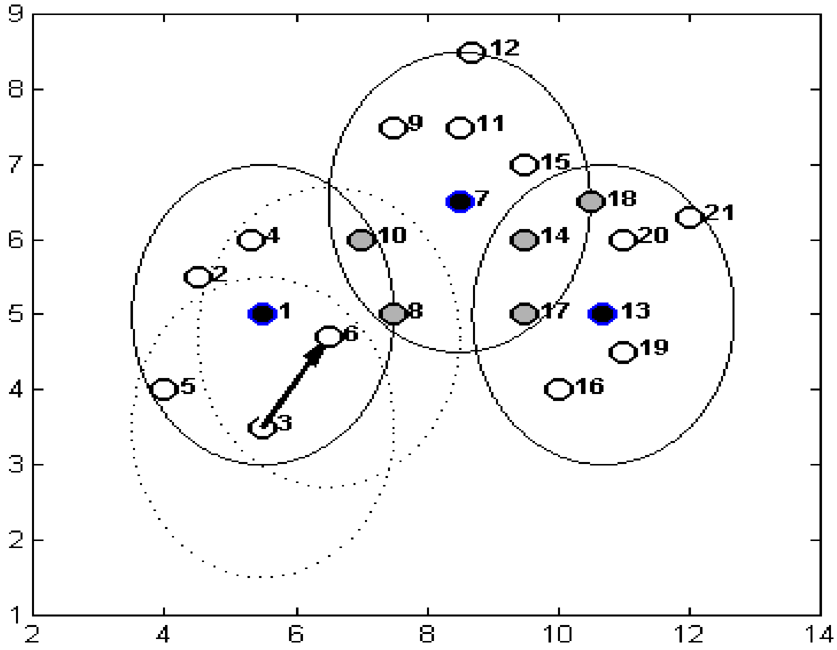


Figure 5.12(e): Stage 5 of LWECM in case II

Stage 6 : Node 6 then forwards the request to destination , as shown Fig. 5.12(f).

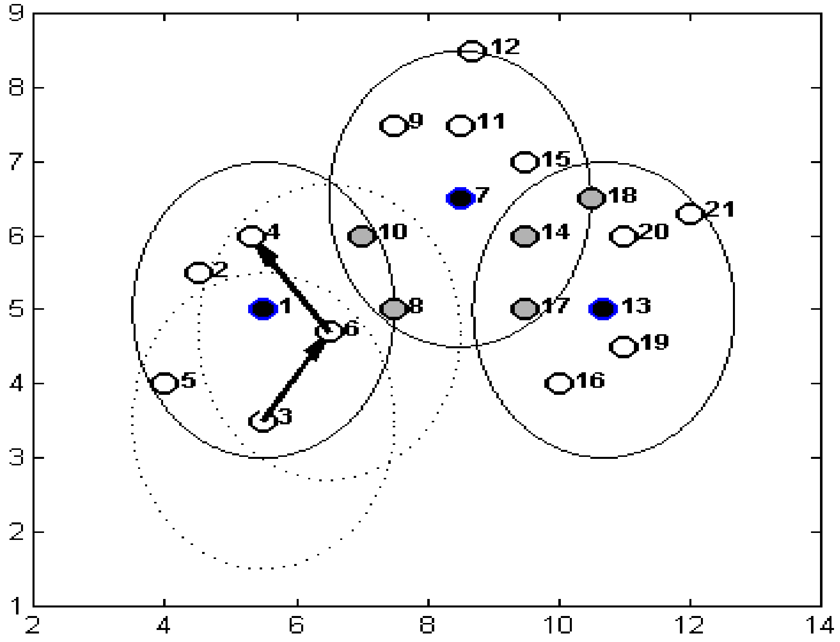


Figure 5.12(f): Stage 6 of LWECM in case II

Stage 7: Destination 4 responds with acknowledgment to node 6, node 6 forward acknowledgments to the source node 3 and path established, as shown Fig. 5.12(g).

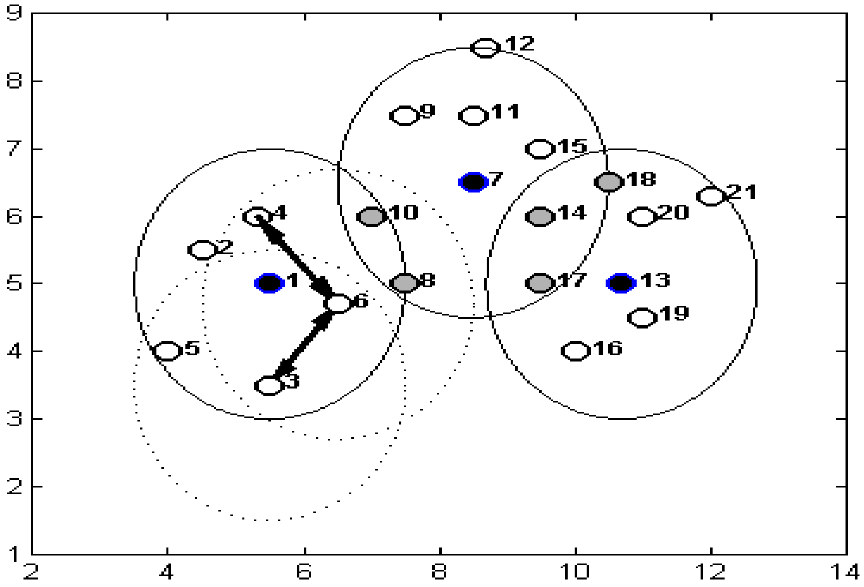


Figure 5.12(g): Stage 7 of LWECM in case II

CASE III: Path construction for inter-cluster communication where source and destination are not in the each other's radio range.

Figure 5.13 (a-h) show the simulation stages of path construction in CBRP for case III. The source node is 5 and destination node 20.

Stage 1: 3 well-formed clusters with three cluster head 1,7,13, nodes in grey color are gateway nodes as shown in Fig. 5.13(a).

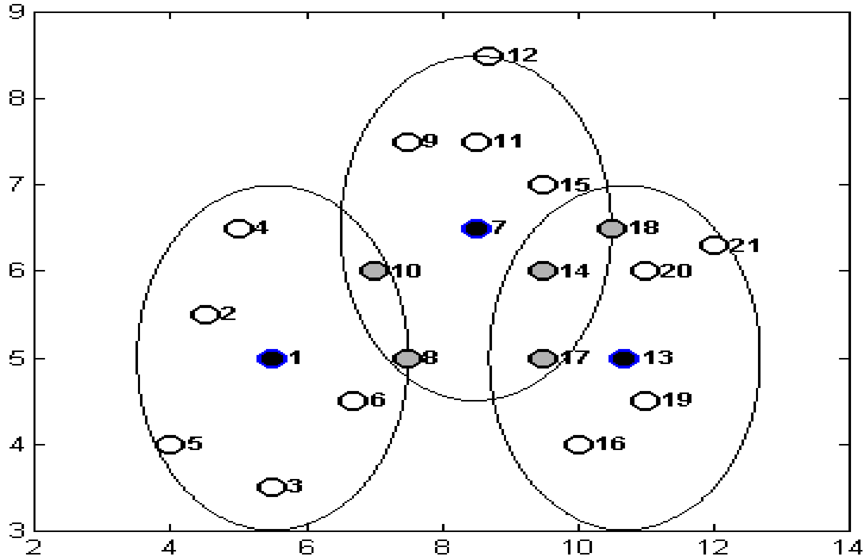


Figure 5.13(a): Stage 1 of CBRP in case III

Stage 2: Source node 5 initiates the path construction for destination 20 and send request its cluster-head 1, as shown in Fig. 5.13(b).

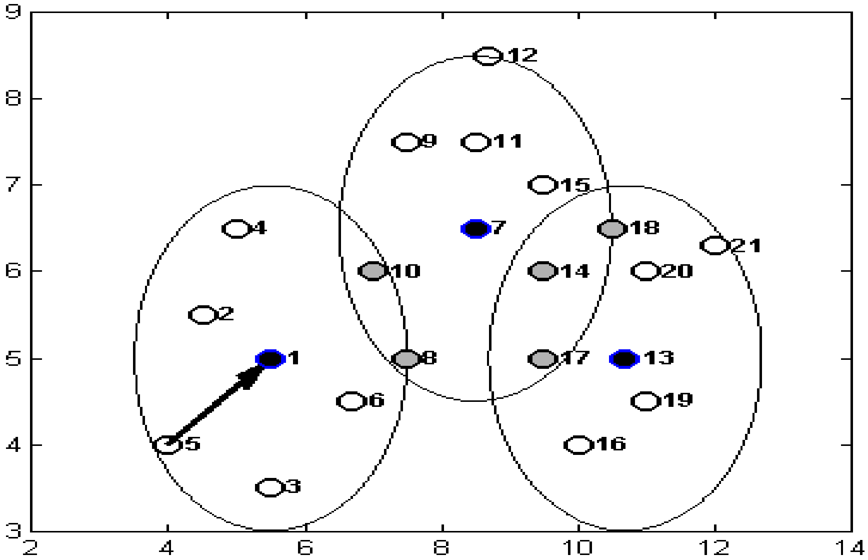


Figure 5.13(b): Stage 2 Stage 1 of CBRP in case III

Stage 3: Cluster head 1 determines the absence of destination 20 and forward the request packet to gateway node 10, as shown in Fig. 5.13(c).

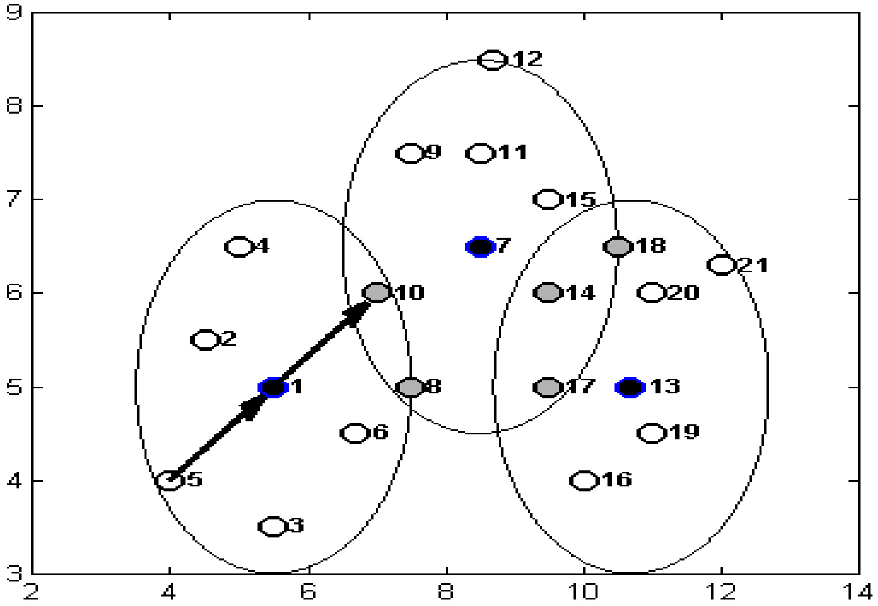


Figure 5.13(c): Stage 3 Stage 1 of CBRP in case III

Stage 4: Gateway node 10 forwards the request packet to nearby cluster head 7, as shown in Fig. 5.13(d).

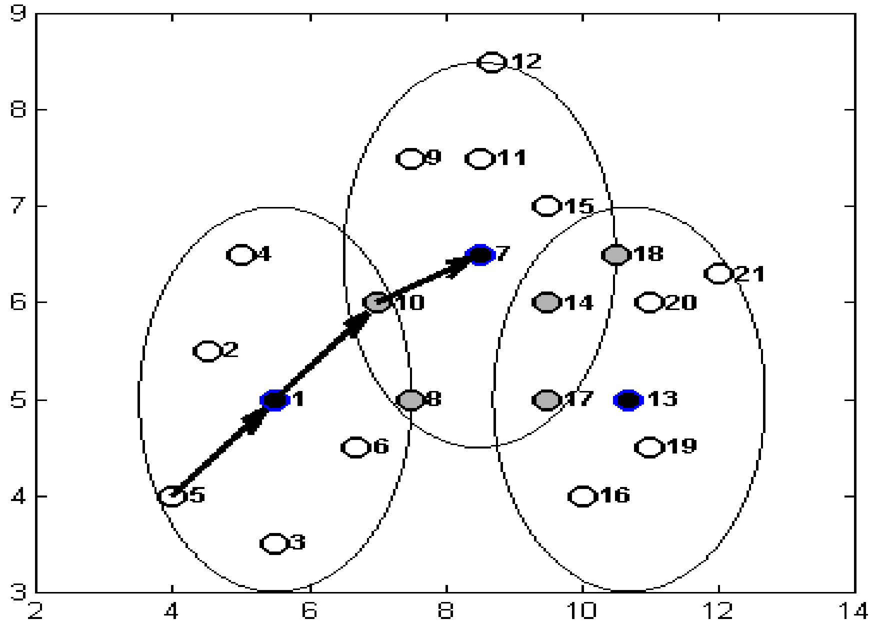


Figure 5.13(d): Stage 4 Stage 1 of CBRP in case III

Stage5: Cluster head 7 determines the absence of destination 20 and forward the request packet to gateway node 14, as shown in Fig. 5.13(e).

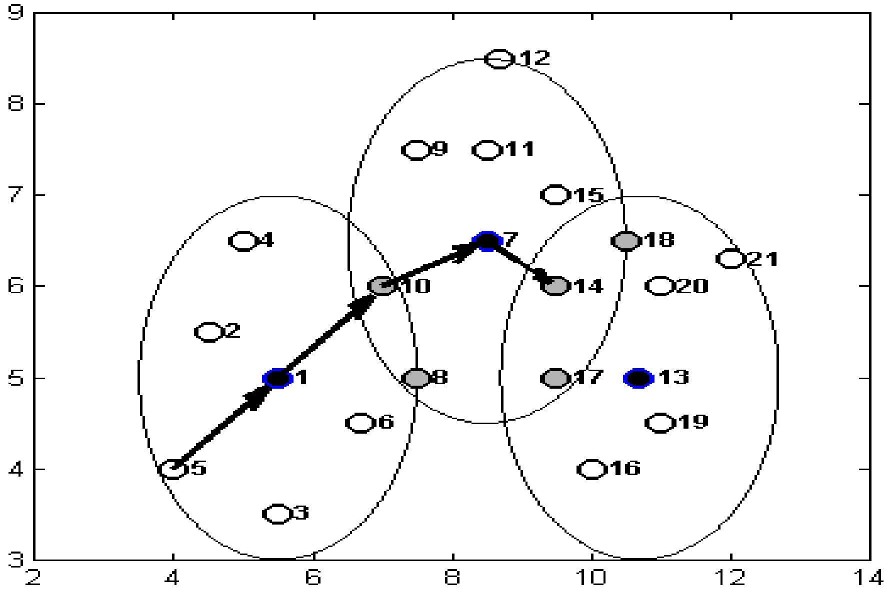


Figure 5.13(e): Stage 5 Stage 1 of CBRP in case III

Stage 6: Gateway node 14 forward the request packet to nearby cluster head 13, as shown in Fig. 5.13(f).

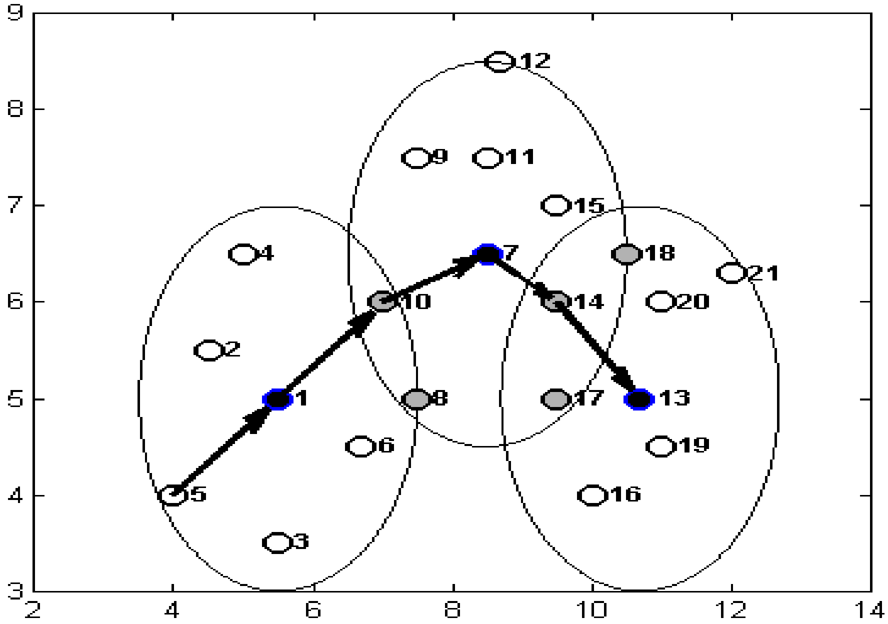


Figure 5.13(f): Stage 6 Stage 1 of CBRP in case III

Stage 7: cluster head 13 determines the presence of destination 20 and forward the request packet to 20, as shown in Fig. 5.13(g).

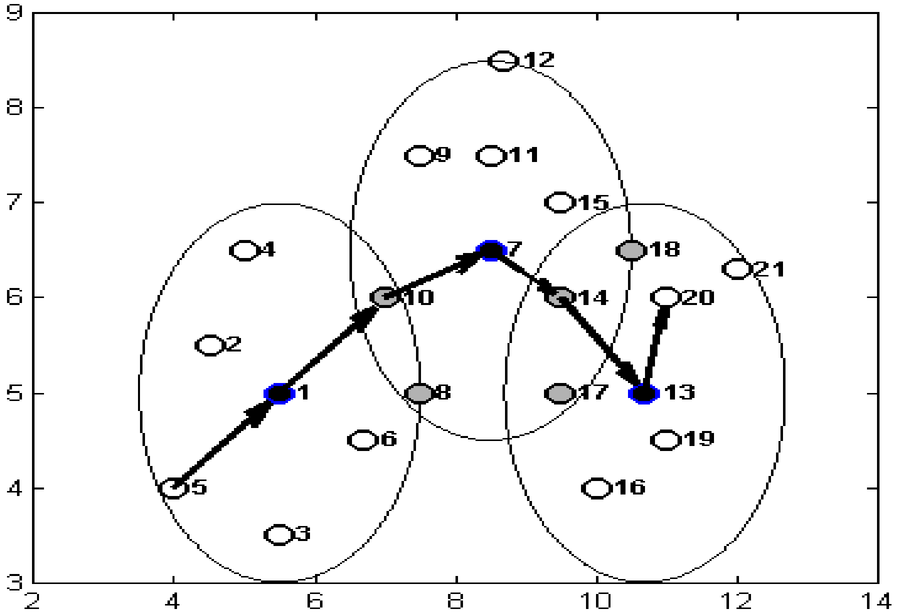


Figure 5.13(g): Stage 7

Stage 8: Destination 20 sends acknowledgment to its cluster head 17 Acknowledgment travels along the reverse path back to the source node 1 and path get established, as shown in Fig. 5.13(h).

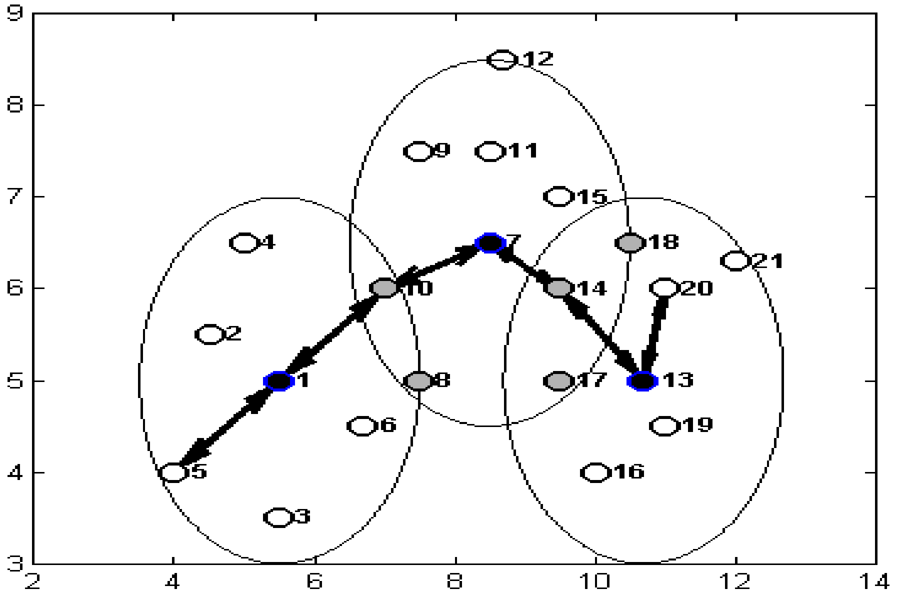


Figure 5.13(h): Stage 8

Figure 5.14 (a-h) shows the simulation stages of path construction in LWECM for Case 3 the source node is 1 and destination node 20.

Stage 1: 3 well-formed clusters with three cluster head 5,10,17, nodes in grey color are gateway nodes, as shown in Fig. 5.14(a)

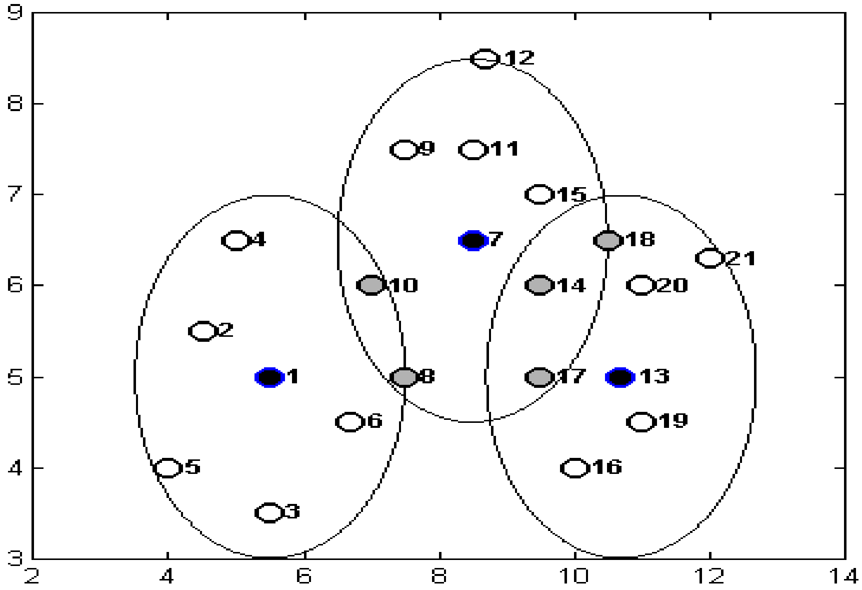


Figure 5.14(a): Stage 1 of LWECM in case III

Stage 2: Source node 5 determines the absence of destination in it radio range and sends route request packet to its cluster-head 1, as shown in Fig. 5.14(b)

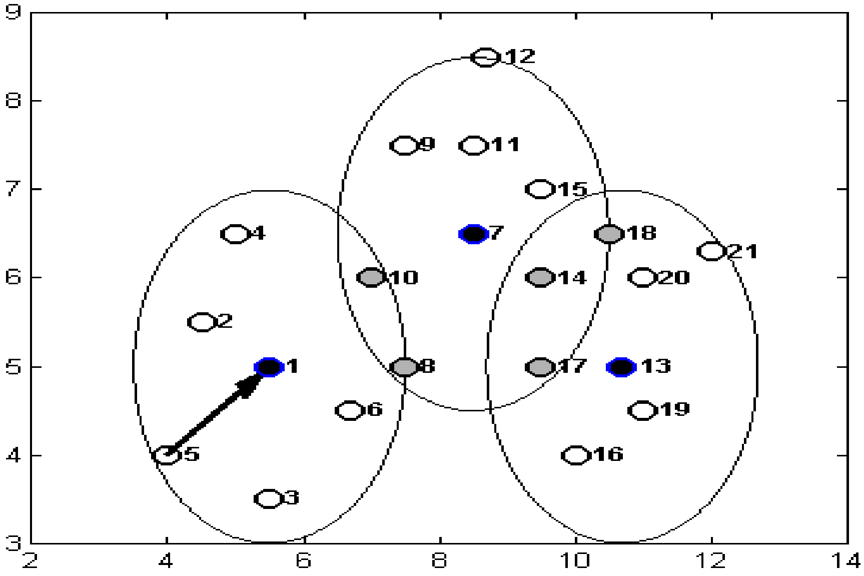


Figure 5.14(b): Stage 2 of LWECM in case III

Stage 3: Cluster head 1 determines the absence of destination 20 and send the route request packet to gateway node 10, as shown in Fig. 5.14(c)

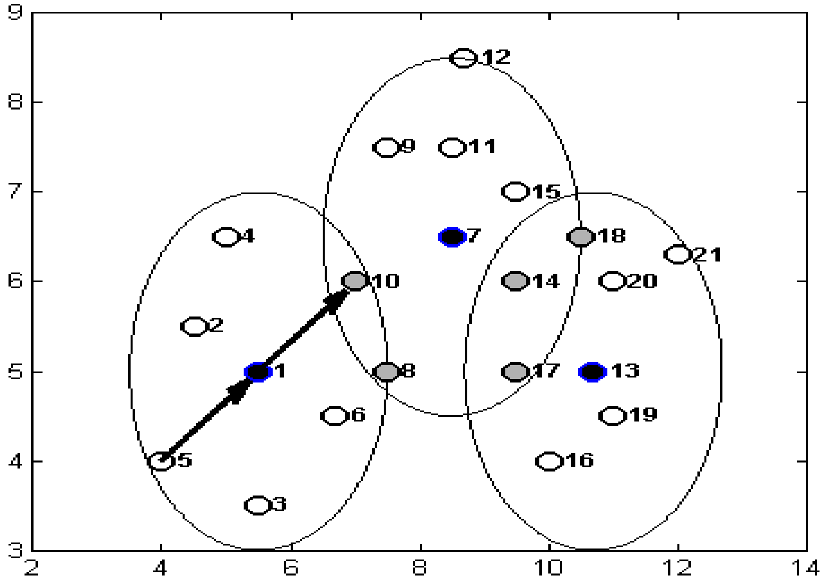


Figure 5.14(c): Stage 3 of LWECM in case III

Stage 4: Gateway node 10 determines the absence of destination 20 in its radio range and forwards the route request packet to nearby cluster head 7, as shown in Fig. 5.14(d).

