PERFORMANCE ANALYSIS OF SWARM BASED ROUTING PROTOCOLS FOR MANETs

A thesis submitted in partial fulfillment of the requirements

for the award of degree of

Master of Engineering

in

Computer Science and Engineering

Submitted By

KUSAKUMAR PUNNANA

(Roll No: 800832029)

Under The supervision of:

DR.A.K.VERMA

Assistant professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004

June 2010

Certificate

I hereby certify that the work which is being presented in thesis entitled " Performance Analysis of Swarm Based Routing Protocols for MANETs" , in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering Submitted in Computer Science and Engineering Department of Thapar University ,Patiala , is an authentic record of my own work carried out under supervision of Dr. Anil Kumar Verma and refers other researchers' works which are duly listed in reference section.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

Kusakumar p.

(Kusakumar punnana)

This is certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Anil Kr. Verma 05/07/10.

(Dr. Anil Kumar Verma)

Assistant professor

Computer Science and Engineering Department

Thapar University

Patiala.

Countersigned By

for 05/07/2010

(Dr. Rajesh Bhatia)

Head

Computer Science and Engineering Dept,

Thapar University,

Patiala.

RK Sharma 8/7/10

(Dr.R.K. Sharma)

Dean (Academic Affairs)

Thapar University

Patiala.

Acknowledgment

No volume of words is enough to express my gratitude towards my guide, Dr. Anil Kumar Verma, Assistant Professor and Computer Science and Engineering Department, Thapar University, who have been very concerned and have aided for all the material essential for the preparation of this thesis report. They have helped me to explore this vast topic in an organized manner and provided me with all the ideas on how to work towards a research oriented venture.

I am also thankful to Dr. Rajesh Bhatia, Head of Department, CSED, and Dr. Inderveer Channa, P.G. Coordinator, for the motivation and inspiration that triggered me for the thesis work.

I would also like to thank the staff members and my colleagues who were always there in the need of the hour and provided with all the help and facilities, which I required, for the completion of my thesis.

I am also grateful to my friends and fellow students, who give me their support in and outside the laboratory. They are so many and from so different countries that I cannot mention all of them. However, they know that I will always appreciate their friendship and help.

Most importantly, I would like to thank my Parents and the Almighty for showing me the right direction out of the blue, to help me stay calm in the oddest of the times and keep moving even at times when there was no hope.

(Kusakumar punnana)

800832029

Abstract

A Mobile Ad-Hoc Network (MANET) is a collection of wireless mobile nodes forming a temporary network without using centralized access points, infrastructure, or centralized administration. To establish a data transmission between two nodes, typically multiple hops are required due to the limited transmission range. Routing is one major task in MANETS. We present ant routing is a new scheme for routing inspired by the behavior of real ants. Real ants are able to find the shortest path to a food source by following the trail of a chemical substance called pheromone deposited by other ants. In ant routing, the ants (control packets) collect information about the network conditions and are used to update and maintain the routing tables. Owing to the striking similarities between self-organizing behavior of ant colonies and self-organization in peer-to-peer networks, intelligence of real ants has been largely exploited by researchers to coordinate population of artificial agents that can collaborate to solve routing problems in highly dynamic mobile ad hoc networks. With intent to compare ant based algorithm for mobile ad hoc networks (MANETs) are delineated in this thesis report. In this, together with the increasing popularity of ad-hoc wireless networks, has given us the idea to adapt ant routing for such mobile networks and determine whether it is suitable or not. A version of this ant routing protocol has been implemented to work within the network simulator NS2. Then, a performance comparison has been done with two well-known ad-hoc routing protocols, i.e. ARA and AntHocNet. Results show that the overhead due to route maintenance is high, so the performance degrades and is inferior to ARA and AntHocNet. However, more simulations in another environment should be done before rejecting this scheme for ad-hoc wireless networks. Each of the routing schemes and algorithms has the common objective of trying to get better throughput and to extend the lifetime of the mobile network.

This thesis report first describes brief overview of existed routing protocols in MANETs and their classification. Finally discusses existing Ant Based Algorithms in MANETs, and comparison has been made between two routing protocols, ARA and AntHocNet on the basis of throughput, packet delay and routing overhead.

Keywords: Wireless Mobile Networks, Ant Colony Optimization (ACO), ARA, AntHocNet.

Table of Contents

List of Figures

Figures:

List of Tables

Tables:

LIST OF ABBREVIATIONS

ACK	Acknowledgement
MANET	Mobile ad-hoc Network
MANETs	Mobile ad- hoc Networks
MAC	Medium Access Control Layer
IETF	Internet Engineering Task Fork
PAN	Personnel Area Network
WLAN	Wireless Local Area Network
BAN	Body Area Network
SI	Swarm Intelligence
GSR	Global State Routing Protocol
DSDV	Destination Sequenced Distance Vector
WRP	Wireless Routing Protocol
HSR	Hierarchical State Routing Protocol
FSR	Fisheye State Routing Protocol
CGSR	Cluster head Gateway Switch Routing
AODV	Ad hoc On Demand Distance Vector Routing
DSR	Dynamic Source Routing
ABR	Associativity Based Routing
TORA	Temporally Ordered Routing Algorithm
ZRP	Zone Routing Protocol

ZHLS	Zone-based Hierarchical Link State
CBRP	Cluster Based Routing Protocol
ALARM	Adaptive Location Aided Routing-Mines
DREAM	Distance Routing Effect Algorithm For Mobility
GPSAL	GPS Ant- Like Routing Algorithm
ARA	Ant colony Routing Algorithm
ACO	Ant Colony Optimization
TSP	Travelling Salesman Problem
PERA	Probabilistic Emergent Routing Algorithm
ADRA	Ant based Distributed Routing Algorithm
ARAAI	Ant Routing Algorithm based on Adaptive improvement
MAARA	Multi Agent Ant based Routing Algorithm
FANT	Forward Ant
BANT	Backward Ant
RERR	Route Error messages
Ant AODV	Ant-ad-hoc on demand distance vector routing
HOPNET	Hybrid ant colony optimization routing algorithm for mobile ad-hoc network
PACONET	Improved ant colony optimization routing algorithm for mobile adhoc network
FC	Fedora Core

FTP	File Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
UDP	User Datagram Protocol
TCP	Transmission Control Protocol
NS	Network Simulator
NAM	Network Animator
GUI	Graphical User Interface
TCL/TK	Tool Command Language/ Tee-kay
OTCL	Object-Oriented Tool Command Language

CHAPTER 1
INTRODUCTION

1.1. Background and Motivation

The history of wireless networks started in the 1970s and the interest has been growing ever since. During the last decade, and especially at its end, the interest has almost exploded probably because of the fast growing Internet. The tremendous growth of personal computers and the handy usage of mobile computers necessitate the need to share information between computers. At present, this sharing of information is difficult, as the users need to perform administrative tasks and set up static, bi-directional links between the computers.

This motivates the construction of temporary networks with no wires, no communication Infrastructure and no administrative intervention required. Such interconnections between mobile computers are called an Ad-hoc Wireless Network. Ad-hoc wireless networks are increasing in popularity, due to the spread of laptops, sensor devices, PDAs and other mobile electronic devices. These devices will eventually need to communicate with each other. In some cases, without an adequate infrastructure to rely on. That's why we need routing protocols that can work without any central gateway to connect with.

1.2. State of the Art

Typically, a wireless mobile node has the capability of sensing, computing, communication. The components of mobile node are integrated on a single or multiple boards, and packaged in a few cubic inches. With state-of-the-art, low-power circuit and networking technologies, a mobile node powered by 2 AA batteries can last for up to three years. A MANETs usually consists of tens to thousands of such nodes that communicate through wireless channels for information sharing and cooperative processing. In a typical scenario, users can retrieve information of interest from a MANETs by injecting queries and gathering results from the base stations or sink nodes, which behave as an interface between users and the network. In this way, MANETs can be considered as a distributed database. The mobile networks will ultimately be connected to the Internet, through which global information sharing becomes feasible.

1.3. Thesis Outline

We have organized the thesis into 7 chapters which include Introduction; Background Information; Literature Review; Problem Statement; Installation, Simulation and Design; Results, Performance Evaluation and Analysis and finally Conclusion and Future Scope.

In chapter 1, we describe Wireless Mobile Ad-hoc Network in general in terms of motivation and finally the whole thesis outline. In chapter 2, we discuss the background information relating to MANETs and its routing. In this chapter, we will navigate deeply in Mobile Ad hoc Networks (MANETs) and give detailed overview about many different points in MANET, such as, classification, some different definitions, applications, special features .