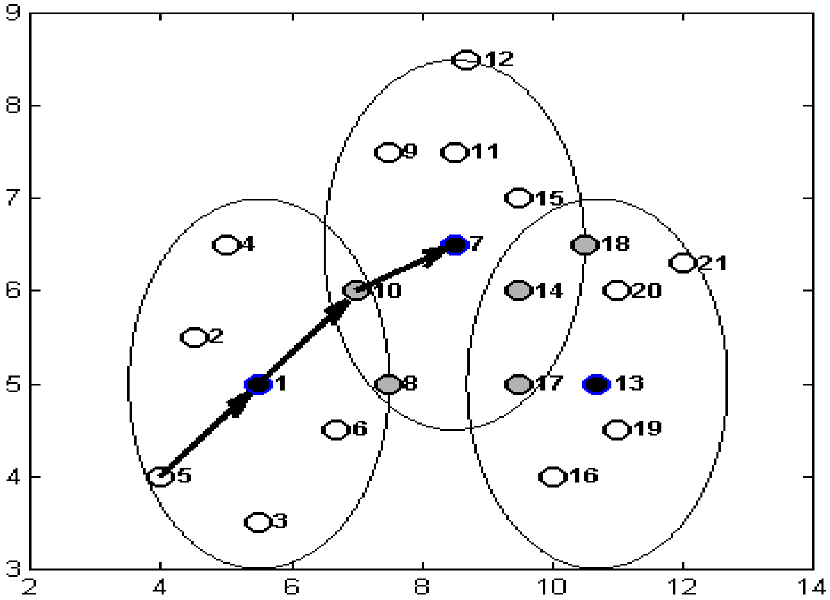


Figure 5.14(d): Stage 4 of LWECM in case III

Stage 5: Cluster head 7 determines the absence of destination 20 and send the route request packet to gateway node 14, as shown in Fig. 5.14(e).

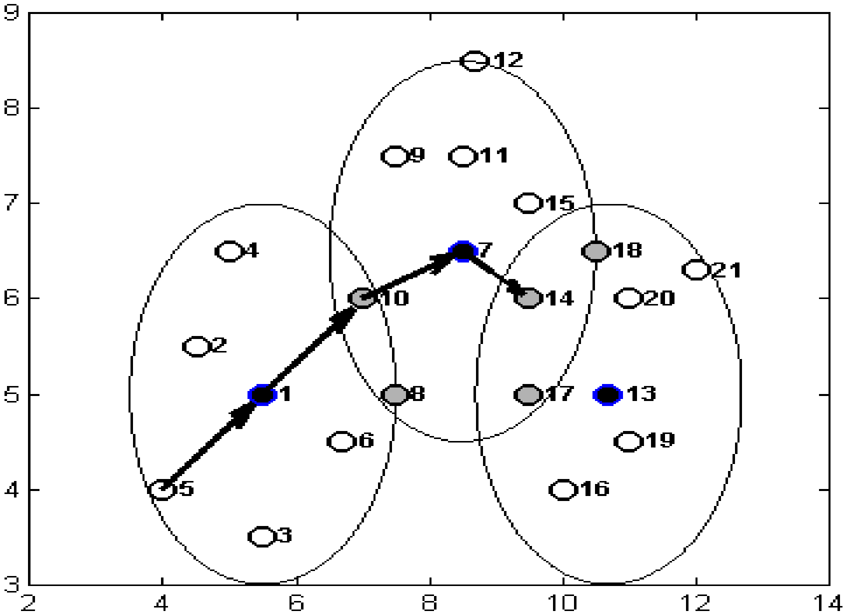


Figure 5.14(e): Stage 5 of LWECM in case III

Stage 6: Gateway node 14 determines the presence of destination 20 in its radio range. and forward the route request packet to destination 20, as shown in Fig. 5.14(f).

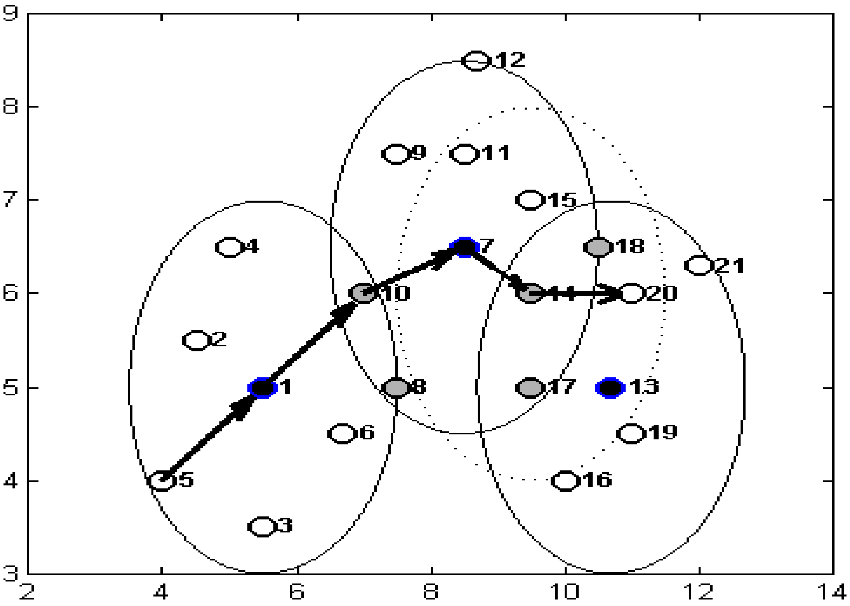


Figure5.14 (f): Stage 6 of LWECM in case III

Stage 7: Destination 20 then sends the acknowledgment packet to gateway 14, as shown in Fig. 5.14(g).

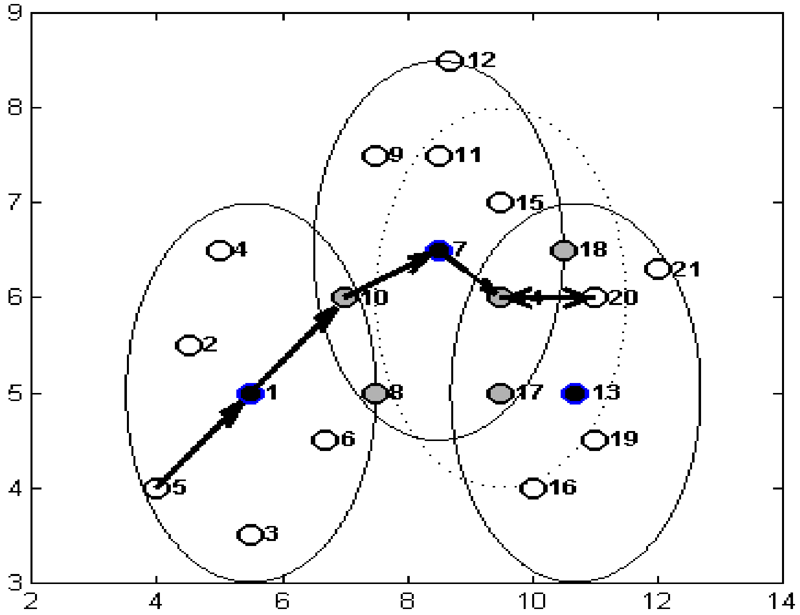


Figure 5.14(g): Stage 7 of LWECM in case III

Stage 8: This way Acknowledgment packet travel along the reverse path back to the source 1 and path get established, as shown in Fig. 5.14(h).

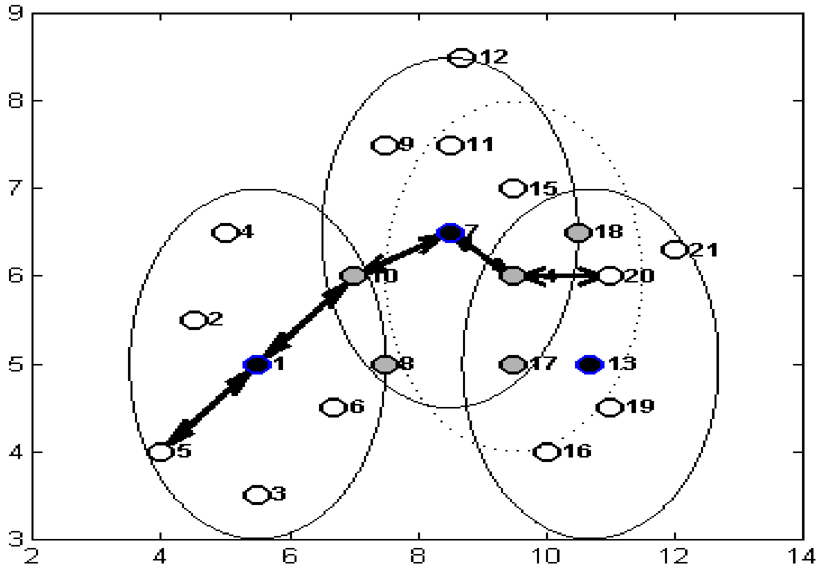


Figure 5.14(h): Stage 8 of LWECM in case III

CASE VI: Path construction for inter-cluster communication where source and destination are in the each other's radio range.

Figure 5.15 (a-g) show the simulation stages of path construction in CBRP for case VI. The source node is 17 and destination node 15.

Stage1: 3 Well-formed clusters with three cluster head 1,7,13. Nodes in grey color are gateway nodes, as shown in Fig. 5.15(a).

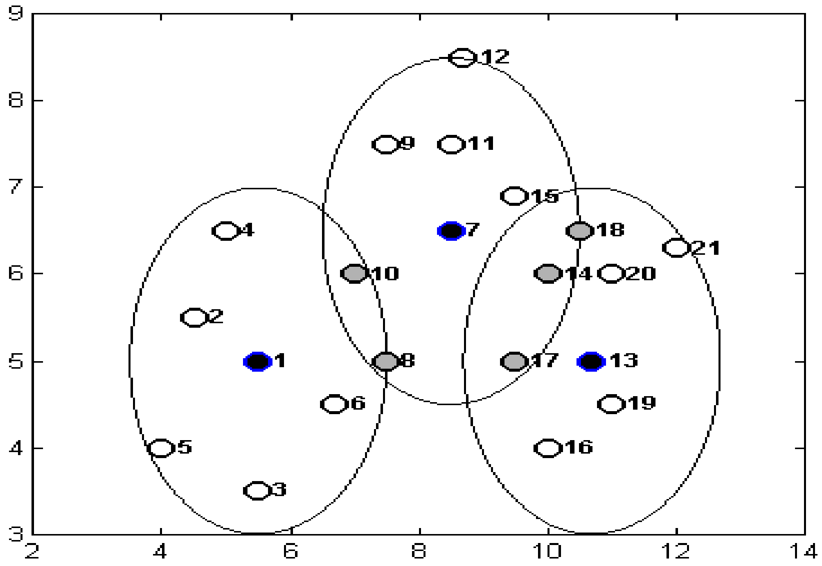


Figure 5.15(a): Stage 1 of CBRP in case IV

Stage 2: Source node 17 initiates the path construction process for destination 15 and sends the route request packet to its cluster head 13, as shown in Fig. 5.15(b).

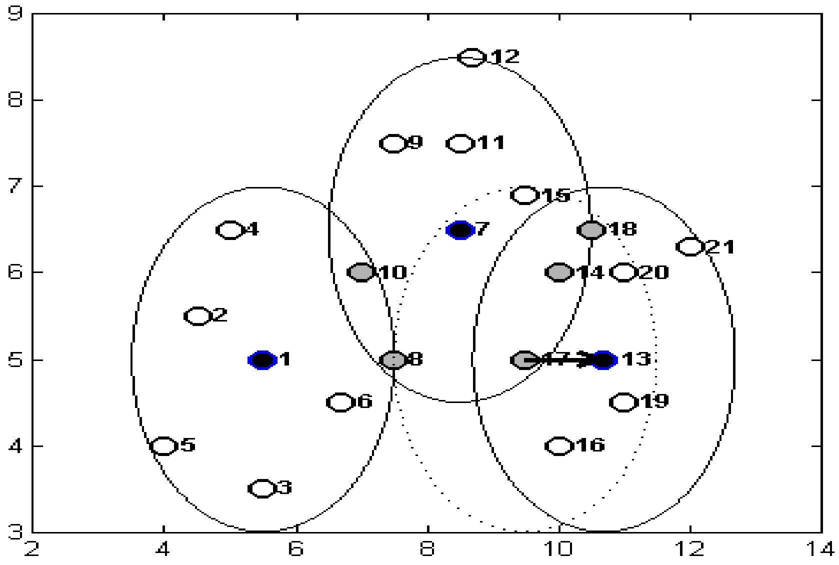


Figure 5.15(b): Stage 2 of CBRP in case IV

Stage 3: Cluster head 13 determines the absence of destination 15 in its cluster and forward the route request to gateway node 18, as shown in Fig. 5.15(c).

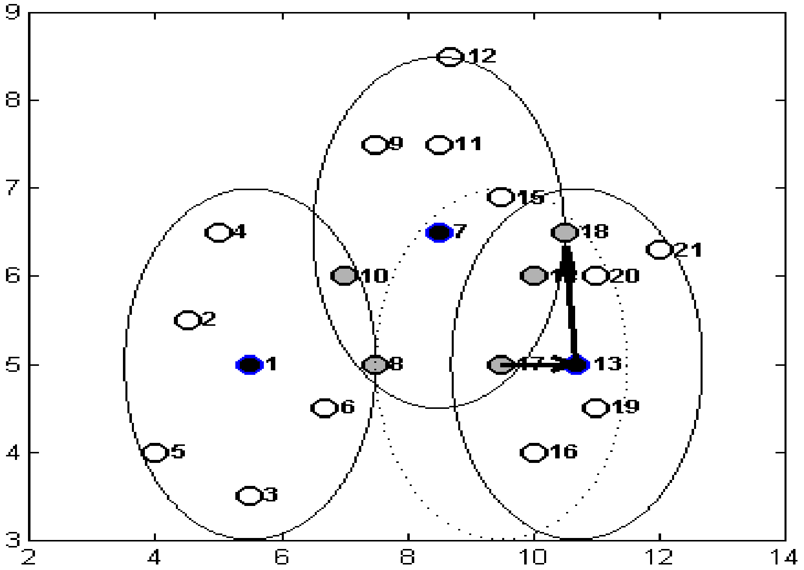


Figure 5.15(c): Stage 3 of CBRP in case IV

Stage 4: Gateway node 18 forwards the route request packet to nearby cluster head 7, as shown in Fig. 5.15(d).

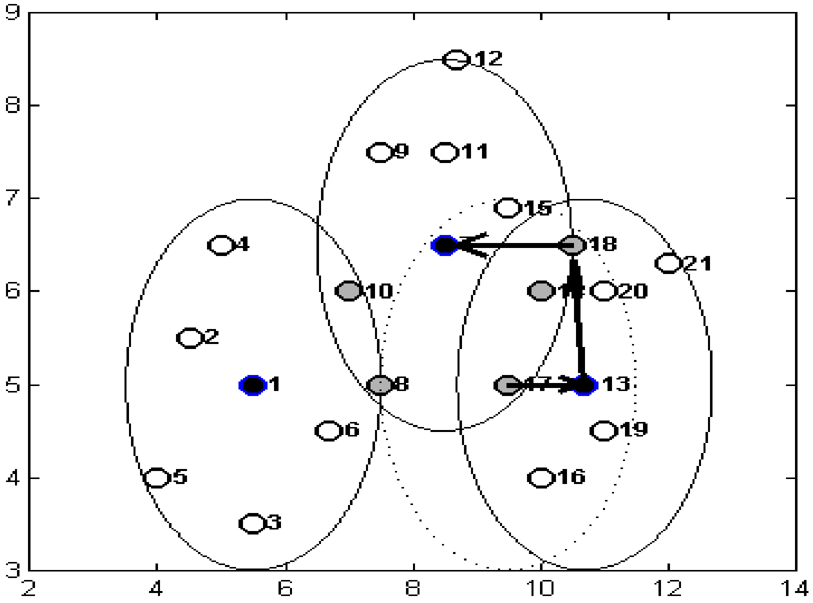


Figure 5.15(d): Stage 4 of CBRP in case IV

Stage 5: Cluster head determines the presence of destination 15 and forwards the route request to node 15, as shown in Fig. 5.15(e).

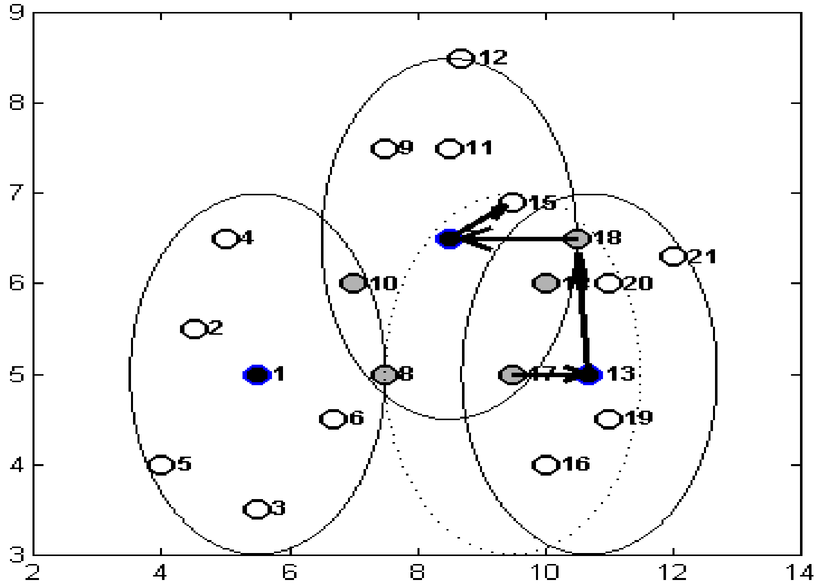


Figure 5.15(e): Stage 5 of CBRP in case IV

Stage 6: Destination node 15 sends back the acknowledgment packet to its cluster head 7, as shown in Fig. 5.15(f).

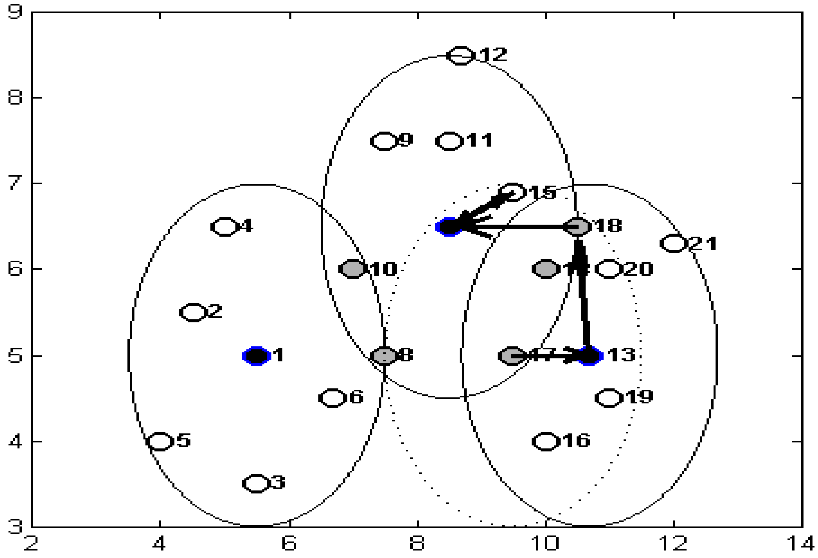


Figure 5.15(f): Stage 6 of CBRP in case IV

Stage 7: Acknowledgment packet travels on reverse path back to the source node 13 and path get established, as shown in Fig. 5.15(g).

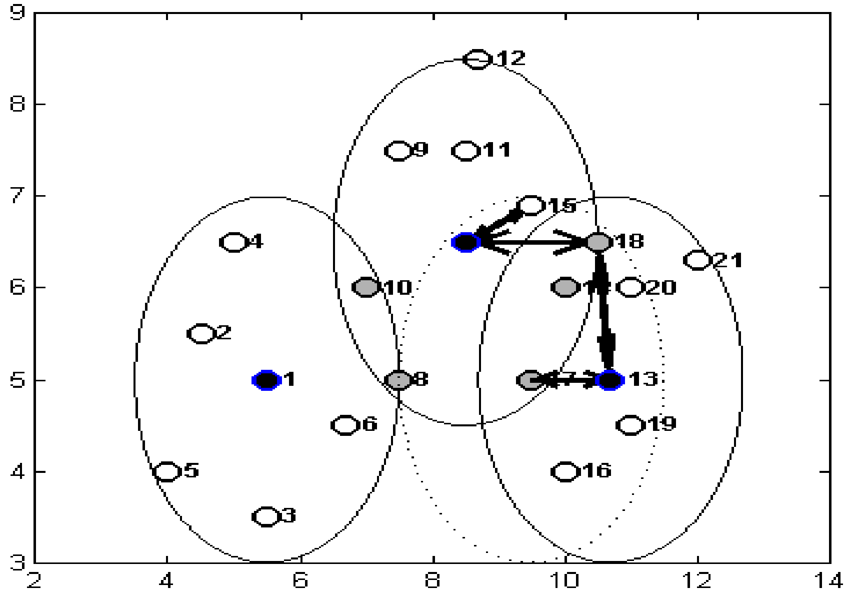


Figure 5.15(g): Stage 7 of CBRP in case IV

Figure 5.16 (a-c) show the simulation stages of path construction in LWECM. for case IV, the source node is 13 and destination node 15.

Stage 1: 3 well-formed clusters with three cluster head 1,7,13. Nodes in grey color are gateway nodes, as shown in Fig. 5.16(a).

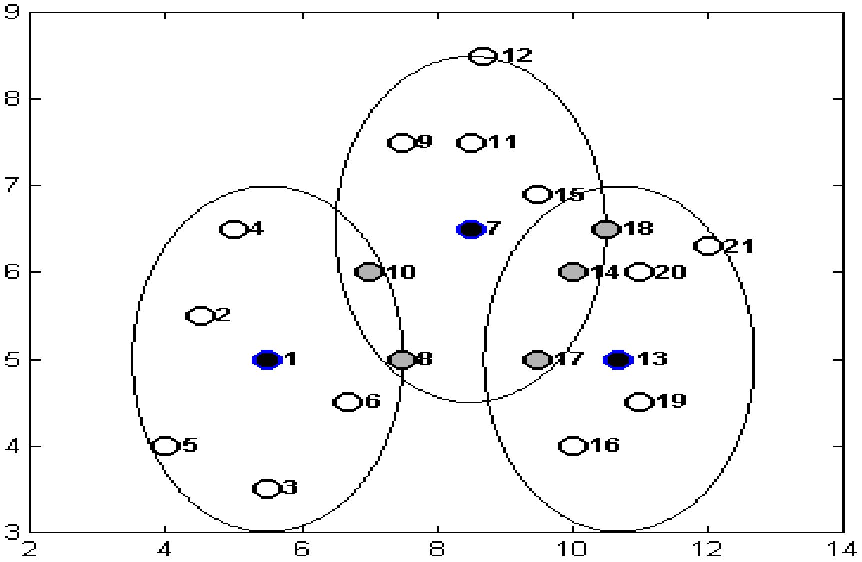


Figure 5.16(a): Stage 1 of LWECM in case IV

Stage 2: Source node 17 determines the presence of destination node 15 in its radio range and sends the route request packet to 15 directly, as shown in Fig. 5.16(b).

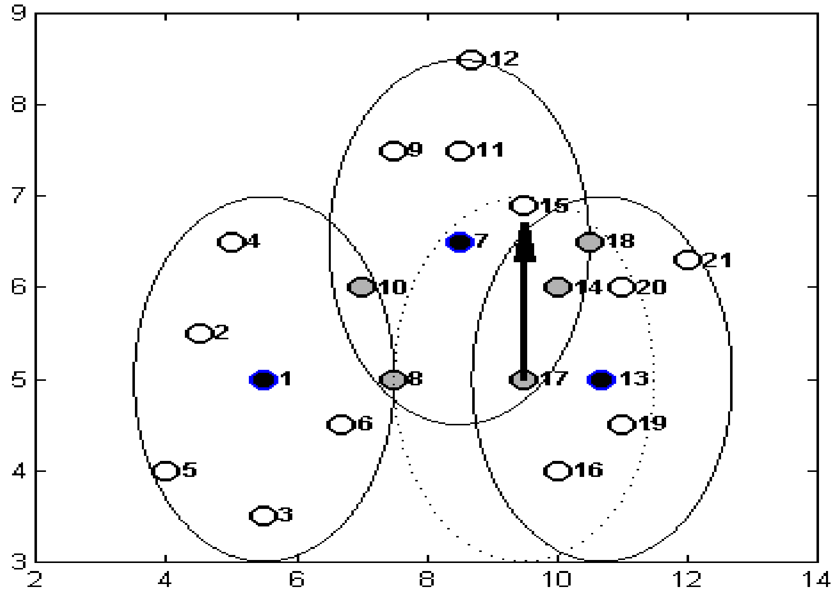


Figure 5.16(b): Stage 2 of LWECM in case IV

Stage 3: Destination node 15 sends acknowledgment packet back to source 17 and path get established, as shown in Fig. 5.16(c).

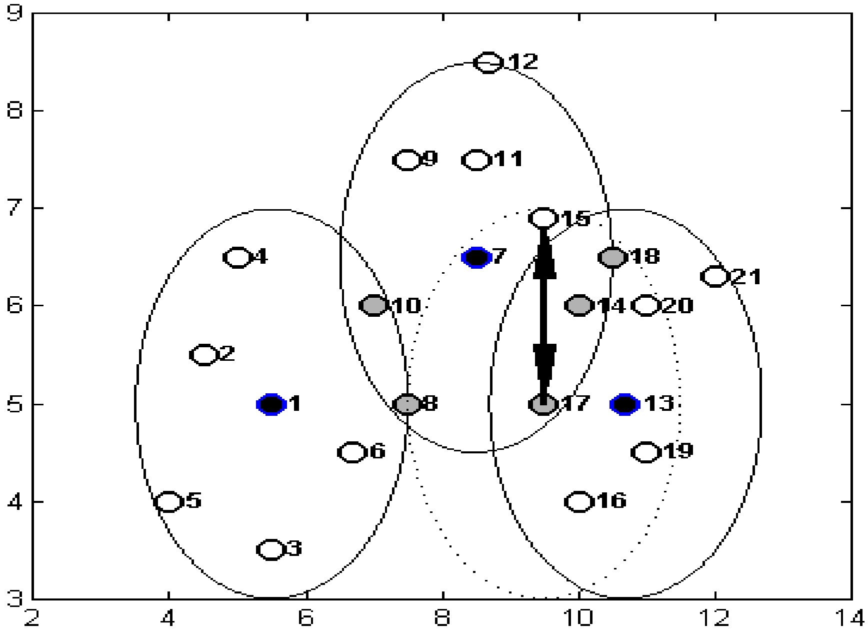


Figure5.16(c): Stage 3 of LWECM in case IV

Simulation findings for path construction of both protocols, for all 4 cases have been tabulated and compared in Table 5.1.

Table 5.1: Comparison of simulation findings of path construction of CBRP and LWECM

	Consideration	**CBRP(Existing)**	**LWECM (Proposed)**
CASE I (Source 5 and destination 3)	Path	5→1→3	5→3
	Number of CH involved	1	0
	Number of Gateway involved	0	0
	Hop Count	2	1
CASE II (Source 3 and destination 4)	Path	3→1→4	3→6→4
	Number of CH involved	1	0
	Number of Gateway involved	0	0
	Hop Count	2	2
CASE III (Source 5 and destination 4)	Path	5→1→10→7→14→13→20	5→1→10→7→14→20
	Number of CH involved	3	2
	Number of Gateway involved	2	2
	Hop Count	6	5
CASE VI (Source 17 and destination 15)	Path	17→13→18→7→15	17→15
	Number of CH involved	2	0
	Number of Gateway involved	1	0
	Hop Count	4	1

It may be observed from the Table 5.1 that proposed approach LWECM is economic and light weighted in comparison to exiting approach CBRP as discussed below.

Case I is intra cluster communication, where source and destination are in radio range of each other CBPR yields the path **5→1→3** from source 5 to destination 3, which involves cluster head 1 unnecessarily with hop count 2. Whereas in LWECM, Source node 5 directly establishes a bidirectional link with destination 3 without bothering its cluster head 1 as destination is present in node 5’s radio range.

Case II is intra cluster communication where source and destination are not in radio range of each other CBPR yields the path **3→1→4** from source 3 to destination 4 by involving the cluster head 1 with hop count 2. On the other hand in LWECM, source 3 sends request to its cluster head 1, thereafter cluster head appoints an intermediate node 6 to spare itself and rest of the communication takes pace by the path **3→6→4.**

Case III is inter cluster communication where source and destination are not in the radio range of each other CBPR yields the path **5→1→10→7→14→13→20** from source 5 to destination 20 which involves three cluster heads 1,7,13 with hop count 6. Whereas LWECM yields the path **5→1→10→7→14→20** with 2 cluster head 1,7 and hop count 5.

The case IV (for intra cluster communication where source and destination are in the radio range of each other) is more interesting and facts are revealing. In case 4, CBRP yields the path **17→13→18→7→15** with two cluster heads 13,7 and hop count 4 in spite of source 17 and 15 are in the radio range of each other due the fact that, every communication has to pass through respective cluster heads. On the other hand in LWECM source 17 establishes a bidirectional link with destination 15 i.e. **17→15** without involvement of either cluster head with 1 hop count only. The cluster-head involvement while path construction using CBRP and LWECM for all four cases has been plotted in Fig 5.17.

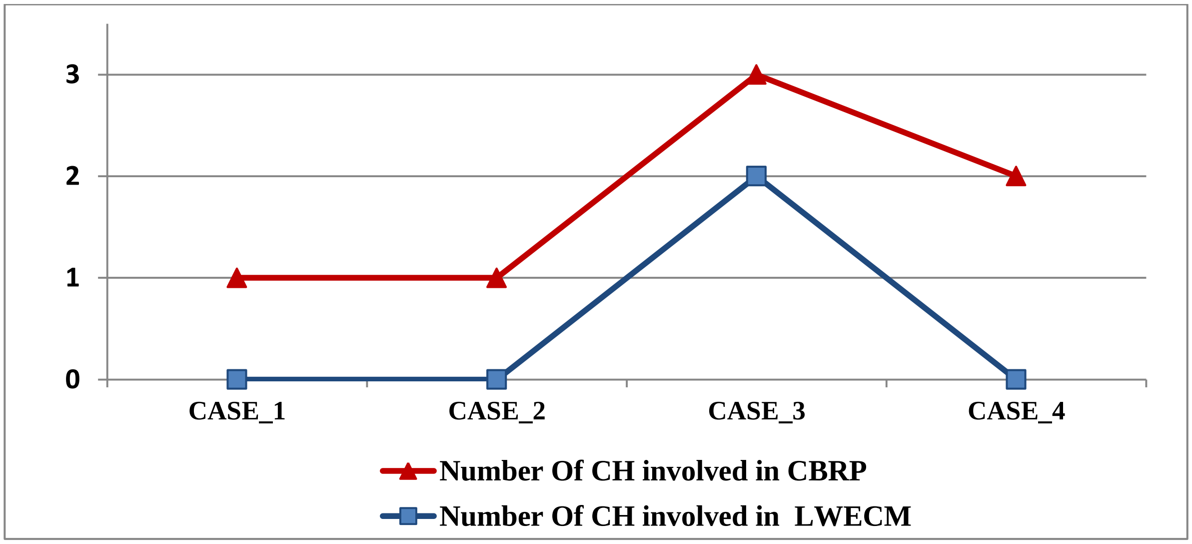


Figure 5.17: Cluster Head involvement in CBRP and LWECM

It may be observed from Fig 5.17 that involvement of cluster-heads using LWECM is less in comparison to CBRP.

Similarly, the number of gateways involvement while path construction by using CBRP and LWECM for all four cases has been plotted in Fig 5.18.

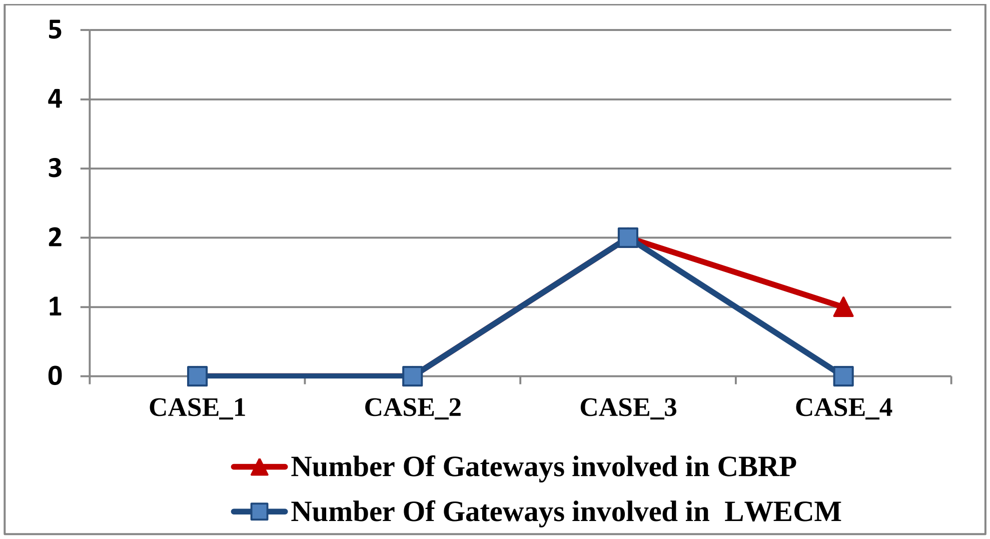


Figure 5.18: Number of Gateway Involved in CBRP and LWECM

It may be observed from Fig 5.18 that involvement of gateways using LWECM is less in comparison to CBRP.

Further, the hop count of constructed path by using CBRP and LWECM for all four cases has been plotted in Fig 5.19.

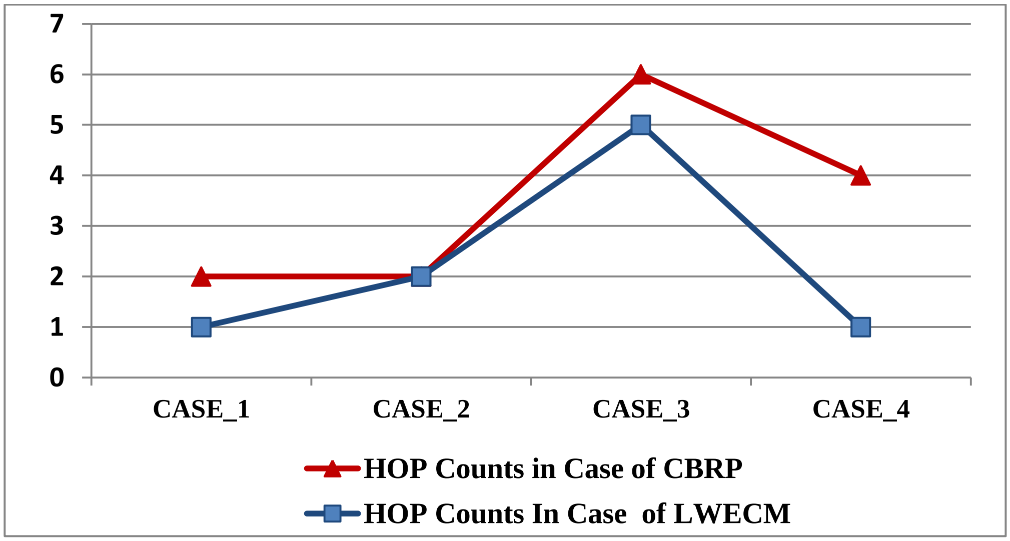


Figure 5.19: Hop counts of constructed path in CBRP and LWECM

It may be observed from Fig 5.19 that the hope count in path yielded by LWECM in three out of four cases is less as compared to CBRP resulting in saving of the network resources.

5.5 SUMMARY

In this work "A Light Weight Efficient Cluster based routing Model for Mobile Ad-hoc Networks (LWECM)" has been proposed. This model is light weighted in the sense that a transmitting node asks the service of cluster-head only when the designation node is situated out of its radio range. The life of a cluster-head is prolonged by sparing it from the unnecessary involvement in communication. Initial setup of cluster structure needs very light calculation at every node. A novel low cost cluster maintenance procedure has been proposed in this research. In case of movement of cluster-head from its cluster, the orphan members re-affiliate themselves from nearby cluster without disturbing the existing cluster structure. The concept of refugee node is introduced to reduce the complexity for maintenance procedure and controls the number of clusters in the network. LWECM ensures the fair load distribution among the cluster-heads by calculating the burden factor BF which is the function of node's degree and remaining power. An affiliation seeking node always affiliates itself with the nearby cluster-head having lowest BF value thereby reducing the burden on affiliating node. The route discovery process of standard CBRP and LWECM has been simulated and findings reveal that LWECM required lighter control overhead and cluster-heads are spared from unnecessary exercise.

In next chapter "Marking based Load balancing Weighted Clustering framework for Mobile ADHOC Network" is presented.

CHAPTER 6

MARKING BASED LOAD BALANCING WEIGHTED CLUSTERING FRAMEWORK FOR MOBILE ADHOC NETWORK

6.1 INTRODUCTION

In mobile ad-hoc network scalability is achieved by partitioning the network into geographical groups known as clusters [82][83]. In other words clustering is grouping of neighboring nodes (two nodes are said to be neighbor if they are in transmission range of each other). In cluster based routing algorithms, mainly three types of nodes are defined: cluster head, doorway and ordinary node. [84][85][86].

During the setting up of cluster structure a cluster head is elected among the network nodes by certain cluster head election approach. Actually a cluster-head election process is to partition the network. After winning the election, cluster head and its neighbors make the cluster and eventually entire network gets partitioned into the clusters. Cluster-head acts as local controller and is responsible for medium access control in its cluster. Cluster-head keeps the status of its members and facilitate them for intra cluster and inter cluster communication. Doorway nodes are the members of cluster which can hear (in transmission range) the neighboring cluster-head or its member. Doorway nodes act as interface between neighboring clusters as they are responsible for forwarding of outgoing traffic and receiving of incoming traffic. Ordinary nodes are cluster members which are neither cluster-head nor doorway nodes. Whenever a cluster member wants to communicate with other node, it transmits the packet to its cluster-head. Cluster-head then looks if the destination is in its member list. If yes, it will forward the packet directly to the destination and if destination is outside the cluster then cluster-head forwards the packet to its doorway nodes. Subsequently doorway nodes forward the packet to nearby cluster this process continues until the destination node is found. Thereafter acknowledgment packet backtrack the reverse path and path gets established. In the network, union of cluster-heads and doorway nodes make the backbone for routing the data called dominating set. In the well-formed clustered network, every node is either a member of dominating set or the neighbor of at least one member of dominating set. Cluster structure is depicted in Fig. 6.1.

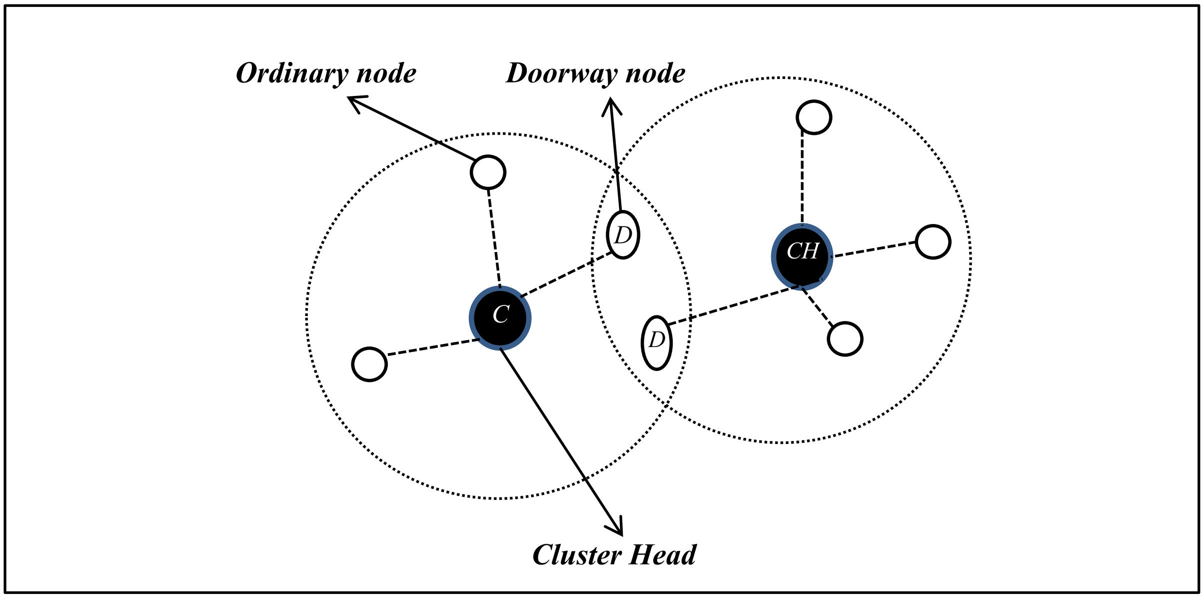


Figure 6.1: Standard Cluster structure

It can be observed from the above discussion that clustering based routing approaches have many recognizable advantages; one of them is scaling of network that is increasing the number of mobile nodes. Clustering ensures this basic functionality with satisfactory performance [87], [88]. Other one is reusability of spatial resources like frequency or codes in non-overlapping clusters [89]. Another advantage is that the routing information exchange is limited to backbone (cluster-heads and doorway nodes) which reduce traffic over the network [90]. Finally nodes in network needs to be aware of their neighbors only (local topology) therefore large and dynamic network seem smaller and steady in the view of every node [91][92]. Along with these advantages clustering have their own consequences like extra effort in creating and maintaining the cluster [93][94][95]. Efficiency of any clustering scheme is judged by the amount of effort required to setup the cluster structure [96] [97] [98]. In some clustering schemes occurrence of a local event like expiration or transit of cluster-head may require the re-clustering of entire network which leads to higher maintenance cost. Many researchers have put the effort to improve the clustering based routing by minimizing the control overhead and ensuring the stable cluster structure [99][100][101][102].

6.2 GAPS AND PROPOSED SOLUTION

In this work standard collective-weight-based clustering algorithm: weighted clustering algorithm (WCA) has been studied which reveals several intrinsic shortcomings. In this section these shortcomings and their corresponding proposed remedies are discussed.

Foremost challenge in the collective weight based clustering algorithms is large computation overhead. In cluster-head selection process every node has to compute its weight and exchange the control messages in its locality for measuring the distance, speed, degree and for claiming the measured weight, therefore it induces the communication overhead. Also, in clustering process every node loses scarce battery power in significant amount consequently reducing the network life time. In the proposed work "*Marking based Load balancing Weighted Clustering framework for Mobile ADHOC Network*" (MLWCM) mark the incompetent nodes whose possibility of winning the election is negligible since they don't have power level up to predefined threshold and spare them to participate in election process. In this manner algorithm blocks the unnecessary injection of traffic and saves the battery power thus increasing the lifetime of network.

Other weakness of existing weighted clustering algorithm is unfair load distribution on cluster-heads. Because WCA takes the generalized ideal number of node δ that a cluster-head can handle and on the basis of δ every node computes the degree deference by the formula $\Delta v = | d_v - \delta |$. However in heterogeneous ad-hoc network environment nodes are distributed randomly, so degree of node remains different for every node, therefore, calculation of degree-difference Δv with common upper bound δ may leave many cluster-heads overburdened.

Moreover, available battery power (aliveness) of every node in the network may not be the same, therefore, capacity to manage the number of nodes varies from node to node. In the proposed work, instead of generalized upper bound, specified ***h*_**values are taken from a bucket called ***h***_table. ***h*_**value is defined as the ideal number of nodes that a cluster-head can handle w.r.t. to its available battery power. The notion is that a node with more battery power can serve more nodes as cluster-head efficiently for an acceptable time. ***h*_**table contains the set of ***h*_**values with respect to range of power level such that $H_i > H_{i+1}$ instance of ***h***_table is shown in Table 6.1. There is a threshold of power level, below which ***h*_**value is set to zero, indicates that node (having battery power less than threshold) is not suitable for being a cluster-head and spared from the election process.

Furthermore, the values in the ***h***_table can be regulated according to network scenario. For example, if the battery power is more crucial, then the tight bound on power-ranges may be kept, otherwise it may be made flexible. In this work 'Handling factor' H_v is calculated by $H_v = \boldsymbol{h} - d(v)$ instead of degree difference. Here the value of H_v is not considered *absolute* purposely. The positive value of H_v reflects that node having the number of neighbors under its capacity of being cluster-head and negative value shows the number of neighbors is over the capacity.

Table 6.1: Instance of *h*_table

Range of power level	Ideal No node a cluster_head can handel(*h*_value)
100 – 80 %	h_1
79-65%	h_2
64-40%	h_3
39-20%	h_4
<20%	**-**

Next shortcoming of existing weighted clustering algorithm is identified as the estimation of energy consumption of a node by the serving time of that node as cluster-head. A cluster-head drains its energy proportionately to the amount of traffic that passes from it. In ad-hoc network as the traffic flows in random fashion, therefore, it can be possible that a newly elected cluster-head may pass more traffic than an older one. Therefore the total of serving time of being a cluster-head cannot be a guarantee of energy consumption. Considering the above reason, in this work precise outstanding battery power Pv is taken for calculating the competence value (weight). In the existing approach, a factor D_v that is sum of distance to all neighbors is taken for weight calculation. But measuring the distance through receiving signal power is not the correct assessment of distance because a closer node transmitting low power signal may appear as far and vice versa. Moreover contribution of sum of distances in weight calculation is not much significant due to the fact that in the broadcast system a node merely needs to be in the transmitting range for communication. Another issue with the existing weighted clustering algorithm is that it allows only one-hop cluster that is a member node can only be connected with its cluster-head through a direct link. This restriction leads to high maintenance cost. Because a small change in topology or the movement of cluster-head may trigger re-clustering the entire network. This phenomenon is called the ripple effect of re-clustering.

The limitation of cluster size up to the single-hop creates larger number of clusters than required, consequently hop count over the path between the terminal will increase. To minimize the maintenance cost, the work done in *3-Hop Between Adjacent Cluster-head (*3hBAC) [25] is taken along with the use of and using a new status of node called inmate node. Whenever a node is unable to get membership of any cluster i.e. not in the transmission range of any cluster-head, but able to hear some member nodes of nearby clusters then it can affiliates itself as *inmate* node through any of these cluster members as shown in Fig. 6.2 (a,b). Inmate node is similar to guest node in 3hBAC [25] but unlike in 3hBAC, in MLWCM inmate node affiliates itself with the member node whose cluster-head have largest handling factor, hence reduces the possibility for a cluster-head to get overburdened. Concept of inmate node overcomes the ripple effect of re-clustering due to the movement or expiration of cluster-head. If a member node is unable to hear its cluster-head then it affiliate itself with nearby cluster as member node or inmate node silently, rather to make a separate cluster and trigger the re-clustering in entire network. Furthermore the concept of inmate affiliation allows the merging of trivial clusters (cluster having only one node, resultant of initial clustering) with primary clusters consequently minimizing the number of clusters in the network. To ensure the separation of two clusters, if two cluster-heads arrive in each other's range then one cluster-head with lower weight relinquishes its cluster-head status. This separation keeps the intra cluster changes local.

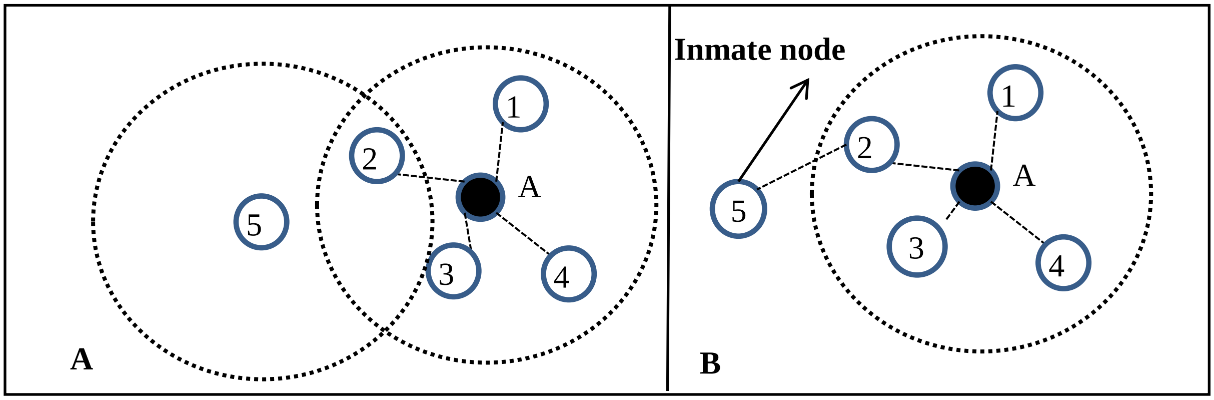


Figure 6.2(a,b): Affiliation as inmate node

6.3 THE PROCESS

Based on the gaps identified in preceding section, in this section proposed clustering framework "Marking based Load balancing Weighted Clustering Framework for Mobile ADHOC Network" MLWCM is being presented

6.3.1 Basis and assumption for proposed work

Clustering algorithm performs in two phases first is initial clustering and second is merging of trivial clusters (cluster containing only one node) with nearby clusters. During the formation of cluster structure, network topology is assumed as static. Any mobile node can be a member of exactly one cluster.

After initial clustering, cluster-head election procedure is invoked only when dominating set is unable to cover all the nodes in the network. Initially single hop clusters are formed, afterword cluster can extend up to two hops (result of inmate affiliation or merging of trivial clusters).

All nodes in the network possess the battery of same capacity (mAh) and nodes are able to sense their outstanding power level. Nodes having the power below a certain threshold are marked as weak and spared from cluster-head election process. They can only be the member of a cluster.

Every node in the network is able to measure its speed. All nodes in network are aware about their suitable ***h*** value contained in ***h*_**table for determining the handling factor H_v. To elect the healthy cluster-head, MLWCM considers following three parameters for calculation of competence value CF.

Outstanding power P_v (%) **:** Outstanding power P_v is the assessment of survival of a node and can be determined by equation 6.1. P_v contributes to upsurge the life time of clustered network. Cluster-head with sufficient outstanding power increases the cluster existence therefore reduces the maintenance cost due to the expiration of cluster-head.

$$P_v(\%)=(\text{present battery level}(v)/\text{total capacity}(v)*100 \qquad (6.1)$$

Handling factor H_v: Handling factor H_v is the assessment of how many member nodes a cluster-head can handle efficiently, considering its outstanding power level. Hv is the difference of the degree of a node $d(v)$ (algorithm for creating neighbor set is shown in Fig. 6.4) and its corresponding ***h*** value calculated by equation 6.2. H_v controls the size of cluster for fair load distribution. Positive value of H_v indicates the number of members under the capacity and conversely negative value is alert of overcapacity.

$$H_v = h - d(v) \tag{6.2}$$

Node average speed Sv : Node speed is assessment of stability of a node. Average speed of a node can be determined by the equation 6.3. As node with high mobility is inappropriate to play cluster-head role as it leads to frequent cluster breakage and consequently higher maintenance cost.

$$Sv = \frac{1}{t} \sum_{t_n=1}^{t} \sqrt{(x_{t_n} - x_{t_{n-1}})^2 - (y_{t_n} - y_{t_{n-1}})^2} \tag{6.3}$$

Where t is the observe time and (x_{t_n} , y_{t_n}) and ($x_{t_{n-1}}$, $y_{t_{n-1}}$) are the coordinates of node V at two consecutive time moments.

Clusterization and cluster-head election mechanism of network is discussed in next subsection.

6.3.2 Clusterization of network

System activation starts with the initialization of weighing factors α**,** β,γ and the ***h***_table to meet the system requirements. For cluster formation all the nodes in the network are assumed to have knowledge of common ***h***_table to determine their suitable ***h*** value corresponding to their outstanding battery level. Cluster formation is done in two phases first, cluster-head election to form 1-hop cluster and second merging of trivial clusters with nearby clusters formed in first phase. Algorithm for phase I is shown in Fig. 6.3

```
ALGORITHM FOR PHASE I OF CLUSTRIZATION
Step 1: Determine the outstanding power $P_v$
Step 2: Select the h value from h_table corresponding to outstanding power $P_v$
If h value > 0
{
 Step 3: Node v creates its neighbor set (nodes in transmission rage), and determine its degree d(v) the number of member in the neighbor set

 Step 4: Calculate the Handling factor
 $H_v = h - d(v)$ by equation 2

 Step 5: Calculate the average speed
 $S_v$ by equation 3 and normalize it
 $S_v = 1/S_v$

 Step 6: calculate the competence value $CF_v$
     $CF_v = \alpha Hv + \beta Sv + \gamma Pv$
Coefficient $\propto$ $\beta$ and $\gamma$ are weight factors according to network parameter such that
 $\alpha + \beta + \gamma = 1$
}
 Step 7: select the node with largest competence factor $CF_v$ as cluster head

 Step 8: All the neighbors of cluster-head selected in step 7 join that cluster and not allow further to participate in election process.

 Step 9: Cluster-head allotted in step 7 and 8 then update its handling factor $H_v$ by following equation
 $H_v = H_v + (d(v) - \text{number of member})$ ;

 Step 10 : Repeat step 1 to 8 until all node assign to a cluster.
```

Figure 6.3: Algorithm for phase I of clustrization

```
ALGORITHM CREATE NEIGHBOR_SET
    Let V is set of nodes in network
    $(v) neighbor set of node n
    r is the transmission range of node n
    Create_Neighbor_set(V)
    {
      For every node $v_i$ in V
        If ( distance (n ,$v_i$) < = r)
        Include $v_i$ in $(n)
    }
```

Figure 6.4 : Algorithm creating neighbor set

In second phase, cluster-heads determine if they have no member node, then they affiliate themselves with nearby cluster by broadcasting the REQ_AFFI message (packet format for REQ_AFFI as shown in Fig. 6.6). On receiving REQ_AFFI message all neighboring nodes respond with REPL_AFFI message containing their cluster-head Id CH_ID along with present handling factor $\mathbf{H}_v$ of its associated cluster-head (packet format for REPL_AFFI is shown in Fig. 6.7). Thereafter requesting cluster-head affiliates itself with the member of cluster-head with largest handling factor by sending the INMATE message (packet format for INMATE is shown in Fig. 6.8) and changes its status to inmate. Steps for Inmate affiliation is given in Fig. 6.5.