In chapter 3, we study the state of the art of various Existing routing protocols in MANETs. ARA and AntHocNet protocol in detail has been discussed covering the description of protocol modes and working, structure of various packets being transferred, procedures followed by the nodes in the particular modes and analyses swarm intelligence based routing with routing protocols in MANET. At the same time, swarm intelligence has been used to solve optimization problems applied to data networks. Routing is one such optimization problem where swarm intelligence has been applied. Several routing protocols take advantage of that, i.e. AntNet AODV, ARA and AntHocNet etc. In chapter 4 we discuss the problem statement and tasks. In chapter 5 we explain the installation of tools and the simulation environment. In chapter 6 we describes the results, evaluates the performance, and analysis and finally in chapter 7 we summarizes the conclusions drawn in the thesis along with future research directions.

CHEPTER 2

BACKGROUND AND INFORMATION

2.1. MANETs [1][2]

Mobile Ad-hoc Networks (MANETs) are communication networks in which all nodes are mobile and communicate with each other via wireless connections. There is no fixed infrastructure. All nodes are equal and there is no centralized control or overview. There are no designated routers: all nodes can serve as routers for each other and data packets are forwarded from node to node in a multi-hop fashion.

MANETs can be applied in situations where no fixed network infrastructure is available, such as military activities in enemy territory and disaster recovery operations. Routing is the task of directing data flows from sources to destinations maximizing network performance. This is particularly difficult in MANETs. Due to the mobility of the nodes, the topology of the network changes constantly and paths which were initially efficient can quickly become inefficient or even infeasible. This means that routing information should be updated more regularly than for instance in wired networks. However, this can be a problem in MANETs, since the bandwidth of the network is limited by the fact that the wireless medium is shared: nodes can only send or receive data if no other node is sending in their immediate neighborhood. The access to the shared channel is controlled by protocols at the Medium Access Control layer (MAC), such as ANSI/IEEE 802.11 DCF3 (the most commonly used in MANETs), which create extra overhead, lowering the effective available bandwidth.

Many MANET routing algorithms have been proposed .These algorithms deal with the dynamic aspects of MANETs in their own way, using reactive or proactive behavior or a combination of both. Reactive behavior means that an algorithm only gathers routing information in response to an event, usually one which triggers the need for new routes, such as the start of a data session or the failure of an existing route. Proactive behavior means that the algorithm also gathers information at other times, so that routing information is readily available when the event happens.

We quote the definition of a mobile ad-hoc network from the charter of the corresponding Internet Engineering Task Force (IETF) “A ‘mobile ad-hoc network’ (MANET) is an autonomous system of mobile routers (and associated hosts) connected by wireless links- the union of which forms an arbitrary graph. The routers are free to move randomly and organize themselves arbitrarily; thus, the network's wireless topology may change rapidly and unpredictably. Such a network may operate in a standalone fashion, or may be connected to the larger Internet".

MANET is a kind of wireless ad-hoc network and it is a self-configuring network of mobile routers (and associated hosts) connected by wireless links – the union of which forms an arbitrary topology. The routers, the participating nodes act as router, are free to move randomly and manage themselves arbitrarily; thus, the network's wireless topology may change rapidly and unpredictably. Such a network may operate in a standalone fashion, or may be connected to the larger Internet [5].

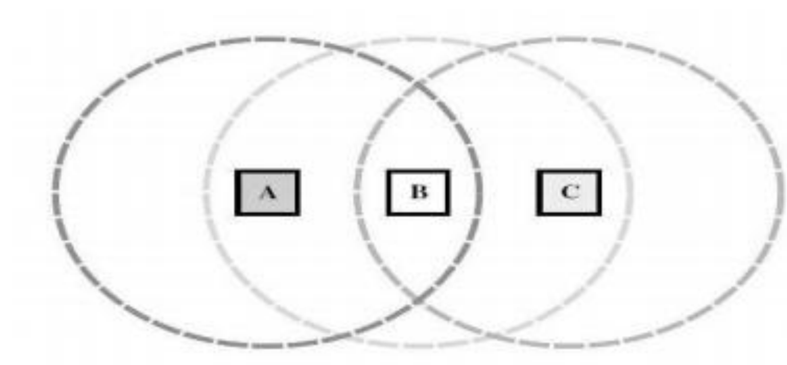

Figure 1.1 Example of a simple ad-hoc network with three participating nodes

Mobile ad hoc network is a collection of independent mobile nodes that can communicate to each other via radio waves. The mobile nodes can directly communicate to those nodes that are in radio range of each other, whereas others nodes need the help of intermediate nodes to route their packets. These networks are fully distributed, and can work at any place without the aid of any infrastructure. This property makes these networks highly robust.

In Figure 1.1 nodes A and C must discover the route through B in order to communicate. The circles indicate the nominal range of each node‘s radio transceiver. Nodes A and C are not in direct transmission range of each other, since A‘s circle does not cover C [5].

The ultimate goal of MANETs is to provide secure routing of data resources to mobile user's at anytime and from anywhere. In conjunction with the existing routing protocols providing security for MANETs give rise to significant challenges and performance opportunities [6].

2.2. MANET Characteristics

MANETs are new paradigm of networks, offering unrestricted mobility without any underlying infrastructure. Basically, ad-hoc network is a collection of nodes communicating with each other by forming a multi-hop network. Following are the characteristics of a MANETs [1, 2].

The characteristics of these networks are summarized as follows:

Dynamic Topologies

Nodes are free to move arbitrarily. The network topology may change randomly and have no restriction on their distance from other nodes. As a result of this random movement, the whole topology is changing in an unpredictable manner, which in turn gives rise to both directional as well as unidirectional links between the nodes.

Energy Constrained Operation

Almost all the nodes in an ad-hoc network rely on batteries or other exhaustive means for their energy. The battery depletes due to extra work performed by the node in order to survive the network. Therefore, energy conservation is an important design optimization criterion.

Bandwidth Constraint

Wireless links have significantly lower capacity [3] than infrastructures networks. Throughput of wireless communication is much less because of the effect of the multiple access, fading, noise, interference conditions. As a result of this, congestion becomes a bottleneck in bandwidth utilization.

Limited Physical Security

MANETs are generally more prone to physical security threats than wireless networks because the ad hoc network is a distributed system and all the security threats relevant to such a system are pretty much present, as a result, there is an increased possibility of eavesdropping, spoofing, masquerading[4], and denial-of-service type attacks.

Scalability

Networks may be large, normally more than 10 nodes and reaching 1000 nodes in a sensor network. Thus, routing protocols should be able to scale to this amount. A number of algorithms have been proposed, and can be categorized as either proactive or reactive protocols.

2.3 Advantages of MANETs

The following are the advantages of MANET:

- They provide access to information and services regardless of geographic position.
- These networks can be set up at any place and time.

2.4 Disadvantages of MANETs

Some of the disadvantages of MANETs are as follows:

- Limited resources and physical security.
- Intrinsic mutual trust vulnerable to attacks.
- Lack of authorization facilities.
- Volatile network topology makes it hard to detect malicious nodes.
- Security protocols for wired networks cannot work for ad hoc networks.

2.5. Applications of wireless mobile networks

Some of the applications of MANETs are as follows:

- Military or police exercises.
- Disaster relief operations.
- Urgent Business meetings.
- Sensor networks.
- Students on campus.
- Free Internet connection sharing.
- Conferences.
- Vehicle communications.
- Personnel Area Network (PAN) for communication of several portable devices.

2.6. Routing

Routing is the act of moving information from a source to a destination in an internetwork. A least one intermediate node within the internetwork is encountered during the transfer of information. Basically two activities are involved in this concept: determining optimal routing paths and transferring the packets through an internetwork .The transferring of packets through an internetwork is called as packet switching which is straight forward, and the path determination could be very complex. Routing protocols use several metrics as a standard measurement to calculate the best path for routing the packets to its destination that could be number of hops, which are used by the routing algorithm to determine the optimal path for the packet to its destination. The process of path determination is that, routing algorithms find out and maintain routing tables, which contain the total route information for the packet. The information of route varies from one routing algorithm to another. The routing tables are filled with entries in the routing table are ip-address prefix and the next hop. Destination/next hop associations of routing table tell the router that a particular destination can be reached optimally by sending the packet to a router representing the "next hop" on its way to the final destination and ip-address prefix specifies a set of destinations for which the routing entry is valid.