ALGORITHM INMATE

*Inmate affiliation algorithm invoked by a node **v***

*Step1: Node **v** Broadcast REQ_AFFI message to all its neighbors.*

*Step2: On receiving the REQ_AFFI massage all neighbors respond with REPL_AFFI containing their CH_ID along with its handling factor **Hv.***

*Step3: Node **v** then processes all REPL_AFFI massage and select neighbor **n** which is associated to cluster-head with highest* $\mathbf{H}_v$.

*Step4: Node **v** then send the INMATE massage to neighbor **n** selected in step 3.*

*Step5: Neighbor **n** then forward INMATE message to its cluster-head **CH**.*

*Step6: Cluster-head (**CH**) then make the entry of node **v** in its cluster table and update the handling factor by **Hv** = **Hv** -1.*

Figure 6.5: Algorithm inmate affiliation

Node_ID	STATUS	TYPE
-	-	-

Figure 6.6: Packet format REQ_AFFI

Node_ID	CH_ID	H_v of CH
-	-	-

Figure 6.7: Packet format REPL_AFF

Node_ID	Member_ID
-	-

Figure 6.8: Packet format for INMATE message

Where **:** Node_ID: node's unique id, STATUS: CH, Gateway, member

H_v_Of_CH **:** Handling factor of CH

6.4 RESULTS AND DISCUSSION

Cluster formation by existing weighted clustering algorithm (WCA) and proposed clustering framework "Marking based Load balancing Weighted Clustering framework" for Mobile ADHOC Network (MLWCM) has been simulated using MATLAB. Simulation has been performed on 40 randomly deployed mobile nodes and they are labeled as 1 to 40. Area for simulation has been taken 200 X 200 m^2. For simplicity battery of each node is considered of same capacity (mAH) and outstanding battery power at every node varies from 1 to 100 %. For the system activation of the proposed work MLWCM, ***h*** values in ***h_table*** is initialized according to outstanding battery level as shown in Table 6.2. Cluster formation for both the protocols is simulated in identical network topology and nodes attributes. Simulation stages for both the protocols WCA and MLWCM are recorded and presented in Fig. 6.9(a-l) and 6.10(a-l) respectively.

Table 6.2: *h*_table

Range of power level	**Ideal No. of node a cluster_head can handel(*h_value*)**
100 – 80 %	10
79-65%	8
64-40%	6
39-20%	4
<20%	0

Simulation outcomes are tabulated and comprehended in Table 6.3, 6.4 and 6.5. The simulation outcome of MLWCM has also been compared with the outcome of WCA and comprehended in table 6.6.

Simulation results show that in case of existing WCA, selected cluster-head are not healthy in terms of available battery life which results in frequent cluster-head expiration and leads to higher maintenance cost.

As the WCA allows single hop cluster, many uncovered single nodes forms their own clusters which results in larger number of clusters than required. Consequently, hop count in the path, between the communicating terminals gets increased.

Stage 1: Initial topology of 40 nodes of an ad-hoc network, as shown in Fig. 6.9(a).

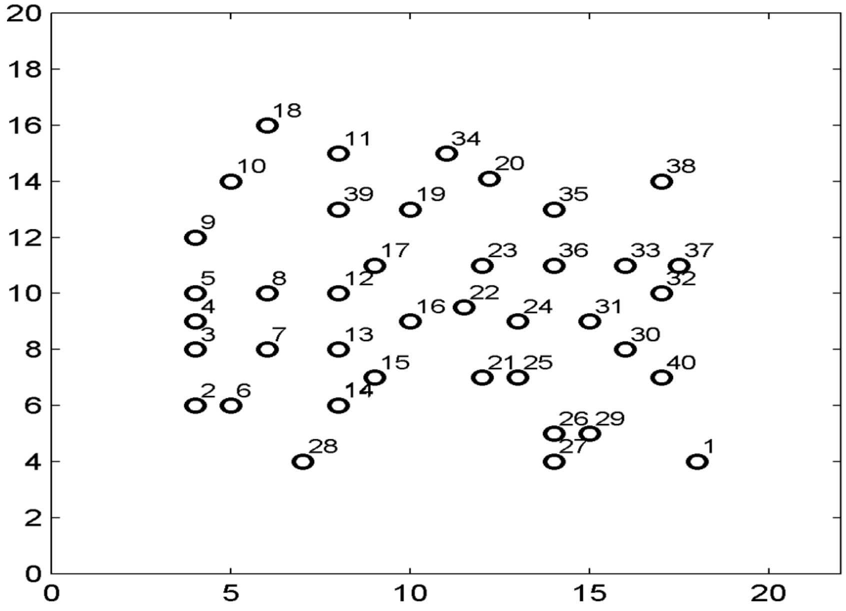


Figure 6.9(a) :Stage 1 of WCA

Stage 2: Node 39 elected as cluster-head then it formes its cluster, as shown in Fig. 6.9(b).

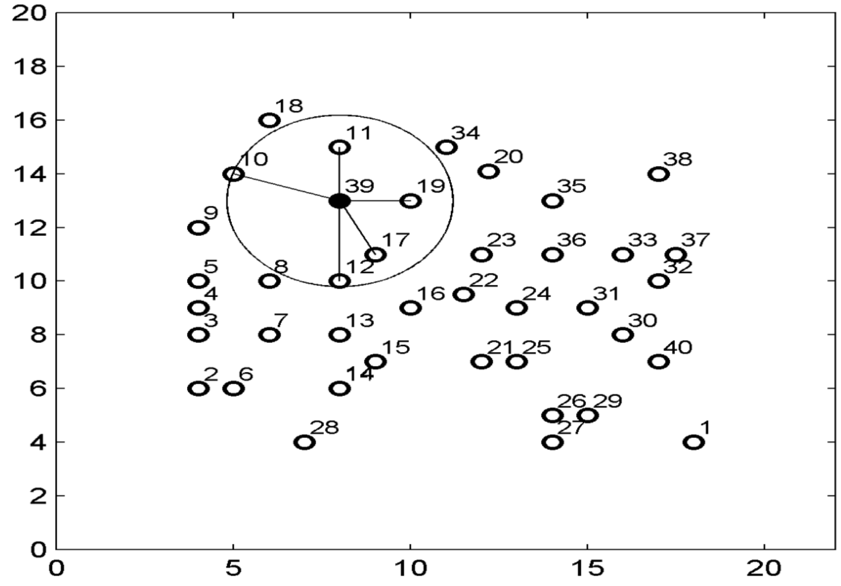


Figure 6.9(b): Stage 2 of WCA

Stage 3: Node 21 elected as cluster-head then, it formes its cluster, as shown in Fig. 6.9(c).

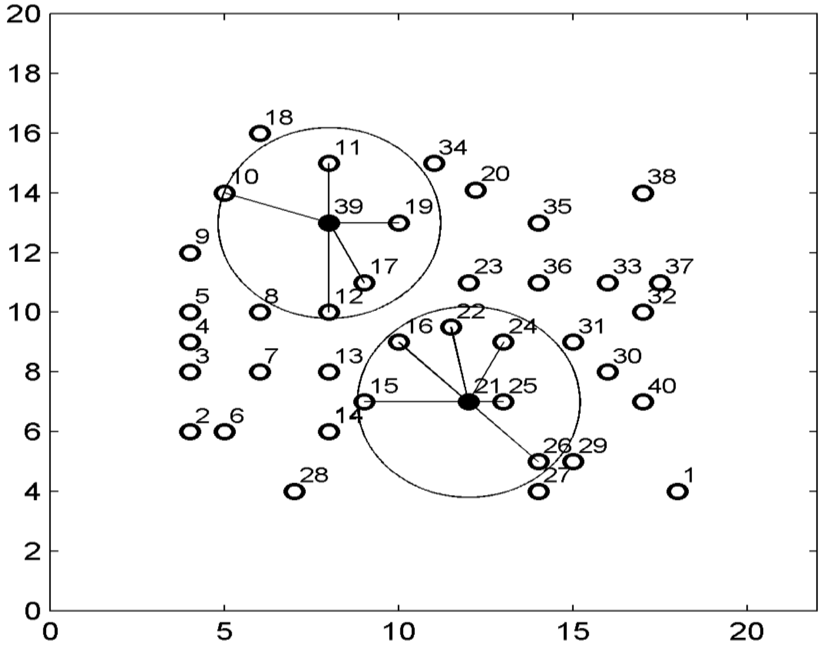


Figure 6.9(c): Stage 3 of WCA

Stage 4: Node 40 elected as cluster-head and formes its cluster, as shown in Fig. 6.9 (d).

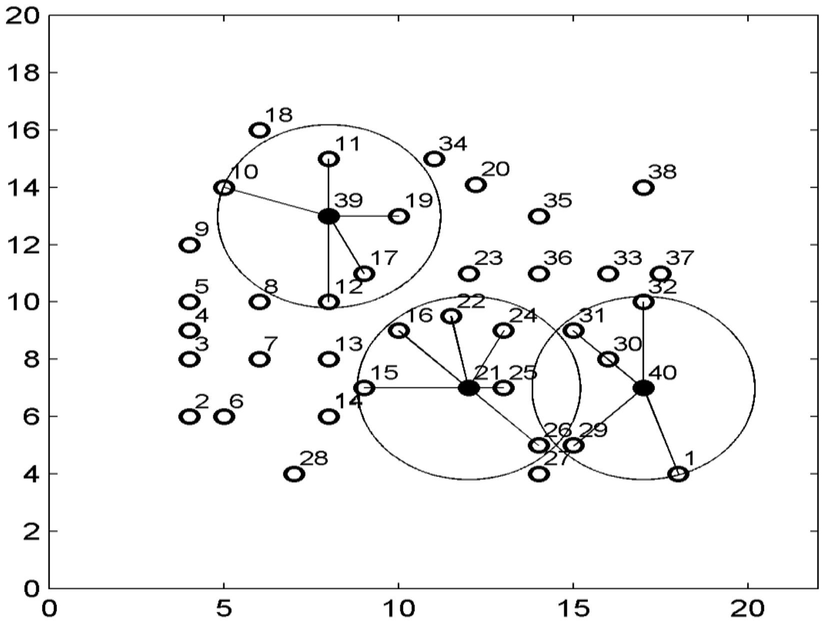


Figure 6.9(d) : Stage 4 of WCA

Stage 5: node 33 elected as cluster-head and formed its cluster, as shown in Fig. 6.9(e).

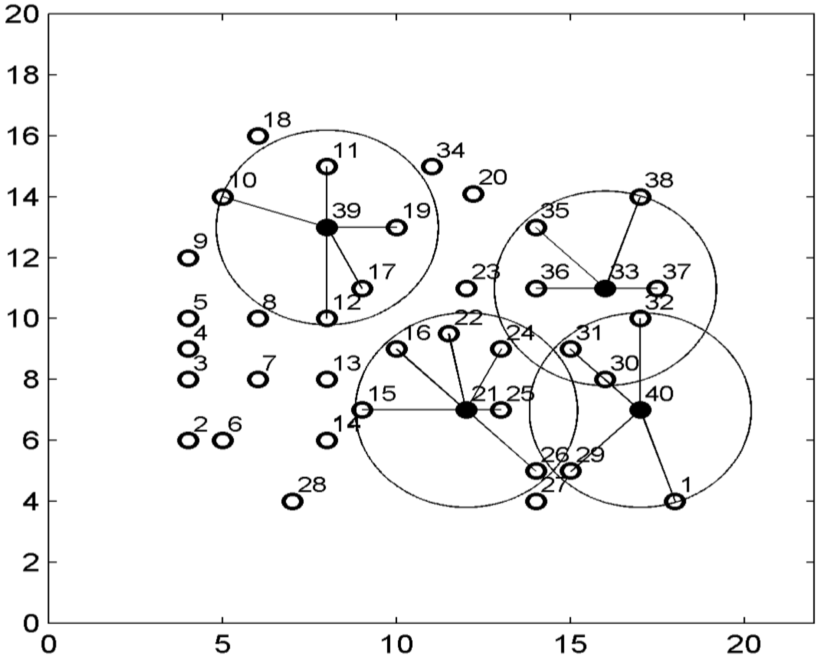


Figure 6.9(e): Stage 5 of WCA

Stage 6: Node 34 elected as cluster-head and formed its cluster, as shown in Fig. 6.9(f).

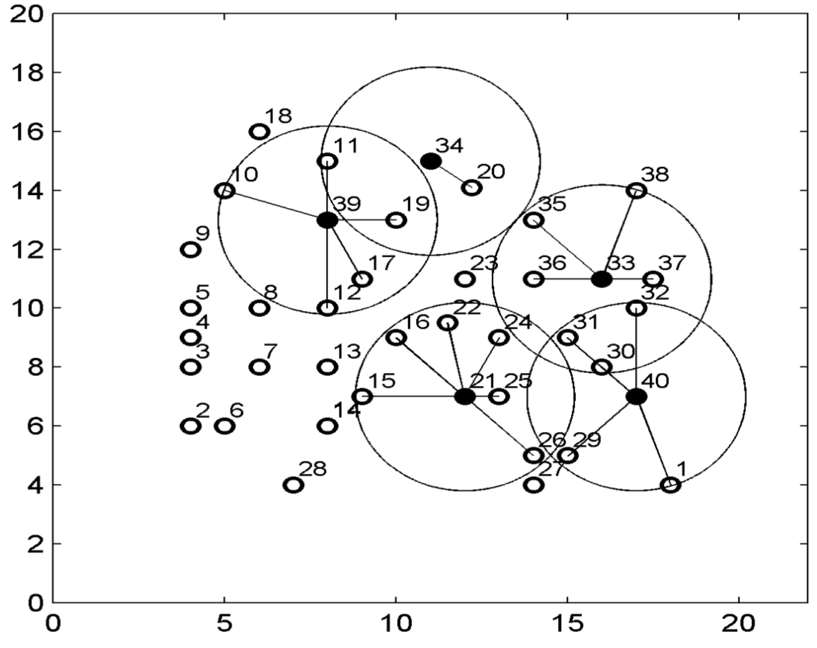


Figure 6.9(f): Stage 6 of WCA

Stage 7: Node 23 elected as cluster-head and formed its cluster, as shown in Fig. 6.9(g).

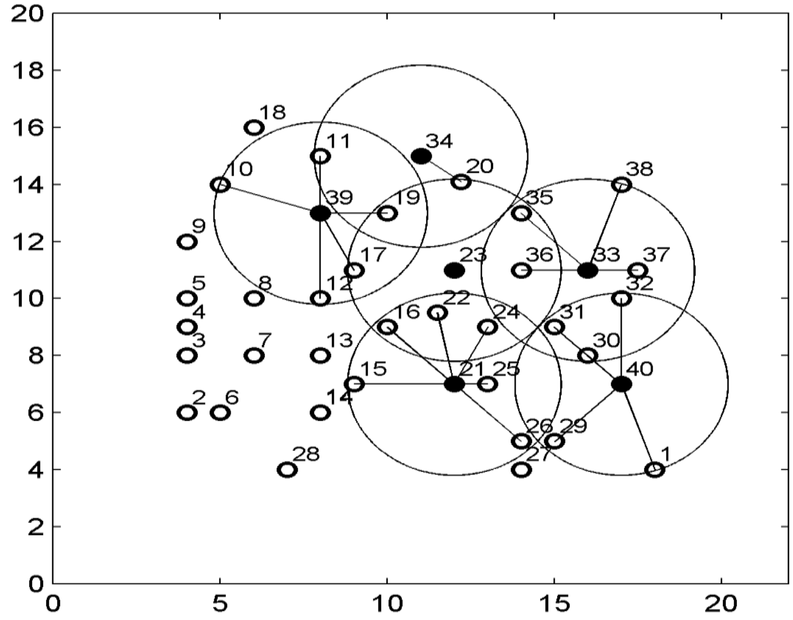


Figure 6.9(g): Stage 7 of WCA

Stage 8: Node 7 elected as cluster-head and formed its cluster, as shown in Fig. 6.9 (h).

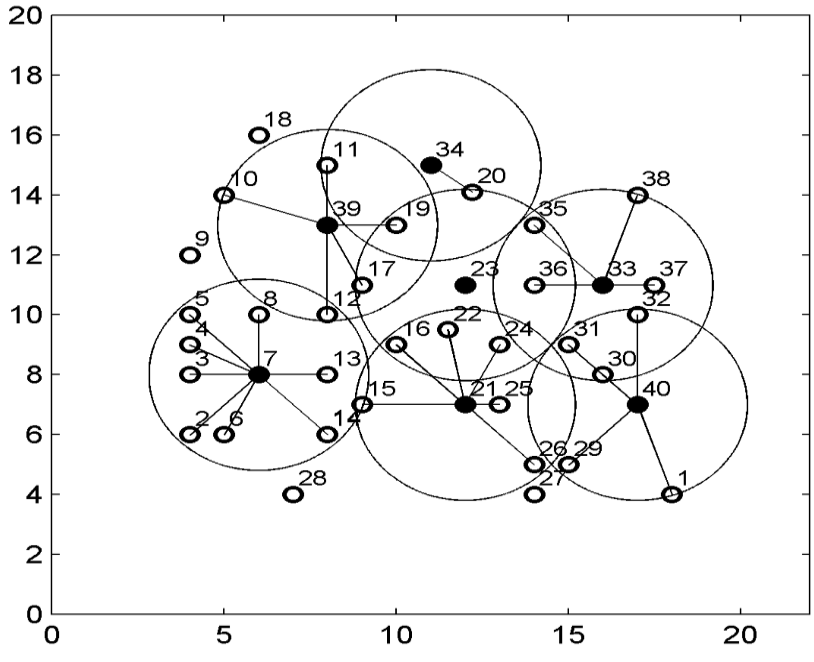


Figure 6.9(h): Stage 8 of WCA

Stage 9: Node 28 elected as cluster-head and formed its cluster, as shown in Fig. 6.9 (i).

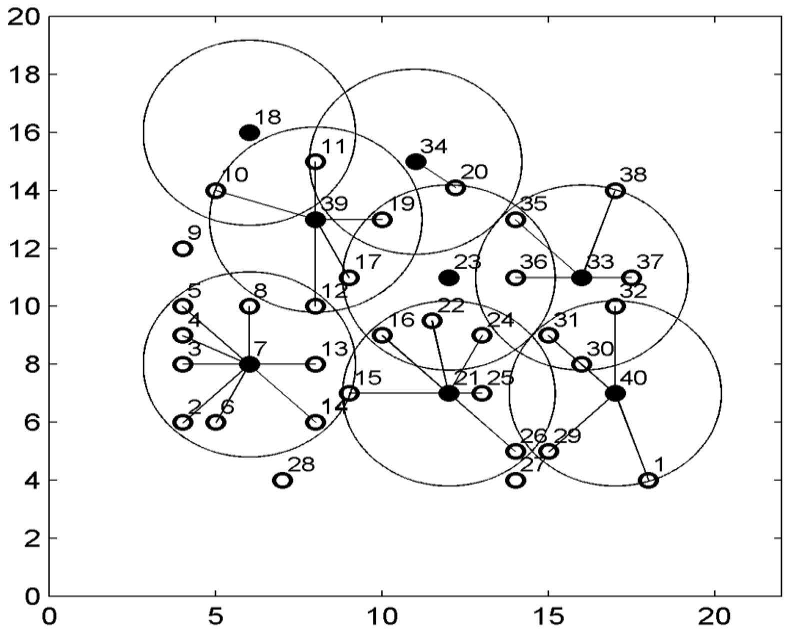


Figure 6.9(i): Stage 9 of WCA

Stage 10: Node 18 elected as cluster-head and formed its cluster, as shown in Fig. 6.9 (j).

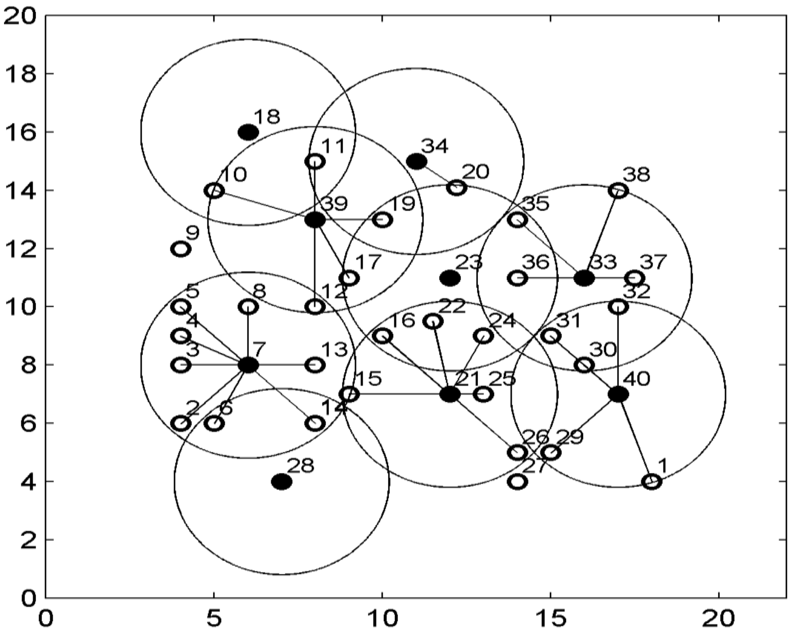


Figure 6.9(j): Stage 10 of WCA

Stage 11: Node 9 elected as cluster-head and formed its cluster, as shown in Fig. 6.9 (k).

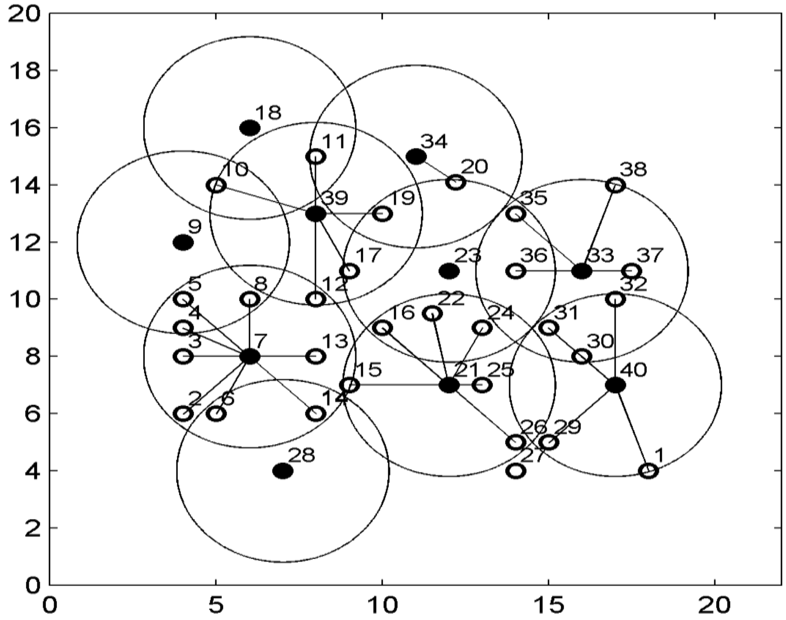


Figure 6.9(k): Stage 11 of WCA

Stage 12: Node 27 elected as cluster-head and formed its cluster, as shown in Fig. 6.9 (l).

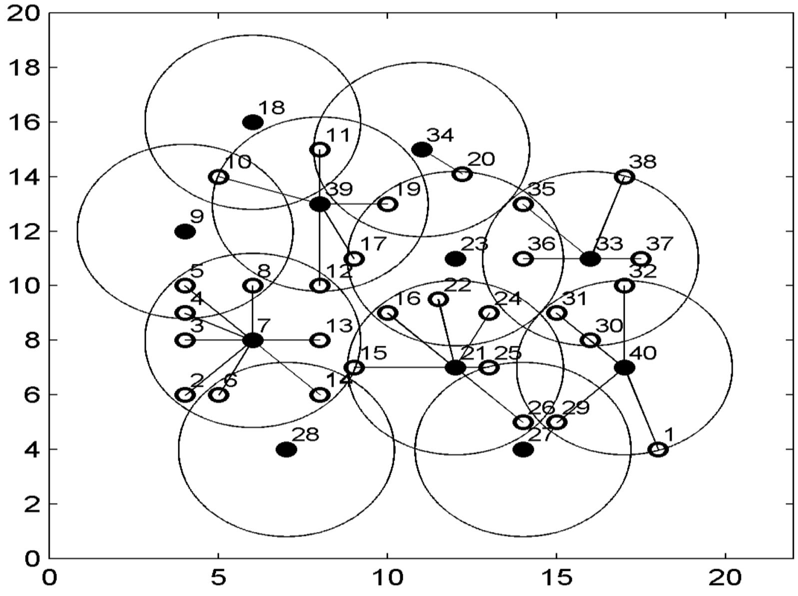


Figure 6.9(l): Stage 12 of WCA

Simulation findings of the cluster formation by using the WCA are shown in Table 6.3.

Table 6.3: Simulation results of cluster formation using WCA

S.N.	CH	Members	NO_OF Members	Degree Difference	Outstanding power	Mobility	Weight_Of CH
1	39	10,11 ,12,17 ,19	5	1	16	0.0100	1.4428
2	21	15,16,22,24,25,	6	0	64	0.0100	1.4472
3	40	1,29,30 ,31,32	5	1	12	0.0100	1.5263
4	33	35, 36,37,38	4	1	67	0.0100	1.8171
5	34	20	1	3	18	5.4500	2.9086
6	23	-	0	2	56	2.1200	3.0769
7	7	2 ,3, 4,5,6 ,8,13,	8	4	85	0.1000	3.3248
8	18	-	0	4	63	3.7100	3.5602
9	28	-	0	4	10	7.8300	3.6554
10	9	-	0	2	19	4.0000	5.8064
11	27	-	0	3	50	5.6100	22.8406

Also by taking the global ideal upper bound **δ**, the numbers of members in the clusters are unfairly distributed which leave some cluster-head overloaded and some clusters are under loaded as shown in Table 6.3. For example cluster-head 39 and 40 have 5 members with only 16% and 12% battery life respectively whereas cluster-head 33 have 4 members with 67% battery life.

More over WCA imposes huge calculation overhead at every node, even if the possibility of node for winning the election is negligible. In case of MLWCM, selected cluster-heads are comparatively stable and healthier. In MLWCM concept of inmate affiliation allow the uncovered node to merge with nearby cluster that yields reasonable amount of clusters.

By keeping ideal cluster head degree ***h_value*** according to outstanding battery life, the number of members per cluster are fairly allotted as shown in Fig. 6.10(6.10(a)- 6.10(l)) and Table 6.4. Also MLWCM identified 16 nodes out of 40 that are not suitable for cluster-head role, hence spare them form heavy weight calculation.

Next the simulation stages of cluster formation in MLWCM are shown in figure 6.10(a-l).

Stage 1: Initial topology of 40 nodes of an ad-hoc network as shown in Fig. 6.10 (a)

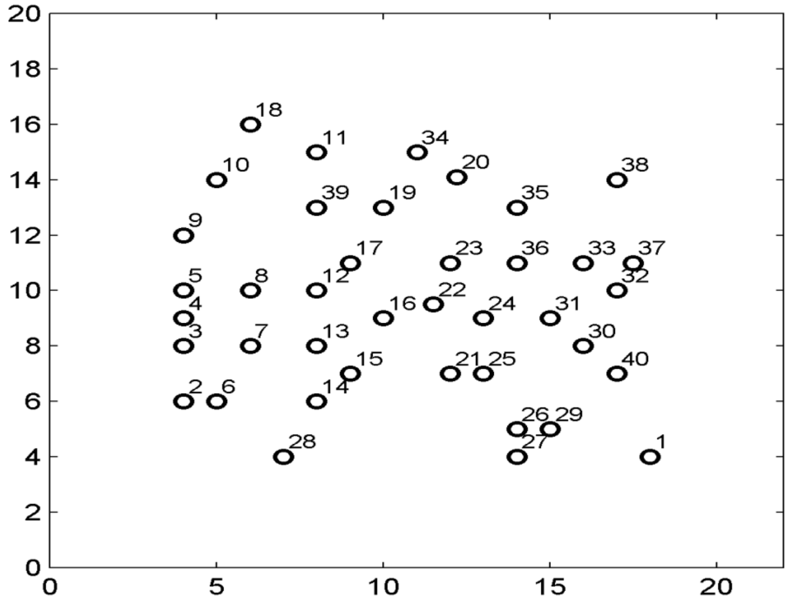


Figure 6.10(a): Stage 1 of MLWCM

Stage 2: Marking of weak nodes before starting the election process, shown as grey circles in Fig. 6.10 (b)

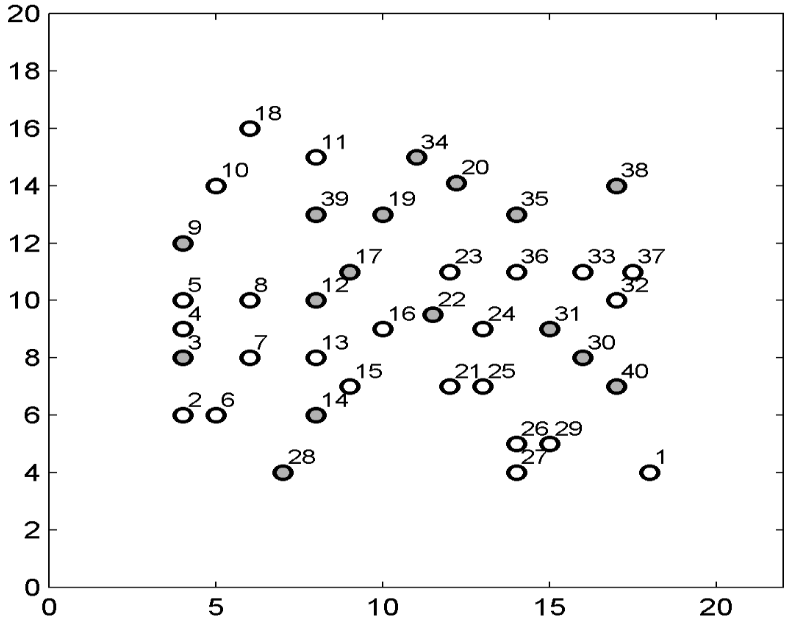


Figure 6.10(b): Stage 2 of MLWCM

Stage 3: Node 16 elected as cluster-head and formed its cluster, as shown in Fig. 6.10(c)

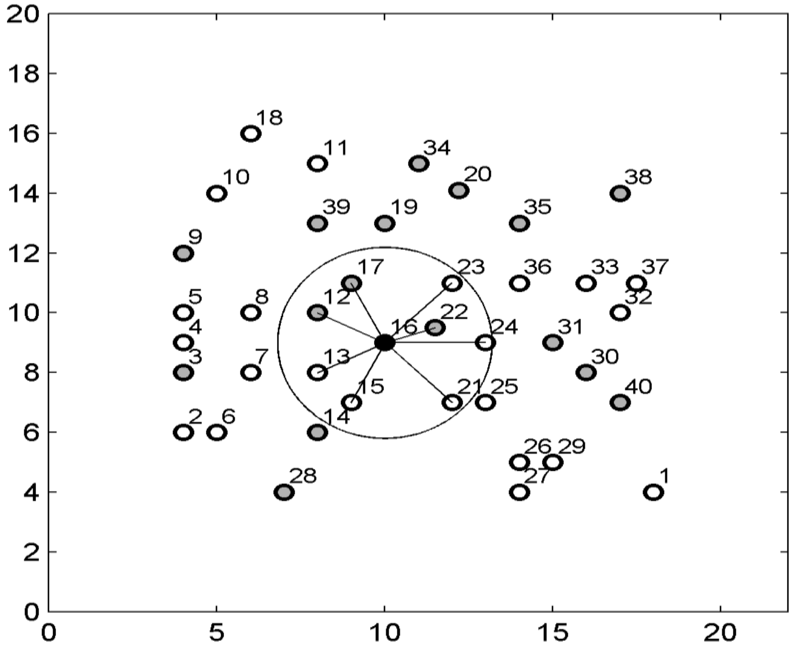


Figure 6.10(c): Stage 3 of MLWCM

Stage 4: Node 11 elected as cluster-head and formed its cluster, as shown in Fig. 6.10(d)

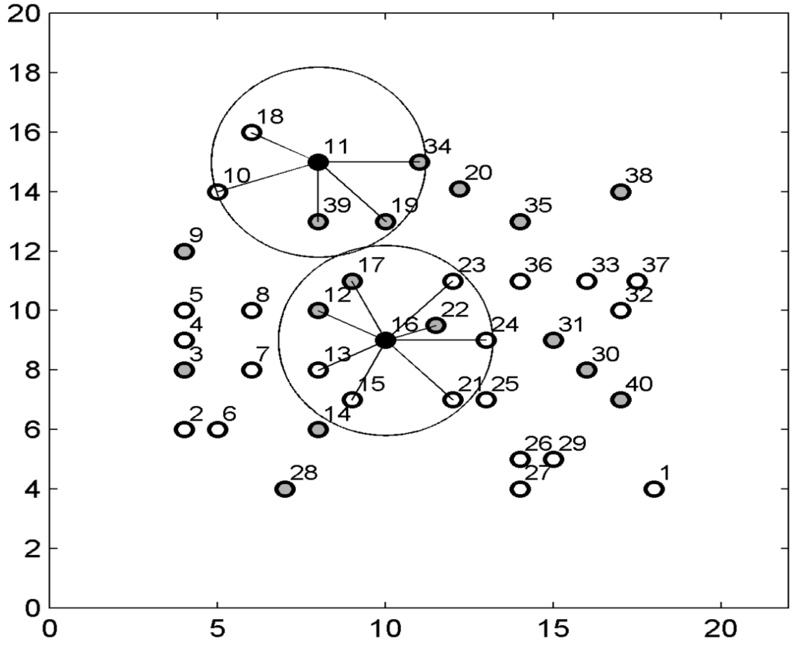


Figure 6.10(d): Stage 4 of MLWCM

Stage 5: Node 33 elected as cluster-head and formed its cluster, as shown in Fig. 6.10(e)

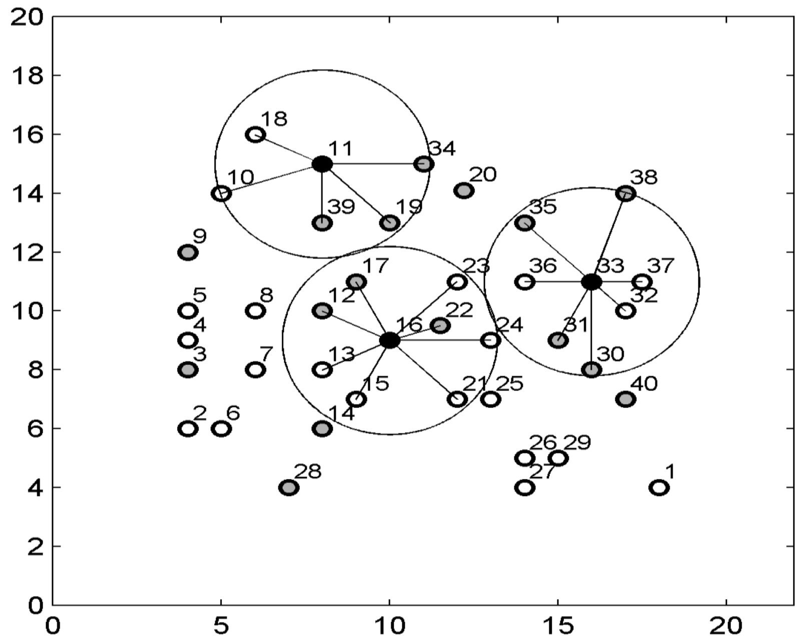


Figure 6.10(e): Stage 5 of MLWCM

Stage 6: Node 29 elected as cluster-head and formed its cluster, as shown in Fig. 6.10(f)

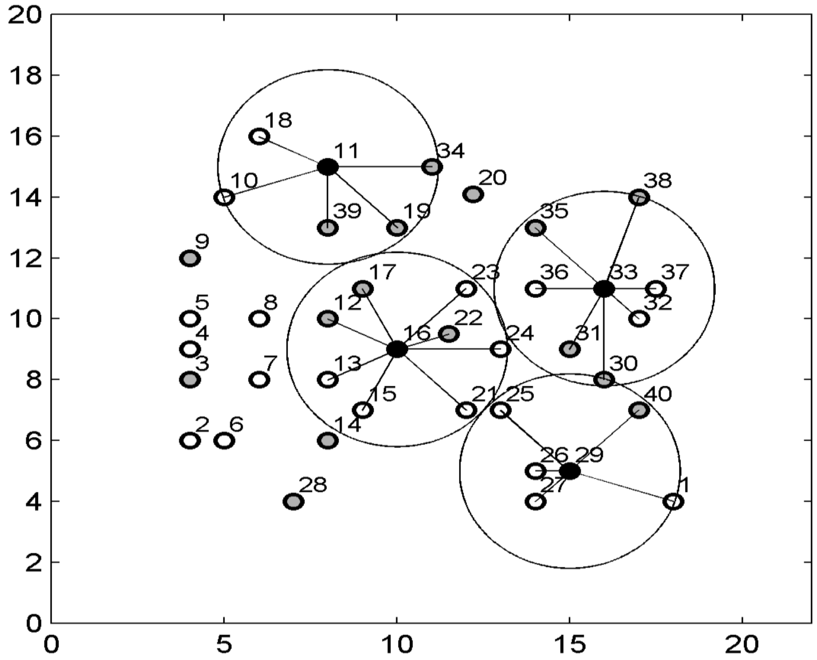


Figure 6.10(f): Stage 6 of MLWCM

Stage 7: Node 7 elected as cluster-head and formed its cluster, as shown in Fig 6.10 (g)

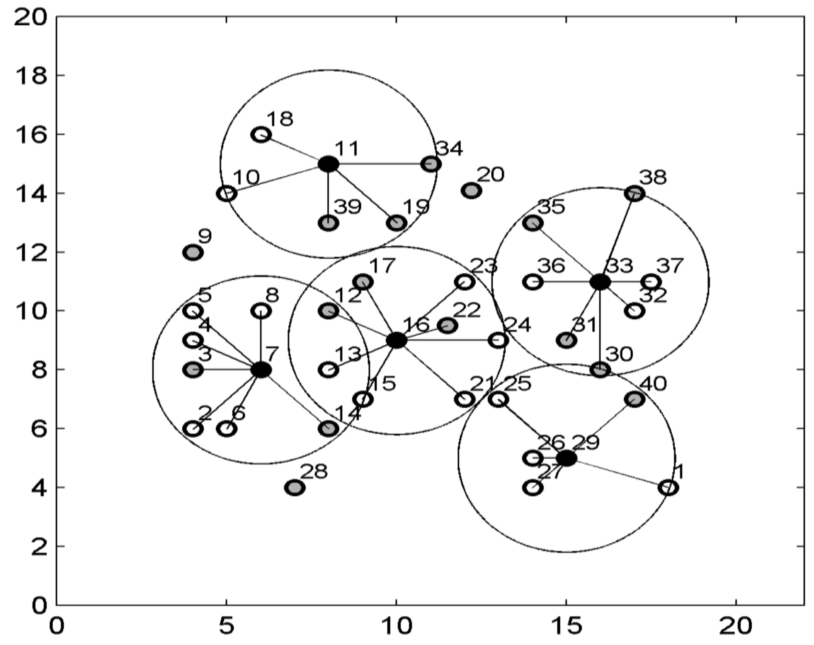


Figure 6.10(g): Stage 7 of MLWCM

Stage 8: Node 9 elected as cluster-head and formed its cluster, as shown in Fig. 6.10 (h)

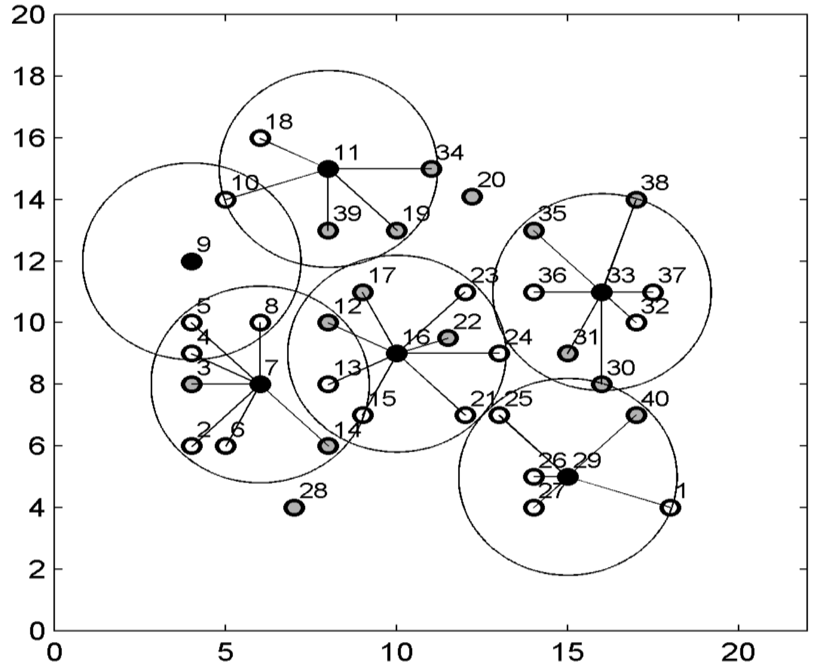


Figure 6.10(h): Stage 8 of MLWCM

Stage 9: Node 28 elected as cluster-head and formed its cluster, as shown in Fig. 6.10 (i)

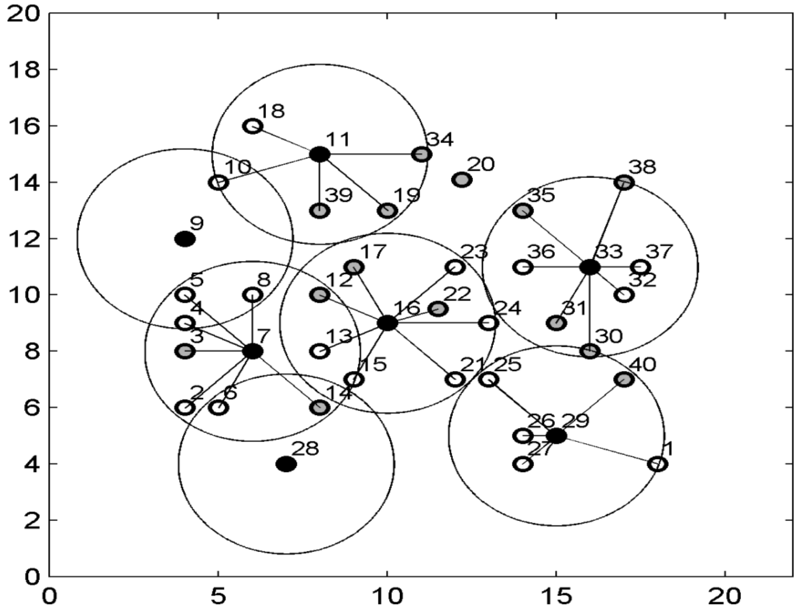


Figure 6.10(i): Stage 9 of MLWCM

Stage 10: Node 20 elected as cluster-head and formed its cluster, as shown in Fig. 6.10(j)

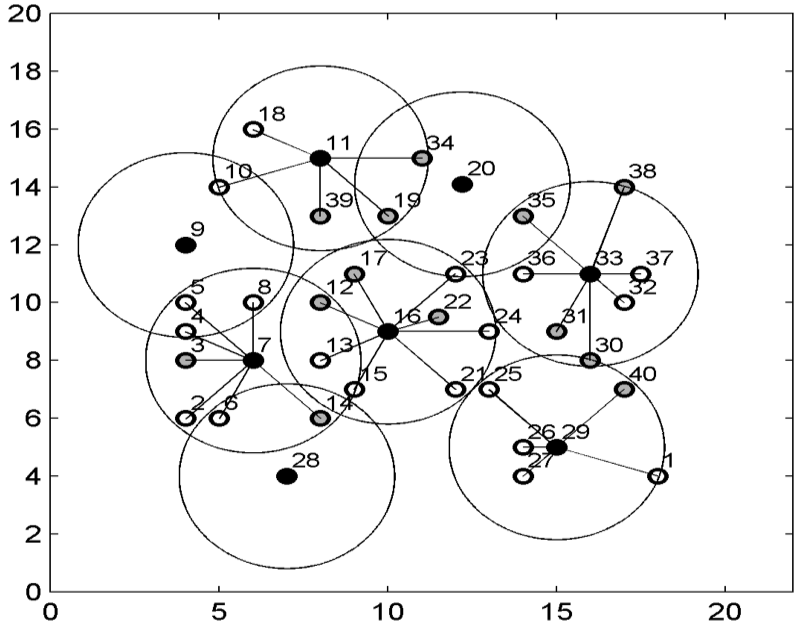


Figure 6.10(j): Stage 10 of MLWCM

Stage 11: Node 9 merged with nearby least burdened cluster as inmate, as shown in Fig. 6.10(k)

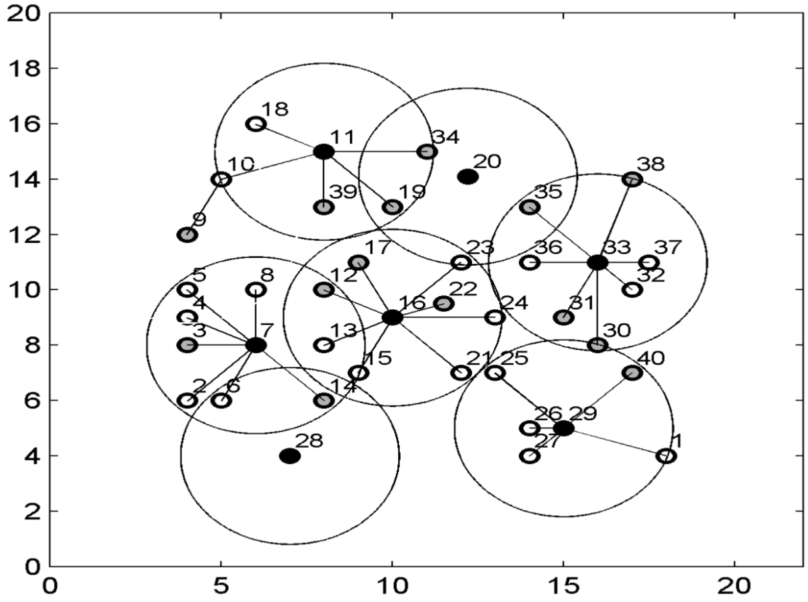


Figure 6.10(k): Stage 11 of MLWCM

Stage 12: Like node 9, node 28 and 20 merged with nearby least-burdened-clusters as inmate and formed final cluster structure with only 5 clusters, as shown in Fig. 6.10(l)

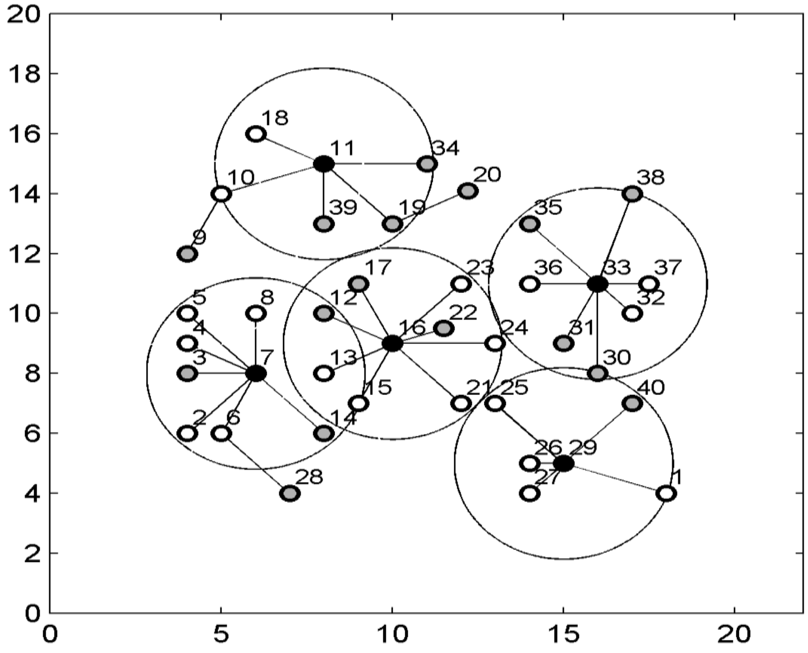


Figure 6.10(l): Stage 12 of MLWCM

Simulation finding of cluster formation in case of MLWCM of phase I are shown in Table 6.4

Table 6.4: Simulation results of phase I of cluster formation using MLWCM

S.N.	CH	MEMBERS	NO_OF Members	Handling factor	Outstanding power	Mobility	Competence Factor_Of CH
1	16	12,13,15,17, 21, 22,23,24	8	2	90	0.0100	66.6000
2	11	10,18,19,34 39	5	3	77	0.0100	61.7000
3	33	30,31, 32,35,36,37,38	7	1	67	0.0100	57.1000
4	29	1,25,26, 27,40	5	1	60	0.0100	54.0000
5	7	2,3,4,5, 6, 8 14	7	1	85	0.1000	32.4000
6	9	-	0	-6	19	4.0000	4.6750
7	28	-	0	-8	10	7.8300	1.0383
8	20	-	0	-6	8	6.1800	0.2485

Simulation finding of cluster formation in case of MLWCM of phase II are shown in Table 6.5

Table 6.5: Simulation results of merging phase of cluster formation using MLWCM

S.N.	CH	MEMBERS	NO_OF Members	Handling factor	Outstanding power	Mobility	Competence Factor_Of CH
1	16	12,13,15,17, 21,22 ,23,24	8	2	90	0.0100	66.6000
2	11	10,18,19,34 39,9,20	5	1	77	0.0100	61.7000
3	33	30,31, 32,35,36,37,38	7	1	67	0.0100	57.1000
4	29	1,25,26, 27,40	5	1	60	0.0100	54.0000
5	7	2,3,4,5, 6, 8 14,28	7	0	75	0.1000	32.4000

Comparison of simulation findings of cluster formation using WCA and MLWCM has been made and presented in Table 6.6.

Table 6.6: Comparison of Outcomes of WCA and MLWCM cluster formation

WCA				MLWCM			
CH	NO_OF Members	Outstanding power	Mobility	CH	NO_OF Members	Outstanding power	Mobility
39	5	16	0.0100	16	8	90	0.0100
21	6	64	0.0100	11	5	77	0.0100
40	5	12	0.0100	33	7	67	0.0100
33	4	67	0.0100	29	5	60	0.0100
34	1	18	5.4500	7	7	75	0.1000
23	0	56	2.1200	-	-	-	-
7	8	85	0.1000	-	-	-	-
18	0	63	3.7100	-	-	-	-
28	0	10	7.8300	-	-	-	-
9	0	19	4.0000	-	-	-	-
27	0	50	5.6100	-	-	-	-

It can be comprehend from the Table 6.4, 6.5 and Fig. 6.10 (a-l) that MLWCM generates 8 clusters of single hop in the first phase. Thereafter trivial clusters containing only cluster-heads 9, 28 and 20 get merged as inmate node with nearby clusters considering their handling factors and end up with only 5 well-formed 2-hop clusters which reduce the hop count between the communicating terminals significantly.

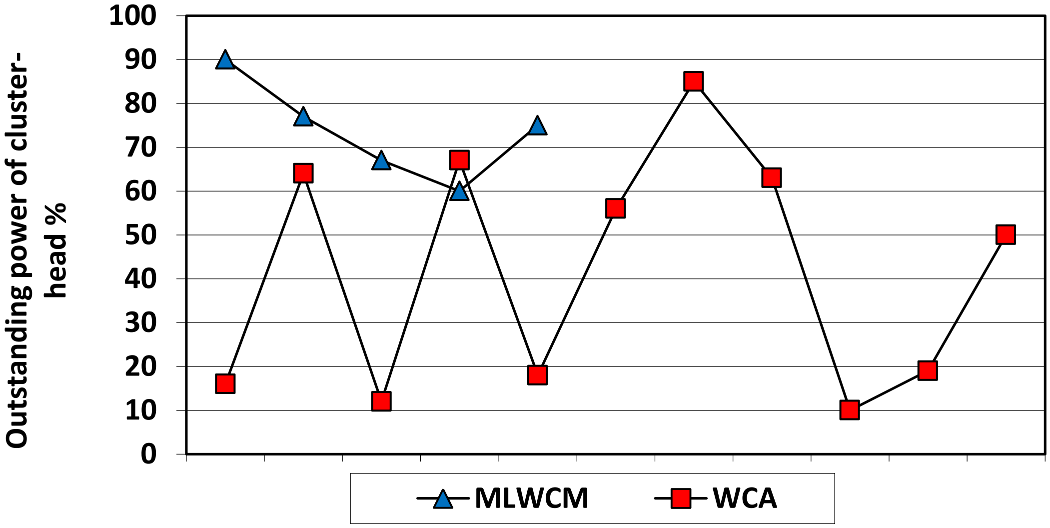


Figure 6.11: Outstanding power of cluster-head in WCA and MLWCM

Fig. 6.11 representing graphically the outstanding power level of elected cluster-heads in both WCA and MLWCM. It reveals that MLWCM elected the healthier cluster-heads comparatively which prolong the network life.

Fig. 6.12 representing graphically the mobility of elected cluster-heads. It also reveals that MLWCM elected relatively stable cluster-head.