Routing is mainly classified into static routing and dynamic routing. Static routing refers to the routing strategy being stated manually or statically, in the router. Static routing maintains a routing table usually written by a networks administrator. The routing table doesn‘t depend on the state of the network status, i.e., whether the destination is active or not [7]. Dynamic routing refers to the routing strategy that is being learnt by an interior or exterior routing protocol. This routing primarily depends on the state of the network i.e., the routing table is affected by the activeness of the destination.

2.6.1. Routing in MANETs

A routing protocol [8] is the mechanism by which user traffic is directed and transported through the network from the source node to the destination node. Objectives include maximizing network performance from the application point of view - application requirements- while minimizing the cost of network itself in accordance with its capacity. The application requirements are hop count, delay, throughput, loss rate, stability, jitter, cost; and the network capacity is a function of available resources that reside at each node and number of nodes in the network as well as its density, frequency of end-to-end connection (i.e. number of communication), frequency of topology changes (mobility rate). The four core basic routing functionality for mobile ad hoc networks are:

- **Path generation**: This generates paths according to the assembled and distributed state information of the network and of the application; assembling and distributing network and user traffic state information.
- **Path selection**: This selects appropriate paths based on network and application state information.
- **Data Forwarding**: This forwards user traffic along the select route forwarding user traffic along the selected route.
- **Path Maintenance:** maintaining of the selected route.

Due to its characteristics, other desirable features of ad hoc routing protocol include- fast route establishment, multiple routes selection, energy/bandwidth efficiency and fast adaptability to link changes. Almost all routing systems respond in some way to the changes in network and user traffic state. However, routing systems vary widely in the types of state changes to which they respond and the speed of their response.

2.6.2. Properties of Ad-Hoc Routing protocols

The properties that are desirable in Ad-Hoc Routing protocols are [9]:

i). Distributed operation: The protocol should be distributed. It should not be dependent on a centralized controlling node. This is the case even for stationary leave the network very easily and because of mobility the network can be partitioned

ii). Loop free: To improve the overall performance, the routing protocol should assurance that the routes supplied are loop free. This avoids any misuse of bandwidth or CPU consumption.

iii). Demand based operation: To minimize the control overhead in the network and thus not misuse the network resources the protocol should be reactive. This means that the protocol should react only when needed and should not periodically broadcast control information.

iv). Unidirectional link support: The radio environment can cause the formation of unidirectional links. Utilization of these links and not only the bi-directional links improves the routing protocol performance.

v). Security: The radio environment is especially vulnerable to impersonation attacks so to ensure the wanted behavior of the routing protocol we need some sort of security measures. Authentication and encryption is the way to go and problem here lies within distributing the keys among the nodes in the ad-hoc network.

vi). Power conservation: The nodes in the ad-hoc network can be laptops and thin clients such as PDA's that are limited in battery power and therefore uses some standby mode to save the power. It is therefore very important that the routing protocol has support for these sleep modes.

vii). Multiple routes: To reduce the number of reactions to topological changes and congestion multiple routes can be used. If one route becomes invalid, it is possible that another stored route could still be valid and thus saving the routing protocol from initiating another route discovery procedure.

viii). Quality of Service Support: Some sort of Quality of service is necessary to incorporate into the routing protocol. This helps to find what these networks will be used for. It could be for instance real time traffic support.

2.6.3. Problems in routing with Mobile Ad hoc Networks

- **Asymmetric links:** Most of the wired networks rely on the symmetric links which are always fixed. But this is not a case with ad-hoc networks as the nodes are mobile and constantly changing their position within network
- **Routing Overhead:** In wireless ad hoc networks, nodes often change their location within network. So, some stale routes are generated in the routing table which leads to unnecessary routing overhead.
- **Interference:** This is the major problem with mobile ad-hoc networks as links come and go depending on the transmission characteristics, one transmission might interfere with another one and node might overhear transmissions of other nodes and can corrupt the total transmission.
- **Dynamic Topology:** Since the topology is not constant; so the mobile node might move or medium characteristics might change. In ad-hoc networks, routing tables must somehow reflect these changes in topology and routing algorithms have to be adapted. For example in a fixed network routing table updating takes place for every 30sec. This updating frequency might be very low for ad-hoc networks.

2.6.4. Routing Objectives

Some Ad-hoc network applications only require the successful delivery of messages between a source and a destination. However, there are applications that need even more assurance. These are the real-time requirements of the message delivery, and in parallel, the maximization of network lifetime.