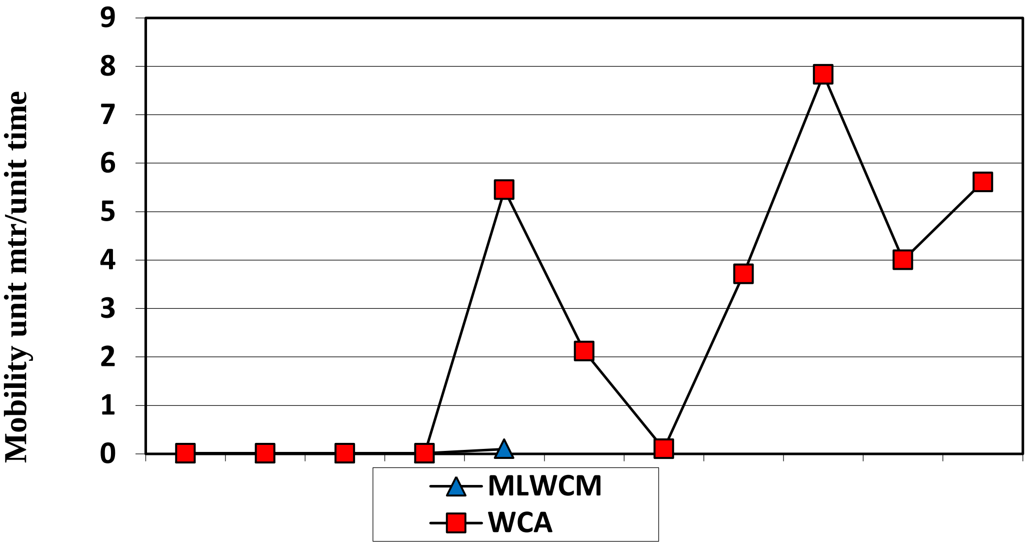


Figure 6.12: Mobility of cluster-head in WCA and MLWCM

Distribution of members throughout the clusters with respect to cluster-head outstanding power is comprehended in the Fig. 6.13 and Fig 6.14 for WCA and MLWCM respectively.

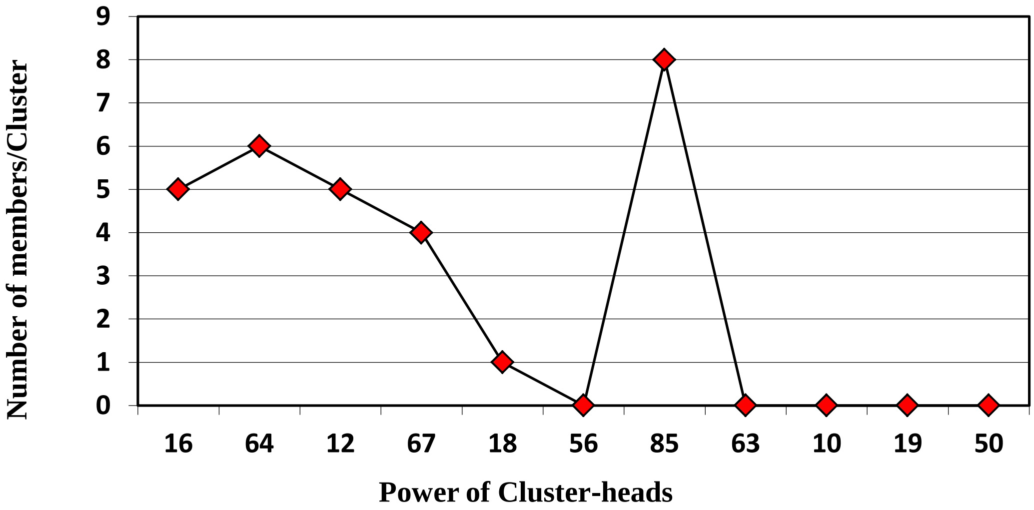


Figure 6.13: Member assigned Vs Outstanding power of cluster-head in WCA

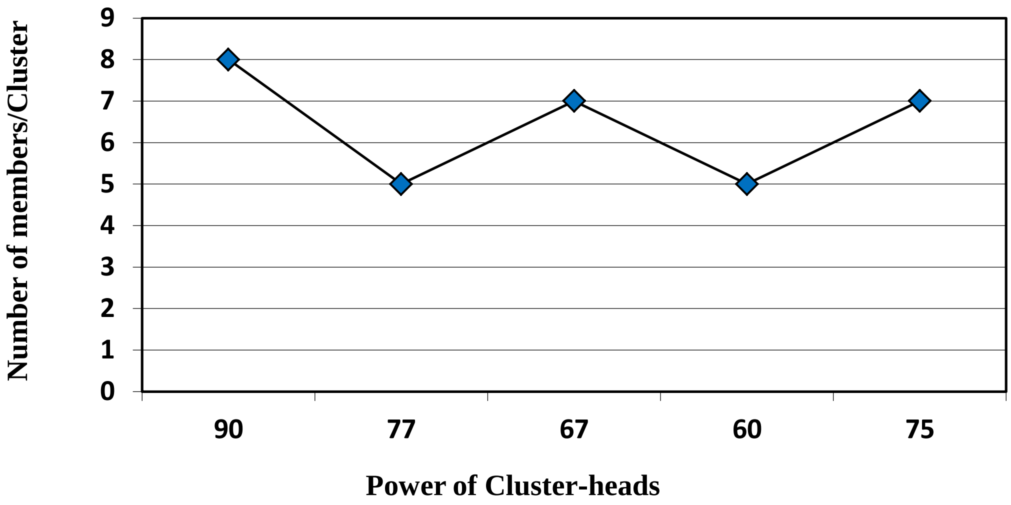


Figure 6.14: Member assigned Vs Outstanding power of cluster-head in MLWCM

It can be observed that the assignment of members to cluster-heads is not reasonable in case of WCA, while in case of MLWCM, members assigned to cluster-heads are almost uniformly distributed according to outstanding battery power.

6.5 SUMMARY

In this work standard cumulative weight based cluster-head selection algorithms are studied in detail and their inherent shortcomings are identified. To counter the gaps in existing algorithms a novel clustering framework "Marking based Load balancing Weighted Clustering framework for Mobile ad-hoc Network" (MLWCM) is proposed. Simulation results reveal that the exiting weighted clustering algorithm produces unnecessary number of clusters with weaker cluster-heads. While in case of MLWCM, healthy cluster-heads are selected and the concept of inmate affiliation yields the reasonable amount of clusters to cover all nodes in the network. The existing weighted clustering algorithm imposes the high calculation overhead over the network and node level. While MLWCM mark the weak nodes prior the beginning of election algorithm and spare them from massive weight calculation and saves the network resources. In existing WCA the movement or expiration of single node may trigger re-clustering in entire network that leads higher maintenance cost. On the other hand, in MLWCM inmate affiliation allows a displaced node to get attach with available nearby cluster silently without triggering the re clustering in entire network. Also in existing WCA the global ideal number of node **δ** for a cluster head, distributes the load among the cluster-heads unfairly. Whereas in MLWCM specific ***h*_**value controls the load distribution over the clusters according the outstanding battery level of cluster-heads. All in all, MLWCM partitions the network in better way by selecting the healthy and stable cluster-heads with lower maintenance cost.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

In this work structural design of mobile ad hoc network has been studied comprehensively. With the understanding of structural characteristics and relative issues, various routing protocols for mobile ad-hoc network have been studied thoroughly. In the literature review phase, **location aided routing** and **cluster-based routing algorithms** appealed because of their inherent advantages over the other classes of algorithms. Thereafter, the research work captivated on above mentioned class of algorithms.

It has been observed that, almost all location-based routing algorithms remain focused on minimising the number of participating nodes in path construction and overlook the Quality of Service. To harness the location aided routing with Quality of Service, a novel algorithm "**QoS Enabled Improved Location Aided routing(QEILA)**" has been proposed and developed, that utilizes the improved location aided routing protocol (ILAR) and has been equipped with Quality of Service check while selecting the next forwarding node for path construction. In QEILA, a novel path preservation procedure is included, which repairs the broken link locally by utilizing candidate next node table.

To improve the efficiency of existing approach, another routing algorithm "**Location Information Based Destination Converging Routing Method (LIBDCR)**" has been proposed and developed, in which path discovery process moves in forward direction only at each successive step to converge at destination. LIBDCR is adaptable to destination node's mobility as intermediate node update the RERQ with fresh location information of destination node. LIBDCR also Provides Quality of Service for real-time traffic like voice and video by selecting the reliable nodes in path having better battery life and band width using the next forwarding node table NFN

It has also been observed that in cluster based routing system, cluster-head becomes the single point of failure because they get overburdened by additional responsibility in which they have to look after its cluster members and are involved in every trivial or non-trivial transmission. To deal with this problem, a novel **"Light Weight Efficient Cluster based Routing Model for Mobile Ad-hoc Networks (LWECM)"** has been proposed that saves the cluster-head's energy by sparing them from trivial communication and that in turn results in prolonged life of cluster-head and reduces the maintenance cost.

In cluster based routing, it has also been observed that "**Weighted Clustering Algorithms (WCA)**" proved better over the other clustering algorithms but they induce the huge control overhead in election process prior to the actual routings with high maintenance cost. To address these issues, a novel clustering framework "**Marking based Load balancing Weighted Clustering framework for Mobile AD-HOC Network (MLWCM)**" has been proposed and developed. MLWCM marks the weak nodes prior to the beginning of election algorithm, spares them from massive weight calculation and saves the network resources. The concept of inmate affiliation reduces the maintenance cost considerably. MLWCM also ensures fair load distribution among the cluster-heads.

All developed algorithms with corresponding existing algorithms have been simulated and compared on **MATLAB 7.8** and findings show that proposed algorithms outperform over the existing algorithms.

7.2 FUTURE SCOPE

Though proposed algorithms have performed better, all proposed algorithms could be equipped with a prediction model. In this prediction model, the mobile nodes would be able to forecast the behavior of link/path according to its persistence and can take the necessary action proactively. The prediction of link/path could be based on the past pattern of packet flow for a defined time window. In this way, mobile node will cultivate an efficient packet routing in a controlled way for given ad-hoc network.

REFRENCES

[1]. "DARPA Home Page," http://www.darpa.mil

[2]. https://apps.dtic.mil/dtic/tr/fulltext/u2/a175642.pdf

[3]. A. Tanenbaum, *Computer Networks*, Prentice Hall, ISBN 0-13-349945-6,

[4]. https://www.routeralley.com/guides/rip.pdf

[5]. Carlos de Morais Cordeiro Dharma Prakash Agrawal “AD HOC & SENSOR NETWORK Theory and Applications” World Scientific Publishing Co. Pte. Ltd. ISBN 981-256-681-3

[6]. C. Siva Ram Murthy and B. S. Manoj “Ad Hoc Wireless Networks Architectures and Protocols” ISBN 0-13-147023-X Prentice Hall Professional Technical Reference

[7]. Ilker Bekmezci, Ozgur Koray Sahingoz and ßamil Temel “Flying Ad-Hoc Networks (FANETs): A survey” Elsevier Ad Hoc Networks Volume 11, Issue 3, May 2013, Pages 1254-1270

[8]. Aatifah Noureen, Zunaira Ilyas, Iram Shahzadi, M. Iqbal, Muhammad Shafiq, Azeem Irshad “Mobile IP Issues and Their Potential Solutions:An Overview” ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 1, No.7 , January 2014 ISSN : 2322-5157

[9]. Ouni, B., Belleudy, C. & Senn "Accurate energy characterization of OS services in embedded systems" E. J Embedded Systems (2012) 2012: 6. https://doi.org/10.1186/1687-3963-2012-6

[10]. M. Conti, S. Giordano, "Mobile ad hoc networking: milestones challenges and new research directions", IEEE Communications Magazine, vol. 52, no. 1, pp. 85-96, 2014.

[11]. Gerhard Hablinger "Traffic engineering supported by Inherent Network Management: analysis of resource efficiency and cost saving potential" INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT 2011; 21: 45–64

[12]. Zhi Ren, Guangyu Wang Qianbin, Chen and Hongbin Li "Modelling and simulation of Rayleigh fading, path loss, and shadowing fading for wireless mobile networks

“Simulation Modeling Practice and Theory Volume 19, Issue 2, February 2011, Pages 626-637

[13]. Radhika Ranjan Roy "Handbook of Mobile Ad Hoc Networks for Mobility Models" 2011 springer

[14]. C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *Computer Communication Review,* October 1994, 234-244

[15]. S. Murthy and J.J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," in *ACM Mobile Networks and Applications Journal,* October 1996, 183-197.

[16]. J. J. Garcia-Luna-Aceves and M. Sphon, "Source Tree Adaptive Routing," Internet Draft, draft-ierf-manet-star-00.txt, work in progress, October 1999.

[17]. P. Jacquet, P. Miihlethaler, T. Clausen, A. Laouiti, A. Qayyum, L. Viennot, "Optimized Link State Routing Protocol for Ad Hoc Networks," in Proceedings of IEEE INMIC, Pakistan, 2001.

[18]. ETSI STC-RES10 Committee, "Radio Equipment and Systems: High Performance Radio Local Area Network (HIPERLAN) Type 1," Functional Specifications, June 1996, ETS 300-652.

[19]. J. Broth, D. B. Johnson, and D. A. Maltz, "The dynamic source routing protocol for mobile ad hoc networks," Internet Draft, 1998.

[20]. D. Johnson and D. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," Mobile Computing, Kulwer Academic, 1996, 153-181.

[21]. C. E. Perkins and E. Royer, "Ad hoc on-demand distance vector routing," in IEEE Workshop on Mobile Computing Systems and Applications, February 1999, 90-100.

[22]. V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile and wireless networks," in Proceeding of IEEE Infocom, April 1997, 103-112.

[23]. M.S. Corson and V.D. Park, "An Internet MANET Encapsulation Protocol (IMEP) Specification," Internet-Draft, November 1997.

[24]. J.P. Grady, A. McDonald. “State of the Art: Ad HocNetworking”. M-Zones State of the Art Paper, SOA paper 05/03. 2003

[25]. M. Gerla, J. Tsai. “Multicluster, mobile, multimedia radio network”. Wireless networks. Vol 1. 1995. pp. 255-265.

[26]. M. Alinci, E. Spaho, A. Lala and V. Kolici, "Clustering Algorithms in MANETs: A Review," 2015 Ninth International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS), Santa Catarina, Brazil, 2015, pp. 330-335.

[27]. Harjeet Singh Chauhan, “Comparative Study of Clustering Based Algorithm in Mobile Ad-Hoc Network”, International Journal of Recent Technology and Engineering (IJRTE), Volume-2, Issue-3, July 2013.pp 110-113

[28]. B. An, S. Papavassiliou. “A mobility-based clustering approach to support mobility management and multicast routing in mobile ad-hoc wireless networks”. International Journal of Network Management. Vol. 11. 2001. pp. 387 395.

[29]. I. Er and W. Seah. “Mobility-based d-hop clustering algorithm for mobile ad hoc networks”. IEEE Wireless Communications and Networking Conference. Vol. 4. 2004. pp. 2359-2364.

[30]. W. Lou, J. Wu. “An enhanced message exchange mechanism in cluster-based mobile ad hoc networks”. Springer-Verlag Berlin Heidelberg ISPA. 2004. pp. 223-232.

[31]. M. Jiang, J. Li, and Y. C. Tay, "Cluster based routing protocol (CBRP) functional specification," Internet Draft, 1998.

[32]. A. Ephremides, J.E. Wieselthier, D.J. Baker. “A design concept for reliable mobile radio networks with frequency hopping signaling”. Proc. IEEE 75. 1987. pp. 56-73.

[33]. A. Parekh. “Selecting routers in ad hoc wireless networks”. Proceedings of the SBT/IEEE International Telecommunications Symposium. 1994.

[34]. G. Chen, F. Nocetti, J. Gonzalez, and I. Stojmenovic, “Connectivity based k-hop clustering in wireless networks”. Proceedings of the 35th Annual Hawaii International Conference on System Sciences. Vol. 7. 2002. pp. 188.3.

[35]. P. Basu, N. Khan, T. D. C. Little. “A mobility based metric for clustering in mobile ad hoc networks”. Proceedings of the 21st International Conference on Distributed Computing Systems. 2001. pp. 413.

[36]. Young-jun Oh and Kang-whan Lee “A Clustering Algorithm Based on Mobility Properties in Mobile Ad Hoc Networks” Hindawi Publishing Corporation

International Journal of Distributed Sensor Networks Volume 2015, Article ID 567269, 15 pages pp: 1-15

[37]. C. Chiang, Hsiao- Kuang Wu, Winston Liu and Mario Gerla, (1997) "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," in Proc. IEEE SICON'97.

[38]. J. Y. Yu and P. H. J. Chong,(2003) "3hBAC (3-hop between Adjacent Clusterheads): a Novel Non-overlapping Clustering Algorithm for Mobile Ad Hoc Networks," in Proc. IEEE Pacrim'03, vol. 1, Aug. 2003, pp. 318–21.

[39]. M. Chatterjee, S. K. Das, and D. Turgut,(2002) "WCA: A weighted clustering algorithm for mobile ad hoc networks," Cluster Computing, vol. 5, no. 2, pp. 193–204, 2002.

[40]. J. C. Navas and T. Imielinski, "Geographic Addressing and Routing," Proc. 3rd ACM/IEEE Int. Conf. Mobile Comp. Net., MobiCom'97, Sept. 1997.

[41]. H. Takagi and L. Kleinrock, "Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals," IEEE Trans. Commun., vol. 32, no. 3, Mar. 1984, pp. 246–57.

[42]. E. Kranakis, H. Singh, and J. Urrutia, "Compass Routing on Geometric Networks," Proc. 11th Canadian Conf. Comp. Geo., Vancouver, Aug. 1999.

[43]. R. Nelson and L. Kleinrock, "The Spatial Capacity of a Slotted Aloha Multihop Packet Radio Network with Capture," IEEE Trans. Commun., vol. 32, no. 6, June 1984, pp. 684–94.

[44]. T.-C. Hou and V. O.K. Li, "Transmission Range Control in Multihop Packet Radio Networks," IEEE Trans. Commun., vol. 34, no. 1, Jan. 1986, pp. 38–44.

[45]. B. N. Karp, "Geographic Routing for Wireless Networks," Ph.D. thesis, Harvard Univ., 2000.

[46]. Y.-B. Ko and N. H. Vaidya, "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks," ACM/Baltzer WINET J., vol. 6, no. 4, 2000, pp. 307–21.

[47]. S. Basagni, Imrich, Violet R. Syrotiuk "A Distance Routing Effect Algorithm for Mobility (Dream)," Proc. 4th Annual ACM/IEEE Int. Conf. Mobile Computing and Networking, MOBICOM '98, Dallas, TX, USA, 1998, pp. 76–84.

[48]. Neng-Chung Wang,Yung-Fa Huang,Jong-Shin ChenAn "Improved Location-Aided Routing Protocol for Mobile Ad Hoc Networks with Greedy Approach"WSEAS TRANSACTIONS on COMMUNICATIONS ,ISSN: 1109-2742,Issue 8, Volume 8, August 2009.

[49]. L. Blazevic Ljubica Blazevic, Levente Buttyán, Srdjan Capkun, Silvia Giordano, Jean-Pierre Hubaux, J.-Y. Le Boudec, “Self-Organization in Mobile Ad Hoc Networks: The Approach of Terminodes,” IEEE Commun. Mag., 2001.

[50]. The grid project homepage, http://www.pdos.lcs.mit.edu/grid

[51]. L. Raghavendar Raju, C.R.K .Reddy” A Survey on Routing Protocols and QoS in Mobile Ad Hoc Networks (MANETs)” Indian Journal of Science and Technology, Vol 10(8), February 2017

[52]. Matthias Riedel,YigangXu "QoS-Aware Hhandover Procedure For Mobilead-Hocnetwork Environments" United States Patent US7693093B2 April 6 2010.

[53]. Mohammad Shojafar Nicola Cordeschi Adaptive, Jemal H Abawajy, Enzo Baccarelli “Energy-Efficient QoS-Aware Scheduling Algorithm for TCP/IP Mobile Cloud" Globecom Workshops (GC Wkshps) IEEE San Diego, CA, USA, 6 Dec 2015

[54]. HosseiniNazhad ,MohammadShojafar"An efficient routing protocol for the QoS support of large- scale MANETs" Wiley Int J Commun Syst. 2017;e3"

[55]. P.K.Suri, M.K.Soni, and ParulTomar “ANovel Approach for QoS in MANET” Journal Of Computing, Volume 2, Issue 12, December 2010.

[56]. V. V. Mandhare R. C.Thool “Improving QoS of Mobile Ad-hoc Network using Cache Update Scheme in Dynamic Source Routing Protocol” Procedia Computer Science 79 (2016) 692 – 699 Elsevier

[57]. Seyed Hossein Hosseini Nazhad Mohammad Shojafar, Shahaboddin Shamshirband “An efficient routing protocol for the QoS support of large‐scale MANETs” International J of Communication Syst. 2018 (Wiley)

[58]. Usman A. D., Sulieman S.M, Agbon E. E and W F Wan Ahmed “Throughput Efficient AODV for Improving QoS Routing in Energy Aware Mobile Ad-Hoc Network” Covenant Journal of Informatics & Communication Technology. Vol. 6 No. 1, June, 2018

[59]. Z.J. Haas and M.R. Pearlman Determining the optimal configuration for the zone routing protocol IEEE Journal on Selected Areas in Communications (Volume: 17, Issue: 8, Aug 1999)

[60]. Z. J. Haas, “A New Routing Protocol for the Reconfigurable Wireless Networks,” Proceedings of the IEEE International Conference on Universal Personal Communications (ICUPC), pp. 562-566, October 1997.

[61]. G. Pei, M. Gerla, X. Hong, and C.-C. Chiang, “A Wireless Hierarchical Routing Protocol with Group Mobility,” Proceedings of 1999 IEEE Wireless Communications and Networking Conference, New Orleans, USA, pp. 1538-1542, September 1999.

[62]. G. Dommety and R. Jain. “Potential Networking Applications of Global Positioning System (GPS),” Technical Report TR-24, Computer Science Department, The Ohio State University, April 1996.

[63]. G. E. Rizos, D. C. Vasiliadis, and E. Stergioy, “Improving Performance on Mobile Ad Hoc Networks by Using Proxy Service,” WSEAS Transactions on Communications, Vol. 7, Issue 12, pp. 1137-1146, December 2008.

[64]. E. M. Royer and C.-K. Toh, “A Review of Current E. M. Royer and C.-K.Toh, “A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks,” IEEE Personal Communications, Vol. 6, No. 2, pp. 46-55, April 1999.

[65]. J. P. Macker and M. S. Corson, “Mobile Ad Hoc Networking and the IETF,” ACM SIGMOBILE Mobile Computing and Communications Reviews, Vol. 2, No. 2, pp. 9-14, January 1998.

[66]. S. Murthy and J. J. Garcia-Luna-Aceves, “A Routing Protocol for Packet Radio Networks,” Proceedings of ACM First International Conference on Mobile Computing and Networking, Berkeley, CA, USA, pp. 86-95, November 1995.

[67]. Z. J. Haas and M. R. Pearlman. “The Zone Routing Protocol for Ad-Hoc Netwroks (Internet Draft),” Mobile Ad-Hoc Networking Group, August 1998.

[68]. Tzay-FarnShih , Hsu-Chun Yen, Location-aware routing protocol with dynamic adaptation of request zone for mobile ad hoc networks, Springer Wireless Newt (2008) 14:321–333,DOI 10.1007/s11276-006-9955-y.

[69]. Dr.P.K.Suri, Dr.M.K.Soni and ParulTomar"Framework for Location Based Power Aware Routing in MANET"IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, May 2011 ISSN (Online): 1694-0814

[70]. C. Prehofer, C. Bettstetter. “Self-organization in communication networks: Principles and design paradigms”. IEEE Communications Magazine. Vol. 43. Issue 7. 2005. pp. 78-85.

[71]. S. Sesay, Z. Yang, J. He. “A survey on mobile ad hoc wireless network”. Information Technology Journal.Vol. 3. 2004. pp. 168-175.

[72]. I. Chatzigiannakis, S. Nikoletseas. “Design and analysis of an efficient communication strategy for hierarchical and highly changing ad-hoc mobile networks”. Mobile Networks and Applications. Vol. 9. 2004. pp. 319-332.

[73]. Y. Yi, M. Gerla, T. J. Kwon. “Efficient flooding in ad hoc networks: a comparative performance study”. Proc. of the IEEE International Conference on Communications (ICC). Vol. 2. 2003. pp. 1059-1063.

[74]. S. Giannoulis, C. Antonopoulos, E. Topalis, S. Koubias, “ZRP versus DSR and TORA: A comprehensive survey on ZRP performance,” In Proc. 10th IEEE Conference on Emerging Technologies and Factory Automation, Catania, 2005, pp. 1017-1024.

[75]. John Sucec , Ivan Marsic, Hierarchical Routing Overhead in Mobile Ad Hoc Networks, IEEE Transactions on Mobile Computing, v.3 n.1, p.46-56, January 2004 [doi>10.1109/TMC.2004.1261816]

[76]. L. Jia, R. Rajaraman, and T. Suel. *An Efficient Distributed Algorithm for Constructing Small Dominating Sets*, In Twentieth ACM Symposium on Principles of Distributed Computing PODC'01, Newport, Rhode Island, Aug. 26--29 2001.

[77]. Pathak, S. & Jain, S. J C.R. Lin, M. Gerla. “Adaptive clustering for mobile wireless networks”. IEEE Journal of Selected Areas in Communications. Vol. 15. 1997. pp. 1265-1275.

[78]. S. Chinara, S. Rath, “A Survey on One-Hop Clustering Algorithms in Mobile Ad Hoc Networks,” Journal of Network and Systems Management, Vol. 17, No. 1-2, 2009, pp. 183-207. doi:10.1007/s10922-009-9123-7.

[79]. O. Younis and S. Fahmy “HEED: A Hybrid, EnergyEffcient, Distributed Clustering Approach for Ad Hoc Sensor Networks,” IEEE Transactions on Mobile Computing, Vol. 3, No. 4, 2004, pp. 366–379.

[80]. M. Anupama and B. Sathyanarayana “Survey of Cluster Based Routing Protocols in Mobile Ad hoc Networks,” International Journal of Computer Theory and Engineering, Vol. 3, No. 6, 2011.

[81]. Y.Y. Su, S.F. Hwang, and C.R. Dow, “An EfficientCluster-Based Routing Algorithm in Ad Hoc Networks with Unidirectional Links,” Journal of Information Science and Engineering 24, 2008, pp.1409-1428.

[82]. Jane y. Yu and peter h. J. Chong "a survey of clustering schemes for mobile ad hoc networks" first quarter, volume 7, no.1 IEEE communications surveys 2005

[83]. Blanca Alicia; L. Ospina; and R. C. Hincapié, “Survey of Clustering Techniques for Mobile ad hoc Networks,” Rev. Fac. Ing. Univ. Antioquia, no. 41, pp. 145–161, 2007.

[84]. A. B. Bondi. Characteristics of scalability and their impact on performance. In Proceedings of the 2nd international workshop on Software and performance, pages 195–203. ACM, 2000.

[85]. A. D. Amis and R. Prakash. Load-balancing clusters in wireless ad hoc networks. In Application-Specific Systems and Software Engineering Technology, 2000. Proceedings. 3rd IEEE Symposium on, pages 25–32. IEEE, 2000.

[86]. Z. Li and H. Shen. Game-theoretic analysis of cooperation incentive strategies in mobile ad hoc networks. IEEE Transactions on Mobile Computing, 11(8):1287–1303, 2012.

[87]. C. E. Perkins , Ad Hoc Networking, Addison-Wesley, 2001.

[88]. E. M. Belding-Royer, “Hierarchical Routing in Ad Hoc MobileNetworks,” Wireless Commun. and Mobile Comp., vol. 2, no.5, , pp. 515–32. (2002)

[89]. T.-C. Hou and T. J. Tsai,(2001) “An Access-Based Clustering Protocol for Multihop Wireless Ad Hoc Networks,” IEEE JSAC, vol. 19,no. 7, July., , pp. 1201–10.

[90]. U. C. Kozat, George Kondylis, Bo RyuBo Ryu and M.K. Marina (2001) “Virtual Dynamic Backbone for Mobile AdHoc Networks,” Proc. IEEE ICC’01, vol. 1, June 2001, pp.250–55.

[91]. A. B. MaDonald and T. F. Znati,(1999) “A Mobility-based Frame Workfor Adaptive Clustering in Wireless Ad Hoc Networks,” IEEE JSAC, vol. 17, Aug. 1999., pp. 1466–87.

[92]. S. J. Francis and E. B. Rajsingh, “Performance Analysis of Clustering Protocols in Mobile Ad hoc Networks,” Journal of Computer Science, 2008, pp. 192-204.

[93]. A. Fathi and H. Taheri. “Enhance Topology Control Protocol(ECEC) to Conserve Energy based clustering in Wireless Ad Hoc Networks,” 3rd IEEE ICCSIT, 2010.

[94]. M.R Monsef, S. Jabbehdari and F. Safaei. “An Efficient Weight-Based Clustering Algorithm for Mobile Ad-hoc Networks,” Journal of Computing, Vol. 3, 2011.

[95]. N. Gupta, M. Shrivastava, A. Singh, “Cluster Based on Demand Routing Protocol for Mobile Ad Hoc Network,” IJERT, Vol. 1, No. 3, 2012.

[96]. R. Pandi Selvam and V.Palanisamy “Stable and Flexible Weight based Clustering Algorithm in Mobile Ad hoc Networks” International Journal of Computer Science and Information Technologies 2011, Vol. 2 (2), 2011, 824-828.

[97]. S. Muthuramalingam, R. RajaRam, Kothai Pethaperumal and V. Karthiga Devi. A Dynamic Clustering Algorithm for MANETs by modifying Weighted Clustering Algorithm with Mobility Prediction. *International Journal of computer and Electrical Engineering* August, 2010, Vol. 2, No. 4, 1793-8163.

[98]. Hui Cheng Jiannong, Cao Xingwei Wang, Sajal K. Das Shengxiang YangStability-aware multi-metric clustering in mobile ad hoc networks with group mobility. *Wireless Communications and Mobile Computing*; 9:759-771, Published online 21 April 2008 in Wiley InterScience.

[99]. Sunil Pathak and Sonal Jain “An optimized stable clustering algorithm for mobile ad hoc networks” EURASIP Journal on Wireless Communications and Networking (2017)

[100]. Lotf, M. Bonab, S. Khorsandi, “A Novel Cluster-based Routing Protocol with Extending Lifetime for Wireless Sensor Networks,” In Proc. 5th IFIP International

Conference on Wireless and Optical Communications Networks (WOCN08), East Java Indonesia, Surabaya, 2008, pp. 1-5.

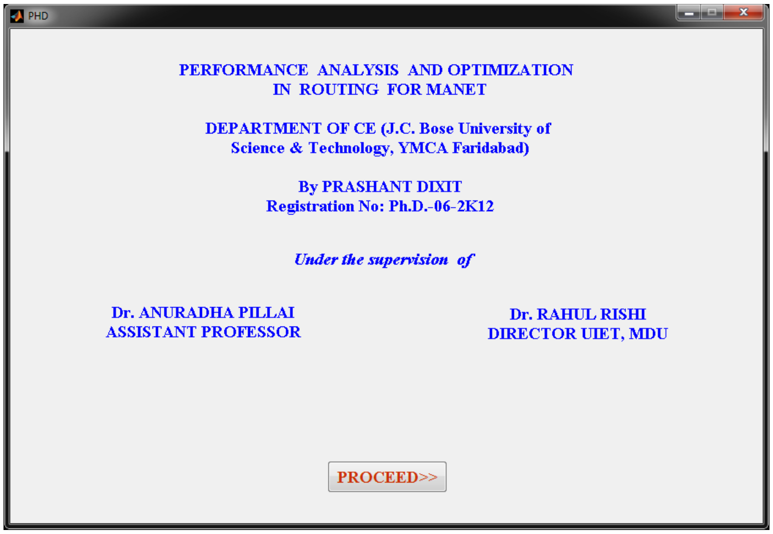
[101]. Abbas Karimi, Abbas Afsharfarnia, Faraneh Zarafshan and S. A. R. Al-Haddad. "A Novel Clustering Algorithm for Mobile Ad Hoc Networks Based on Determination of Virtual Links' Weight to Increase Network Stability" Hindawi Publishing Corporation The Scientific World Journal Volume 2014, Article ID 432952, 11 pages

[102]. Mohamed Aissa and Abdelfettah Belghith "Quality of Clustering in mobile Ad Hoc networks" Procedia Computer Science 32 (2014) 24 –252

APPENDIX

In this appendix, screenshots of graphical user interface of simulation setup, screen by screen, are presented.

Fig A.1 shows Screenshot of the title of thesis with a push button labeled as ‘PROCEED’, clicking on ‘PROCEED’ button navigates to next screen 2.



PERFORMANCE ANALYSIS AND OPTIMIZATION
IN ROUTING FOR MANET

DEPARTMENT OF CE (J.C. Bose University of
Science & Technology, YMCA Faridabad)

By PRASHANT DIXIT
Registration No: Ph.D.-06-2K12

Under the supervision of

Dr. ANURADHA PILLAI
ASSISTANT PROFESSOR

Dr. RAHUL RISHI
DIRECTOR UIET, MDU

Figure A.1: Screenshot 1of Title Screen

Fig. A.2 shows Screenshot 2 of ‘Home screen’ with four push buttons labeled as ‘QEILA’,‘LIBDCR’,’LWECM’ and ‘MLWCM’. These buttons are corresponding to the four protocols, that proposed and developed in this research work. Clicking on each of these buttons, navigates to corresponding simulation screens.

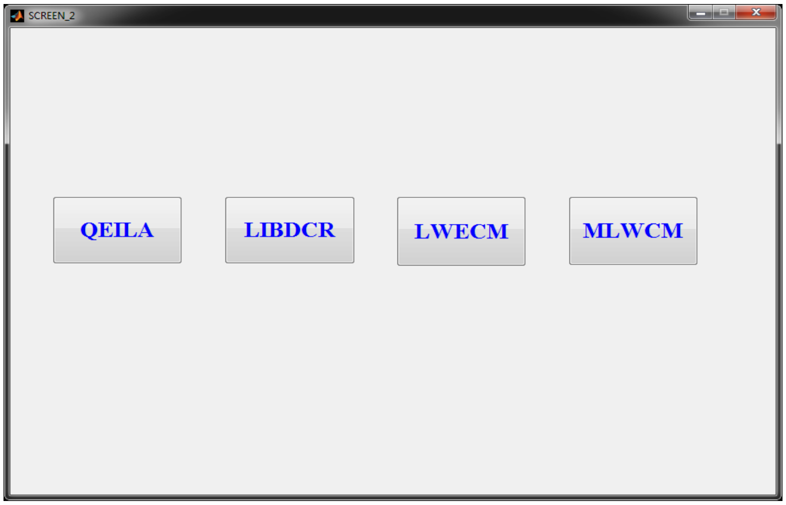


Figure A.2: Screenshot 2 of Home screen

Fig A.3 shows Screenshot of the simulation screen of ILAR and QEILA. Clicking on push buttons labeled as ‘ILAR’ and ‘QEILA’ triggers the simulations of corresponding protocol. Push buttons labeled as ‘CLEAR’ clears the simulation area and push button labeled as ‘BACK’ navigates to home screen.

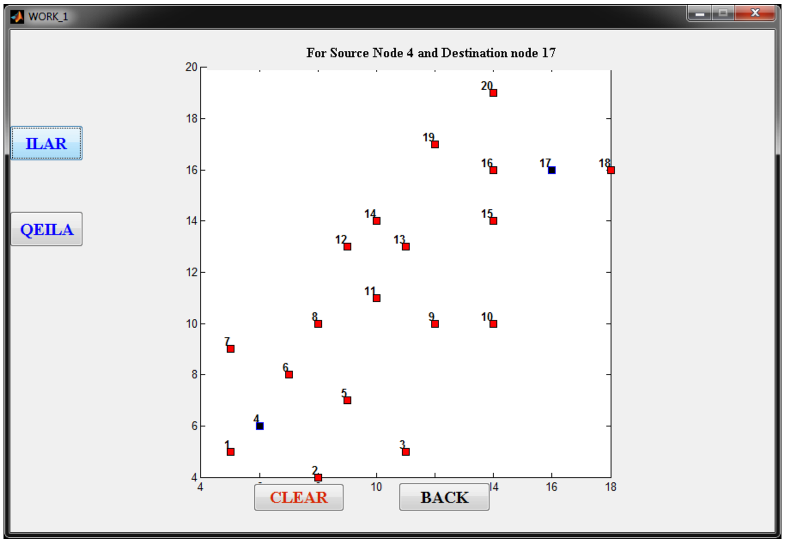


Figure A.3: Screenshot of simulation screen of ILAR and QEILA

Fig.A.4 shows the screenshot of simulation outcome of path construction in ILAR, as source node 4 and destination 17.

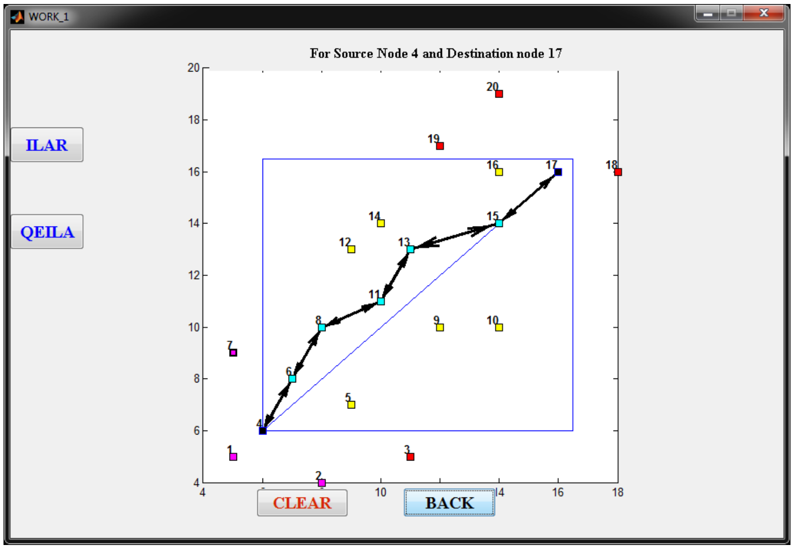


Figure A.4: Screenshot of simulation outcome of ILAR

Fig. A.5 shows the screenshot of simulation outcome of path construction in QEILA, as source node 4 and destination 17

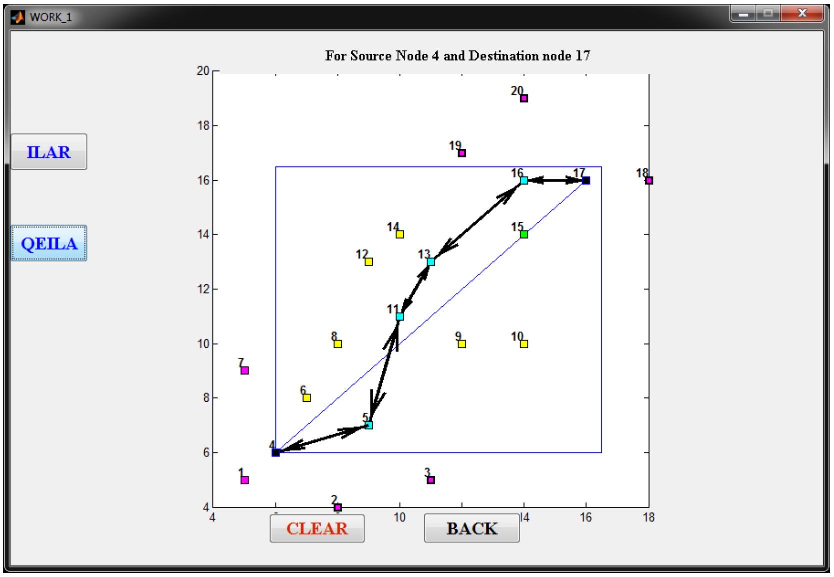


Figure A.5: Screenshot of simulation outcome of QEILA

Fig. A.6 shows the screenshot of the simulation screen of LBPAR and LIBDCR. Clicking on push buttons labeled as ‘LBPAR’ and ‘LIBDCR’, triggers the simulations of corresponding protocol. Push buttons labeled as ‘CLEAR’ clears the simulation area and push button labeled as ‘BACK’ navigates to home screen.

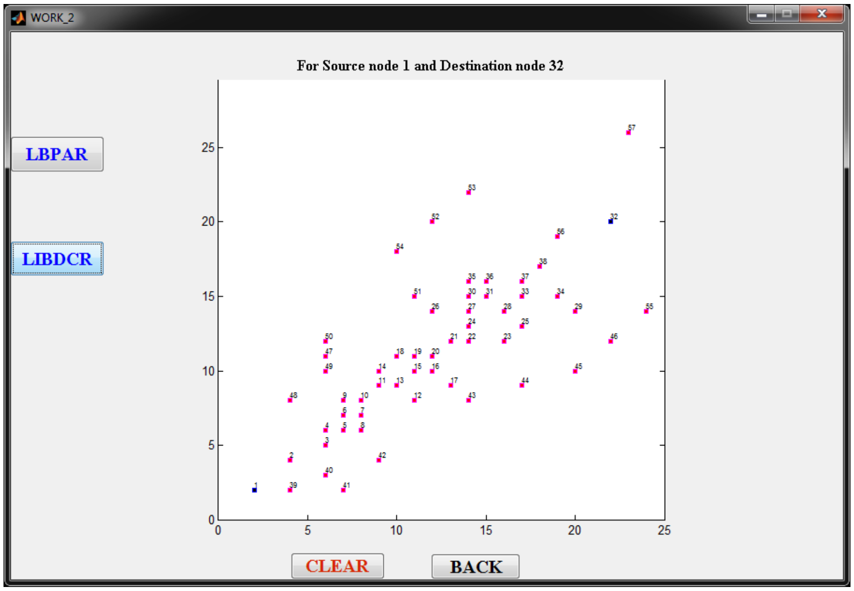
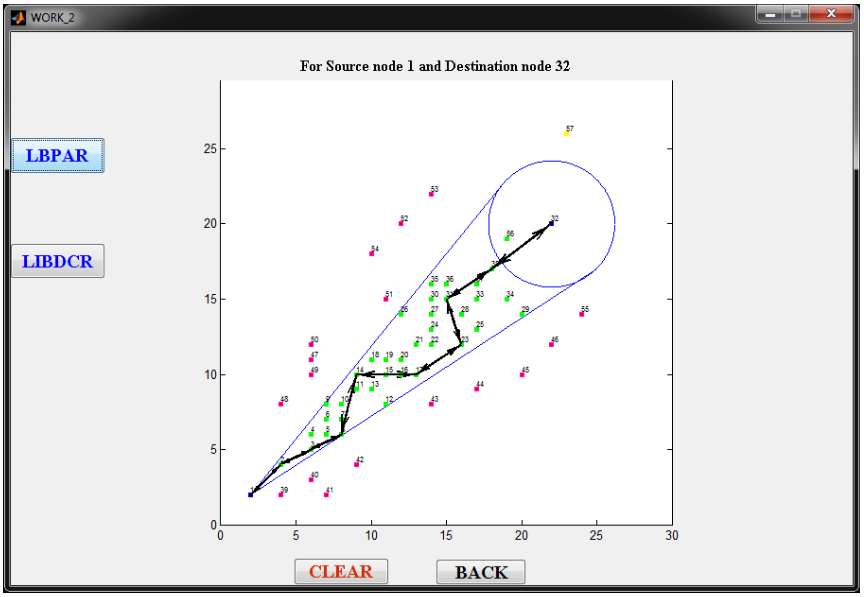


Figure A.6: Screenshot of Simulation screen of LBPAR and LIBDCR

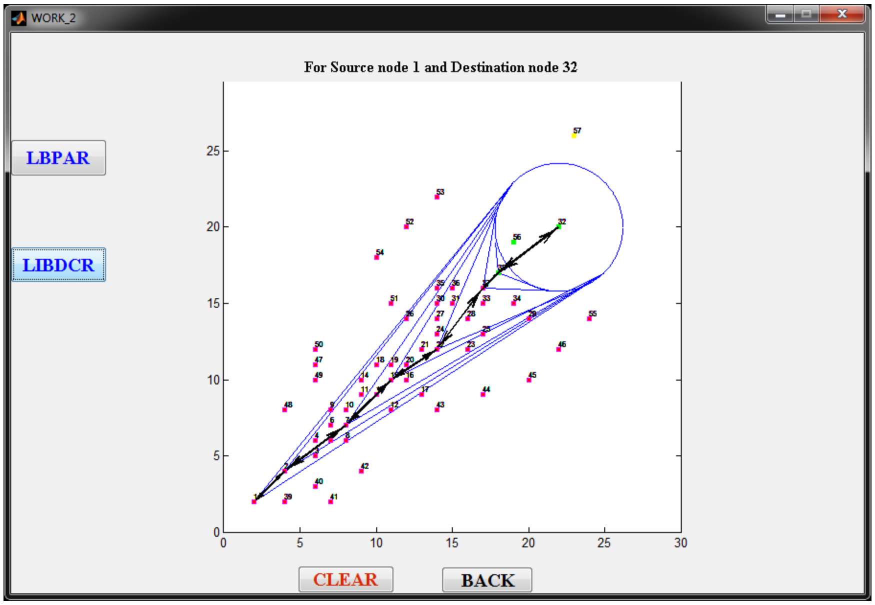
Fig. A.7 shows the screenshot of simulation outcome of path construction in LBPAR, as source node 1 and destination 32



For Source node 1 and Destination node 32

Figure A.7: Screenshot of Simulation outcome of LBPAR

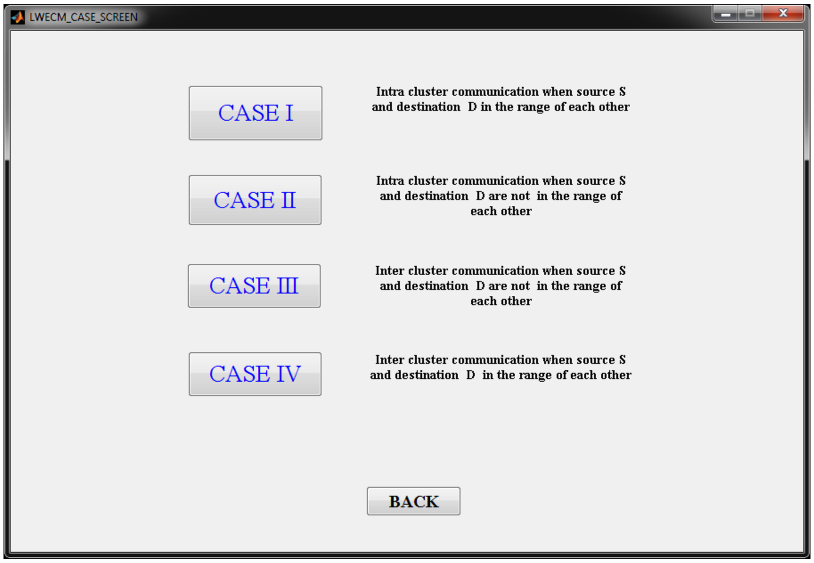
Fig. A.8 shows the screenshot of simulation outcome of path construction in LIBDCR, as source node 1 and destination 32



For Source node 1 and Destination node 32

Figure A.8: Screenshot of Simulation outcome of LIBDCR

Fig. A.9 shows the screenshot of the case screen for simulation of CBRP and LWECM. Clicking on push buttons labeled as ‘CASE I’, ‘CASE II’, ‘CASE II’, ‘CASE VI’ navigates simulations screen of corresponding case. Push buttons labeled as ‘BACK’ navigates to home screen.



Intra cluster communication when source S and destination D in the range of each other

Intra cluster communication when source S and destination D are not in the range of each other

Inter cluster communication when source S and destination D are not in the range of each other

Inter cluster communication when source S and destination D in the range of each other

Figure A.9: Screenshot of case screen of CBRP and LWECM

Fig. A.10 shows the screenshot of the simulation screen of CBRP and LWECM for case I. Clicking on push buttons labeled as ‘CBRP’ and ‘LWECM’, triggers the simulations of corresponding protocol. Push buttons labeled as ‘CLEAR’ clears the simulation area and push button labeled as ‘BACK’ navigates to case screen.

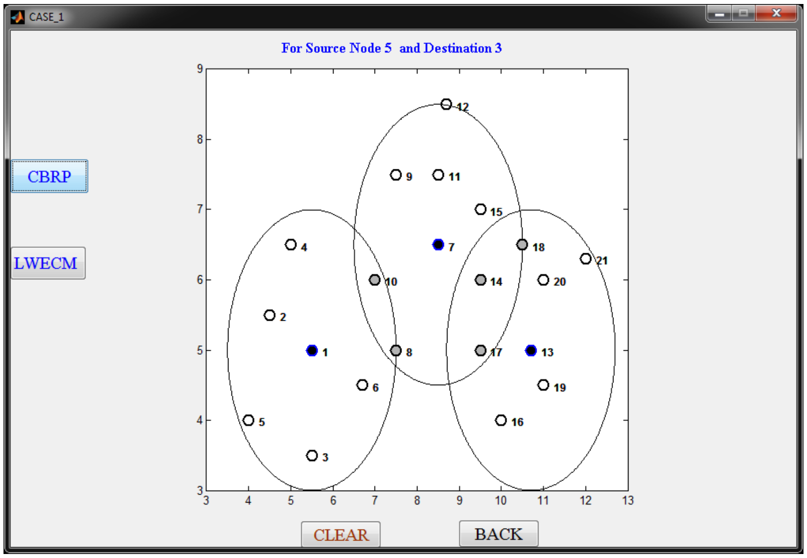


Figure A.10: Screenshot of simulation screen of CBRP and LWECM for case I

Fig. A.11 shows the screenshot of simulation outcome of path construction in CBRP for case I, as source node 5 and destination 3

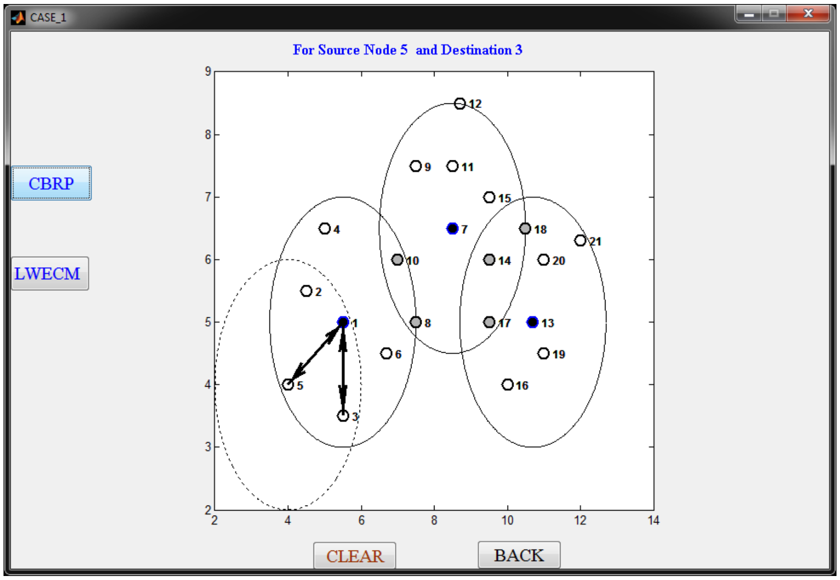


Figure A.11: Screenshot of Simulation outcome of CBRP for case I

Fig. A.12 shows the screenshot of simulation outcome of path construction in LWECM for case I, as source node 5 and destination 3.

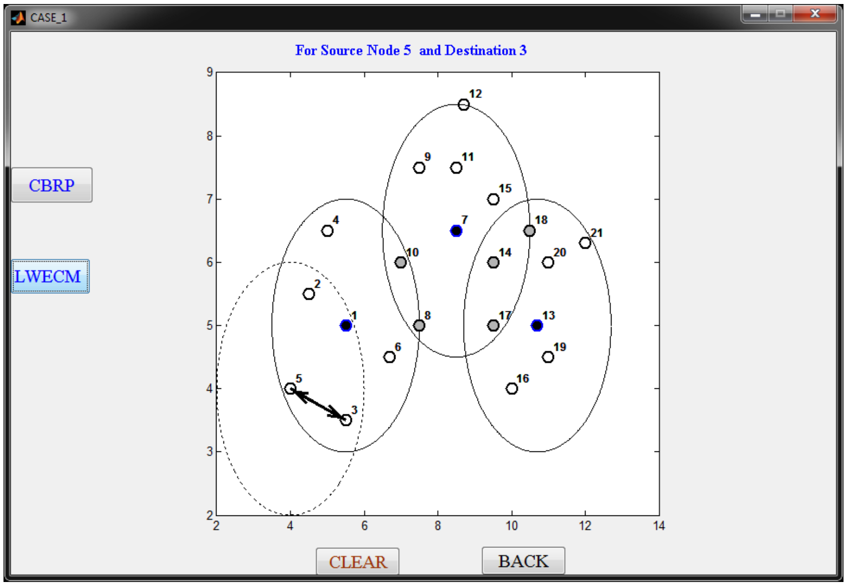


Figure A.12: Screenshot of Simulation outcome of LWECM for case I

Fig. A.13 shows the screenshot of the simulation screen of CBRP and LWECM for case II. Clicking on push buttons labeled as ‘CBRP’ and ‘LWECM’, triggers the simulations of corresponding protocol. Push buttons labeled as ‘CLEAR’ clears the simulation area and push button labeled as ‘BACK’ navigates to case screen.