- **Non-real time delivery:** The assurance of message delivery is indispensable for all routing protocols. It means that the protocol should always find the route between the communicating nodes, if it really exists. This correctness property can be proven in a formal way, while the average-case performance can be evaluated by measuring the message delivery ratio.

- **Real-time delivery:** Some applications require that a message must be delivered within a specified time, otherwise the message becomes useless or its information content is decreasing after the time bound. Therefore, the main objective of these protocols is to completely control the network delay. The average-case performance of these protocols can be evaluated by measuring the message delivery ratio with time constraints.
- **Network lifetime:** This protocol objective is crucial for those networks, where the application must run on sensor nodes as long as possible. The protocols aiming this concern try to balance the energy consumption equally among nodes considering their residual energy levels. However, the metric used to determine the network lifetime is also application dependent. Most protocols assume that every node is equally important and they use the time until the first node dies as a metric, or the average energy consumption of the nodes as another metric. If nodes are not equally important, then the time until the last or high-priority nodes die can be a reasonable metric.

2.6.5. Characteristics of Routing protocols

A good routing protocol should be:

- Application specific.
- Data centric.
- Capable of a aggregating data.
- Capable of optimizing energy consumption.

2.7. Classification of routing Protocols in MANETs

Classification of routing protocols in MANET's can be done in many ways, but most of these are done depending on routing strategy and network structure. In general Routing states can be divided into three categories - Static, Quasi Static and Dynamic in MANETs. Further, each of the three basic routing functions may be implemented in three ways- Centralized, Decentralized and Distributed. The routing protocols can be mainly categorized as: Flat routing, Hierarchical routing and Location aware routing on network structure. In flat-based routing, all nodes play the same role. In hierarchical –based routing, however, nodes will play different role in network. In location aware-based routing, nodes positions are exploited to route data in the network. This chapter describes several concepts concerning the operation of that kind of routing protocols. According to the routing strategy the routing protocols can be categorized as Table-driven and source initiated, while depending on the network structure these are classified as flat routing, hierarchical routing and location aware routing. Both the Table-driven and source initiated protocols come under the Flat routing.

2.7.1. Flat Routing (Data centric Routing)

There are two schemes in flat routing, namely, table-driven (or proactive) routing protocols and on-demand (or reactive) routing protocols [10, 11]. One thing is general for both protocol classes is that every node participating in routing play an equal role. They have further been classified after their design principles; proactive routing is mostly based on LS (link-state) while on-demand routing is based on DV (distance-vector).

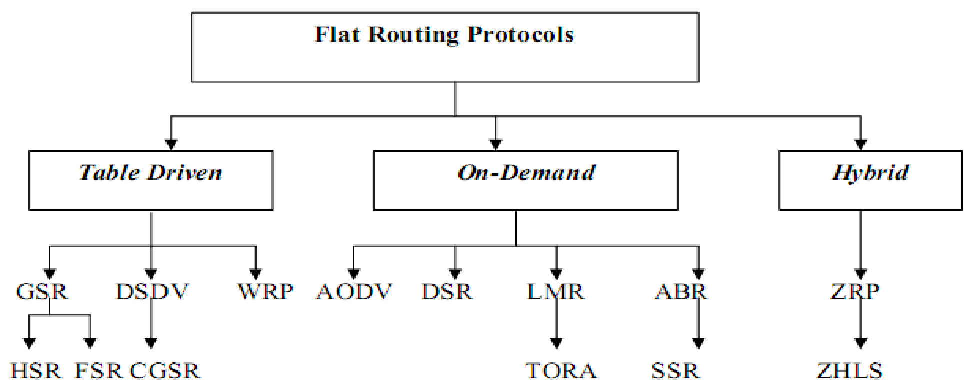

Figure 2.1. Classification of Routing Protocols in MANETs

Figure.2.1. provides a brief outline of the different flat-routing protocols proposed for MANETs.

I. Table-Driven Routing (Global/Proactive protocols)

In proactive routing protocols, the routes to all the destinations (or parts of the network) are determined at the start up, and maintained by using a periodic route update process. In proactive routing protocol each node maintains the information about the other nodes in the tables. Though the number of tables used by the different protocols differs. The various proactive routing protocols differ in the way in which they update the routing information in the tables.

The data broadcast by each node will contain its new sequence number and the following information for each new route:

– The destination address

– The number of hops required to reach the destination and

– The new sequence number, originally Stamped by the destination.

Proactive MANET protocols are also called as table-driven protocols and will actively determine the layout of the network. Through a regular exchange of network topology packets between the nodes of the network, at every single node an absolute picture of the network is maintained. There is hence minimal delay in determining the route to be taken. This is especially important for time-critical traffic [7]. When the routing information becomes worthless quickly, there are many short-lived routes that are being determined and not used before they turn invalid. Therefore, another drawback resulting from the increased mobility is the amount of traffic overhead generated when evaluating these unnecessary routes. This is especially altered when the network size increases. The portion of the total control traffic that consists of actual practical data is further decreased. Lastly, if the nodes transmit infrequently, most of the routing information is considered redundant. The nodes, however, continue to expend energy by continually updating these unused entries in their routing tables as mentioned, energy conservation is very important in a MANET system design. Therefore, this excessive expenditure of energy is not desired. Thus, proactive MANET protocols work best in networks that have low node mobility or where the nodes transmit data frequently. This type of protocols maintains fresh lists of destinations and their routes by periodically distributing routing tables throughout the network. The main disadvantages of such algorithms are:

- Respective amount of data for maintenance.
- Slow reaction on restructuring and failures.

Examples of Proactive MANETs Protocols are:

- Optimized Link State Routing (OLSR)
- Fish-eye State Routing (FSR)
- Destination-Sequenced Distance Vector (DSDV)
- Cluster-head Gateway Switch Routing Protocol (CGSR), etc...

II. On-Demand Routing (Reactive Protocols)

In reactive protocols, routes are determined when they are required by the source using a route discovery process. These protocols were designed to reduce the overhead encountered in proactive protocols by maintaining information for active routes only. This means that the routes are determined and maintained for the nodes that are required to send data to a particular destination. Route discovery usually occurs by flooding route request packets through the network. When a node with a route to the destination (or the destination itself) is reached a route reply is sent back to the source node using link reversal if the route request has traveled through the bi-directional links or by piggy-backing the route in a route reply packet via flooding. Therefore, the route discovery overhead (in the worst case scenario) will grow by O(N+M) when link reversal is possible and O(2N) for unidirectional links (where, N represents the total number of nodes and M represents the total number of nodes in the localized region).

Reactive protocols can be classified into two categories:

i. Source routing, and

ii. Hop-by-Hop routing

In Source routed on-demand protocols, each data packets carry the complete source to destination address. Therefore, each intermediate node forwards these packets according to the information kept in the header of each packet. This means that the intermediate nodes do not need to maintain up-to-date routing information for each active route in order to forward the packet towards the destination. Furthermore, nodes do not need to maintain neighbor's connectivity through periodic beaconing messages. The major drawback with source routing protocols is that in large networks they do not perform well.

In hop-by-hop routing (also known as point-to-point routing), each data packet only carries the destination address and the next hop address. Therefore, each intermediate node in the path to the destination uses its routing table to forward each data packet towards the destination. The advantage of this strategy is that routes are adaptable to the dynamically changing environment of MANETs, since each node can update its routing table when they receiver fresher topology information and hence forward the data packets over fresher and better routes. The disadvantage of this strategy is that each intermediate node must store and maintain routing information for each active route and each node may require being aware of their surrounding neighbors through the use of beaconing messages. This type of protocols finds a route on demand by flooding the network with Route Request packets. The main disadvantages of such algorithms are -

- High latency time in route finding.
- Excessive flooding can lead to network clogging.

The different types of On Demand driven protocols are:

- Ad hoc On Demand Distance Vector (AODV)
- Dynamic Source routing protocol (DSR)
- Temporally ordered routing algorithm (TORA)
- Ant Based routing (ABR)
- Signal Stability-Based Adaptive Routing (SSA)
- Location-Aided Routing Protocol (LAR)

III. Hybrid Protocols

Hybrid routing protocols combine the basic properties of the two classes of flat routing protocols into one. That is, they are both reactive and proactive in nature. Each group has a number of different routing strategies, which employ a flat or a hierarchical routing structure. Hybrid routing protocols are new generation of protocols, which are both proactive and reactive in nature. These protocols are designed to increase scalability by allowing nodes with close proximity to work together to form some sort of a backbone to reduce the route discovery overheads. This is mostly achieved by proactively maintaining routes to nearby nodes and determining routes to far away nodes using a route discovery strategy. Most hybrid protocols proposed to date are zone-based, which means that the network is partitioned or seen as a number of zones by each node. Other nodes group into trees or clusters. This type of protocols combines the advantages of proactive and of reactive routing. The routing is initially established with some proactively prospected routes and then serves the demand from additionally activated nodes through reactive flooding. The choice for one or the other method requires predetermination for typical cases. The main disadvantages of such algorithms are

- Advantage depends on amount of nodes activated.
- Reaction to traffic demand depends on gradient of traffic volume.