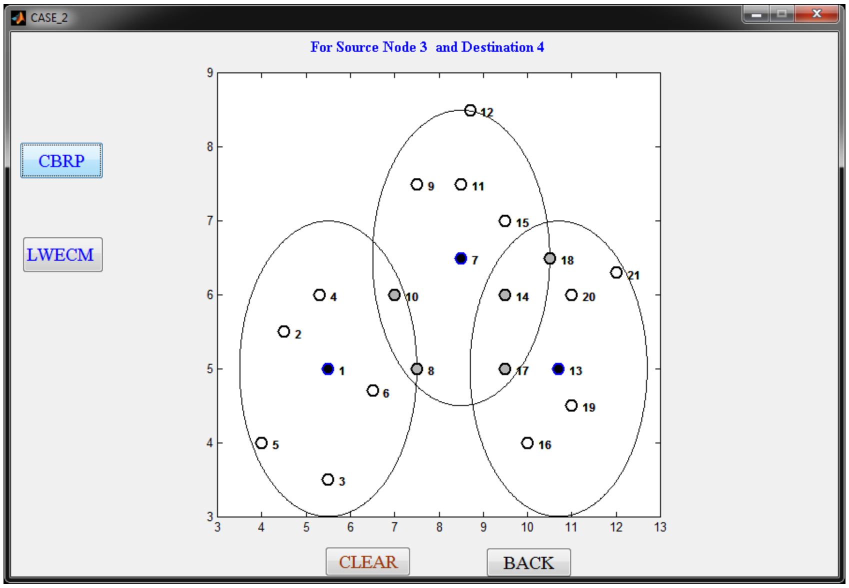

For Source Node 3 and Destination 4

Figure A.13: Screenshot of simulation screen of CBRP and LWECM for case II

Fig. A.14 shows the screenshot of simulation outcome of path construction in CBRP for case II, as source node 3 and destination 4

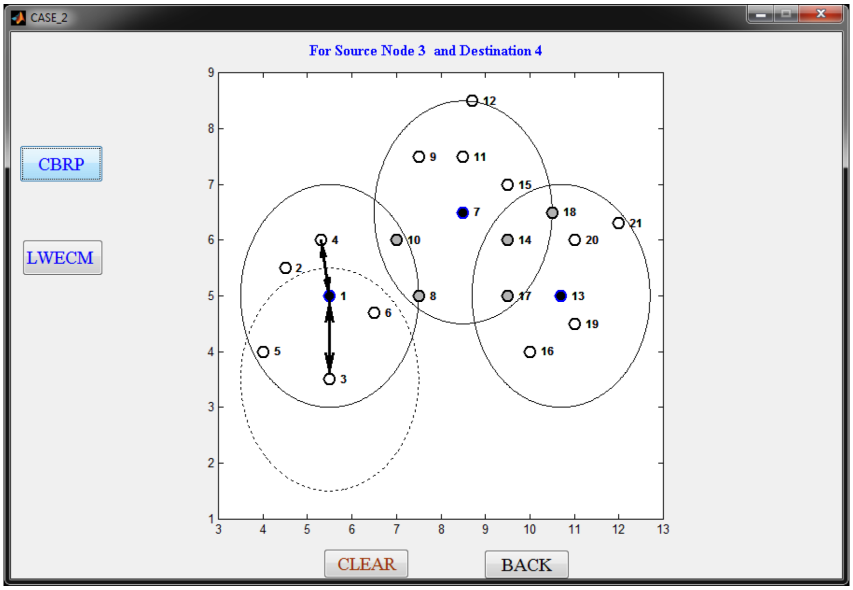

For Source Node 3 and Destination 4

Figure A.14: Screenshot of Simulation outcome of CBRP for case II

Fig. A.15 shows the screenshot of simulation outcome of path construction in LWECM for case II, as source node 3 and destination 4.

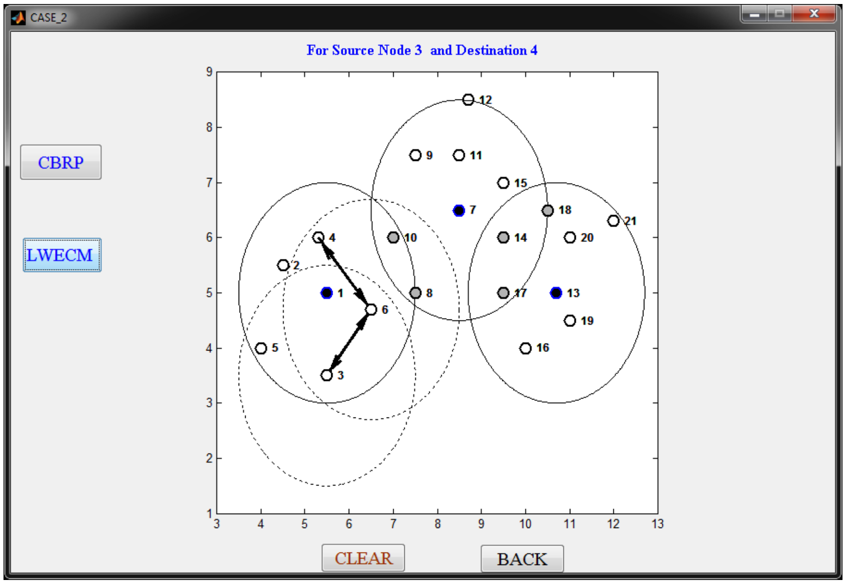


Figure A.15: Screenshot of Simulation outcome of LWECM for case II

Fig. A.16 shows the screenshot of the simulation screen of CBRP and LWECM for case III. Clicking on push buttons labeled as ‘CBRP’ and ‘LWECM’, triggers the simulations of corresponding protocol. Push buttons labeled as ‘CLEAR’ clears the simulation area and push button labeled as ‘BACK’ navigates to case screen.

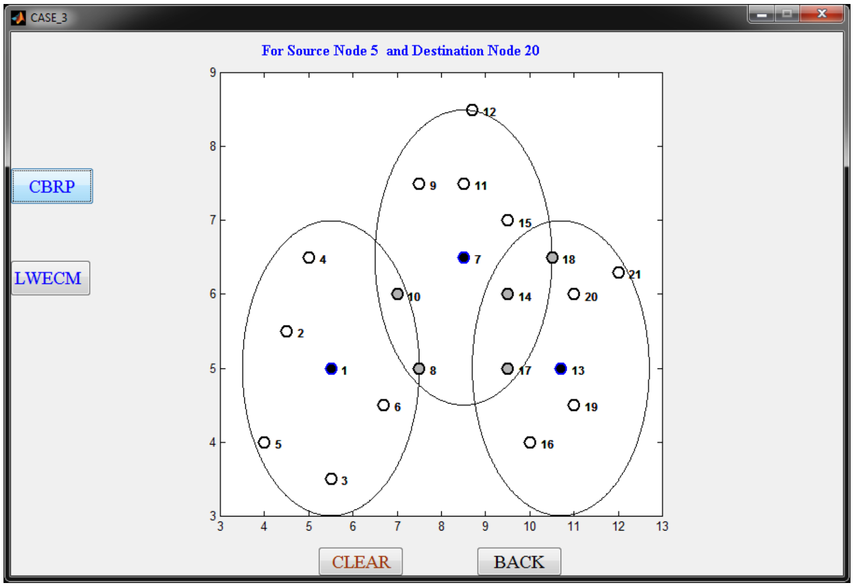


Figure A.16: Screenshot of simulation screen of CBRP and LWECM for case III

Fig. A.17 shows the screenshot of simulation outcome of path construction in CBRP for case III, as source node 5 and destination 20

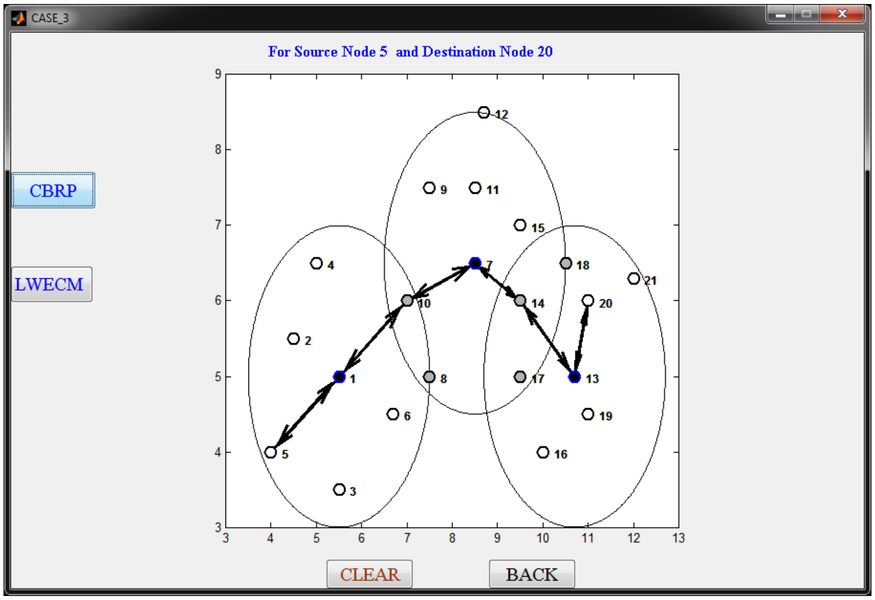


Figure A.17: Screenshot of Simulation outcome of CBRP for case III

Fig. A.18 shows the screenshot of simulation outcome of path construction in LWECM for case III, as source node 5 and destination 20.

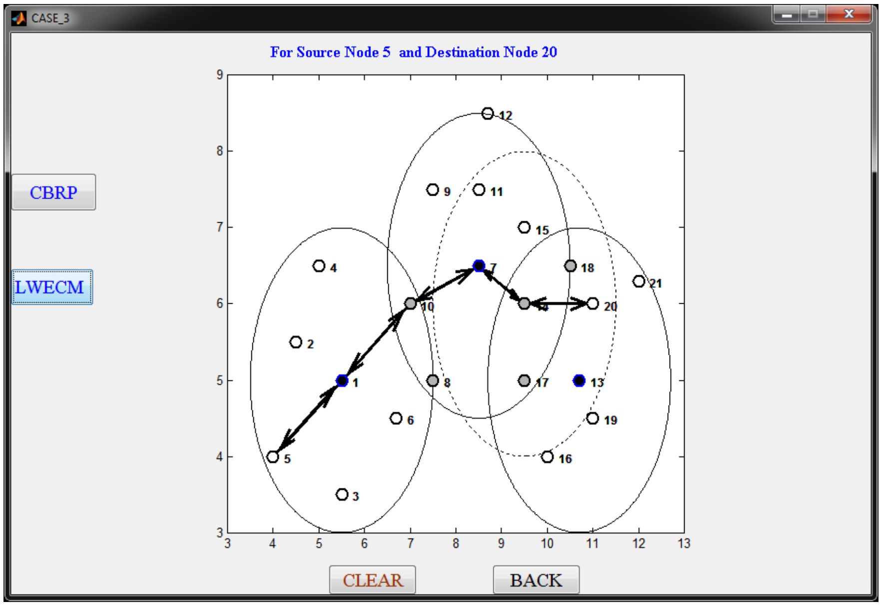


Figure A.18: Screenshot of Simulation outcome of LWECM for case III

Fig. A.19 shows the screenshot of the simulation screen of CBRP and LWECM for case IV. Clicking on push buttons labeled as ‘CBRP’ and ‘LWECM’, triggers the simulations of corresponding protocol. Push buttons labeled as ‘CLEAR’ clears the simulation area and push button labeled as ‘BACK’ navigates to case screen.

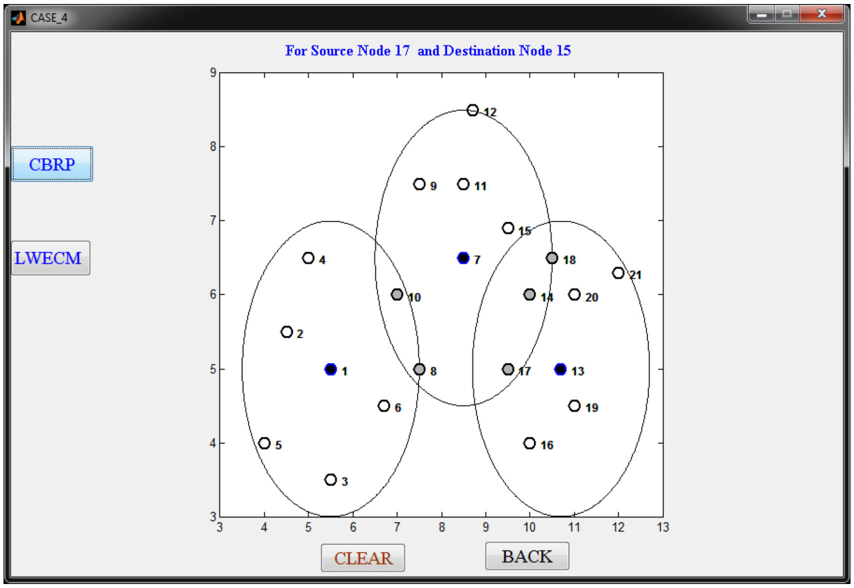


Figure A.19: Screenshot of simulation screen of CBRP and LWECM for case IV

Fig. A.20 shows the screenshot of simulation outcome of path construction in CBRP for case VI, as source node 17 and destination 15

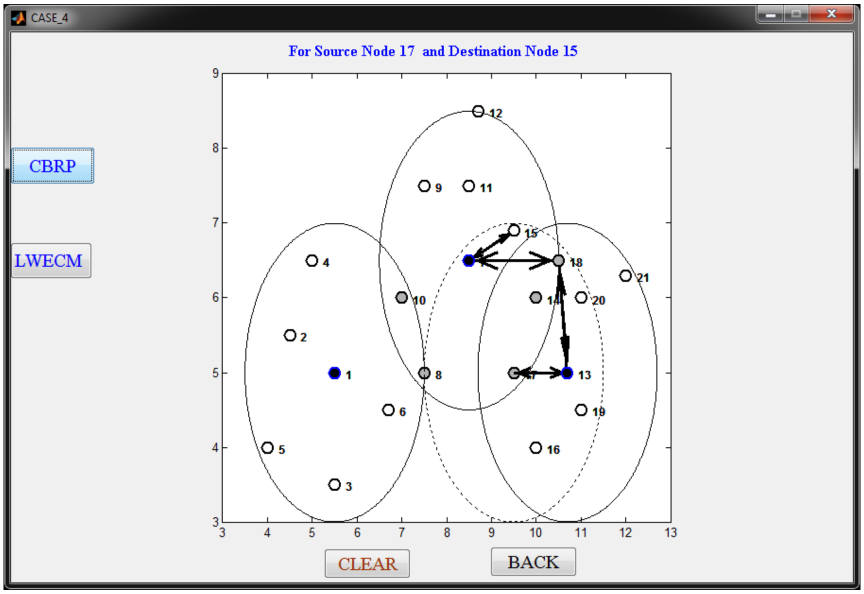


Figure A.20: Screenshot of Simulation outcome of CBRP for case IV

Fig. A.21 shows the screenshot of simulation outcome of path construction in LWECM for case VI, as source node 17 and destination 15.

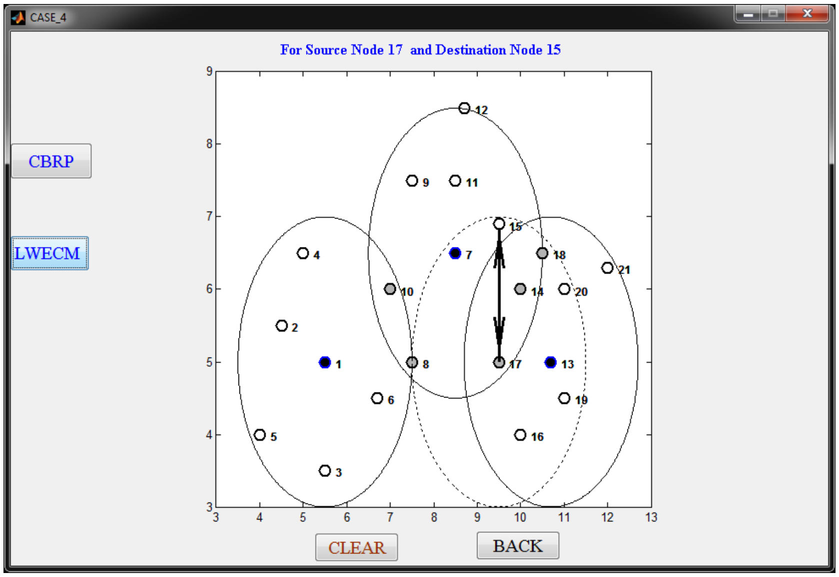

For Source Node 17 and Destination Node 15

Figure A.21: Screenshot of Simulation outcome of LWECM for case IV

Fig. A.22 shows the screenshot of the simulation screen of WCA and MLWCM. Clicking on push buttons labeled as ‘WCA’ and ‘MLWCM’, triggers the simulations for cluster formation of corresponding protocol . Push buttons labeled as ‘CLEAR’ clears the simulation area and push button labeled as ‘BACK’ navigates to home screen.

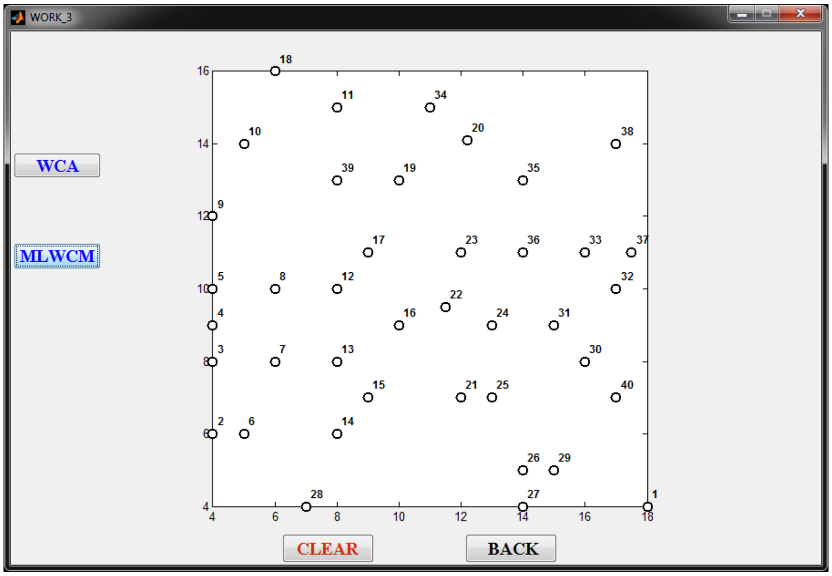


Figure A.22: Screenshot of simulation screen of WCA and MLWCM

Fig. A.23 shows the screenshot of simulation outcome of cluster formation in WCA in an arbitrary topology of 40 nodes.

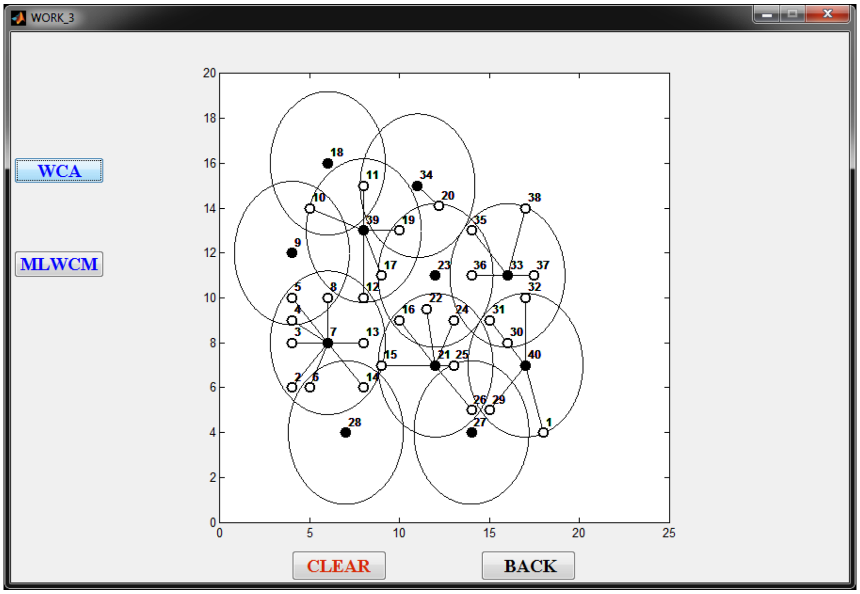


Figure A.23: Screenshot of Simulation outcome of cluster formation in WCA

Fig. A.24 shows the screenshot of simulation outcome of cluster formation in MLWCM in an arbitrary topology of 40 nodes.

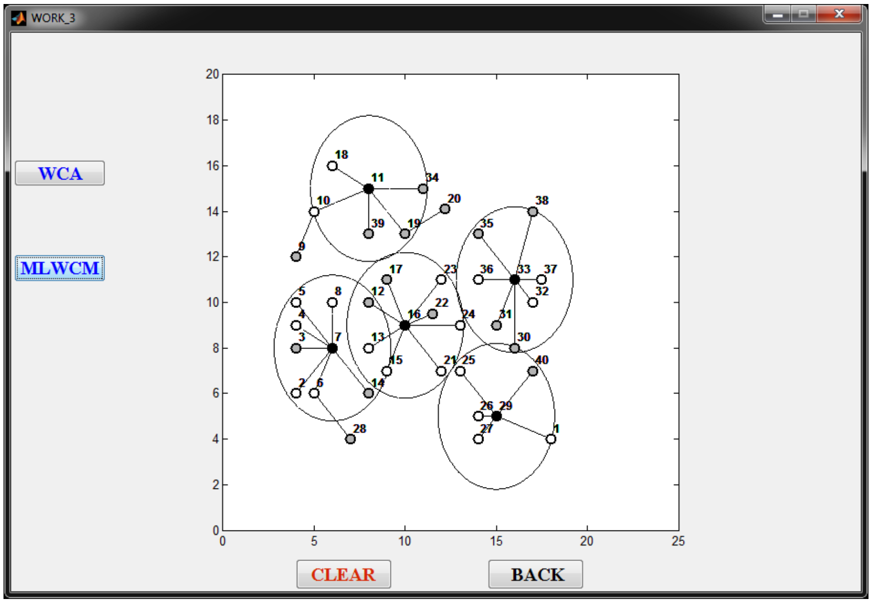
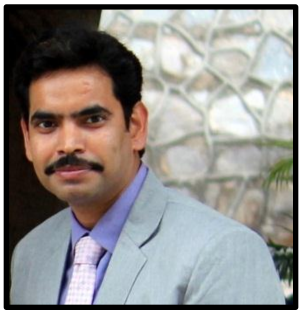


Figure A.24: Screenshot of Simulation outcome of cluster formation in MLWCM

BRIEF PROFILE OF RESEARCH SCHOLAR



Prashant Dixit did his M. Tech. in Information Technology from *J.C. Bose University of Science & Technology, YMCA*, Faridabad in 2012 and B. Tech. (Computer Science and Engineering) from *RGPV Bhopal* in 2008. Mr. Prashant has over 10 years of experience in teaching B. Tech and M. Tech courses. His areas of interest include Data Structure , Theory of Computation, Mobile Ad-hoc Networks, Algorithms . He has published 14 research papers in various journals and conferences of international and national fame. He is working as Assistant Professor in the department of Computer Science & Engineering at Faculty of Engineering and Technology , Manav Rachna International institute of Research and Studies, Faridabad, Haryana.

PUBLICATIONS

List of Papers in Journals

S. No	Title of the paper along with volume, Issue no., year of publication	Indexing	Publisher	Impact Factor	Refereed or Non-Refereed	Whether You paid any money or not for publication	Remarks
1	Prashant Dixit, Anuradha Pillai, Rahul Rishi" **QoS Enabled Improved Location Aided routing (QEILA**)" Recent Patents on Engineering ISSN: 2212-4047 (Online)ISSN: 1872-2121 (Print) Vol 13 2019	**SCOPUS**	Bentham Science Publishers	**0.82**	Yes	No	Published
2	Prashant Dixit, Anuradha Pillai, Rahul Rishi" **Location Information Based Destination Converging Routing Method (LIBDCR)**" Procedia Computer Science 132 (2018) 572–580 **ELSEVIE**R ISSN: 1877-0509	**SCOPUS**	ELSEVIER	**1.08**	YES	No	Published
3	Prashant Dixit, Anuradha Pillai, Rahul Rishi "**Back tracking with exclusion: A solution for local maxima problem in greedy location based packet forwarding for MANETs**" International Journal of Computer Sciences and EngineeringVol.6 , Issue.9 , pp.361-364, Sep-2018	**UGC Approved**	IJCSE	3.022	Yes	No	Published
4	Prashant Dixit, Anuradha Pillai, Rahul Rishi "**Middle in Forwarding Movement (MFM): An efficient greedy forwarding approach in location aided routing for MANET**" International Journal of Management, Technology And Engineering Vol 8, issue 9 Page 151-156 (2018)	**UGC Approved**	IJMTE	6.3	Yes	Paid for open access	Published

5	Prashant Dixit, Anuradha Pillai, Rahul Rishi” “**Analysis and comparison of various routing methods for Mobile adhoc Networks (MANETs)**” Advances in Computational Sciences and Technology ISSN 0973-6107 Volume 10, Number 9 (2017) pp. 2897-2909	**UGC Approved (OLDLIST)**	RIP	-	YES	Paid for open access	Published
6	Prashant Dixit, Anuradha Pillai, Rahul Rishi” **A Light Weight Efficient Cluster based routing Model for Mobile Ad-hoc Networks (LWECM)**” International Journal of Information Technology SSN: 2511-2104 (print version) ISSN: 2511-2112 **SPRINGER Journal No.: 41870**	**UGC Approved**	**SPRINGER**	-	YES	NO	Accepted In press

List of Papers in Conferences

	Title of the paper along with volume, Issue no., year of publication	Indexing	Publisher	Remarks
1	Prashant Dixit, Anuradha Pillai, Rahul Rishi” **Comparative Analysis of Location Aided Routing Protocols in Mobile Adhoc Networks**” Proceedings of the 12th INDIACom; INDIACom-2018; **IEEE** Conference ID: 428352018 5th International Conference on “Computing for Sustainable Global Development”, 14th - 16th March, 2018	**SCOPUS**	**IEEE**	Published
2	Prashant Dixit, Anuradha Pillai, Rahul Rishi “**Comaprision based study of various routing strategy for Mobile adhoc Networks (MANETs)** ” International Conference on Sustainable Development through Research in Engineering and Management (SDREM 2016)			Published
3	Prashant Dixit, Anuradha Pillai “**Study of Clustering Techniques in Mobile adhoc Network(MANETs)**” National Conference New Horizons in Technology for Sustainable Energy and Environment (NHTSEE **2017**).			Published

List of Communicated Paper

S. No	Title of the paper along with volume, Issue no., year of publication	Indexing	Publisher	Impact Factor	Refereed or Non-Refereed	Whether You paid any money or not for publication	Remarks
1	Prashant Dixit, Anuradha Pillai, Rahul Rishi" **Marking based Load balancing Weighted Clustering framework for Mobile ADHOC Network**" International Journal of Sensor Networks	**SCI Exp.**	Inderscience	0.878	YES	NO	Communicated