2.7.2. Hierarchical Routing protocols

The Hierarchical routing protocols are used when the network size increases and the flat routing becomes in feasible due to increase in processing overhead. In this approach, the MANET is partitioned into different groups and the nodes are assigned different functions within and outside the group. With this type of protocols the choice of proactive and of reactive routing depends on the hierarchic level where a node resides. The routing is initially established with some proactively prospected routes and then serves the demand from additionally activated nodes through reactive flooding on the lower levels. The choice for one or the other method requires proper attributation for respective levels. The main disadvantages of such algorithms are

- Advantage depends on depth of nesting and addressing scheme.
- Reaction to traffic demand depends on meshing parameters.

Some of the routing protocols are: Cluster Based Routing Protocol (CBRP), Cluster head Gateway Switch Routing (CGSR), Fisheye State Routing protocol (FSR), Global State Routing protocol (GSR), Zone Routing Protocol (ZRP), Hierarchical State Routing (HSR) etc.

2.7.3. Geographical routing protocols

The Geographical routing protocols imply that the hosts participating in the routing process should be aware of their geographic positions. This type of protocols acknowledges the influence of physical distances and distribution of nodes to areas as significant to network performance. The main disadvantages of such algorithms are

- Efficiency depends on balancing the geographic distribution versus occurrence of traffic.
- Any dependence of performance with traffic load thwarting the negligence of distance may occur in overload.

Some of these routing protocols are: Adaptive Location Aided Routing – Mines (ALARM), Distance Routing Effect Algorithm for Mobility (DREAM), GPS Ant-Like Routing Algorithm (GPSAL) etc.

2.7.4. Location-based protocols

In most cases location information is needed in order to calculate the distance between two particular nodes so that energy consumption can be estimated. Generally two techniques are used to find location, one is to find the coordinate of the neighboring node and other is to use GPS (Global Positioning System). Since, there is no addressing scheme for Mobile networks like IP-addresses and they are spatially deployed on a region, location information can be utilized in routing data in an energy efficient way.

CHAPTER 3
LITERATURE REVIEW

3.1. SWARM INTELLIGENCE

Swarm intelligence [12, 13] appears in biological swarms of some social insect species such as ants. Through simple interaction of autonomous swarm members, the group behavior gives rise to complex and intelligent behavior. Since 1999, there is a great interest in applying swarm intelligence to solve hard static and dynamic optimization problems. These problems are solved using cooperative agents that communicate with each other modifying their environment, like ant colonies or others insects do. That is why these agents are commonly called ants. SI is defined as 'the emergent collective intelligence of groups of simple agents'. This field is inspired by the surprising capabilities of collective social insects. SI is used to refer to systems whose design is inspired by models of social insect behavior.

Key characteristics of these models are:

- Large numbers of simple agents.
- Agents may communicate with each other directly.
- Agents may communicate indirectly by affecting their environment, a process known as stigmergy.
- Intelligence contained in the networks and communications between agents.
- Local behavior of agents causes some emergent global behavior.

Due to the use of mobile agents and stigmergy, swarm intelligence boats a number of advantages.

- ➢ **Scalability**: population of the agents can be adapted according to the problem size.
- ➢ **Fault tolerance**: swarm intelligence do not rely on a centralized control mechanism. Therefore, the fail of a few agents does affect the reliability of the system.

- **Adaptation**: agents can change according to system changes.
- **Autonomy**: almost no human supervision is required.
- **Parallelism**: agent's operations are inherently parallel.

The idea of ant algorithms derives from real swarms of certain insects, which develop a form of 'swarm intelligence' and solve complex problems through cooperation. One autonomous ant itself obeys primitive instincts and only performs limited, specialized tasks. However, the colony at large shows a global intelligent behavior. Swarm Intelligence (SI) is an artificial intelligence technique based around on the study of collective behavior in decentralized, self-organized systems. SI systems are typically made up of a population of simple agents interacting locally with one another and also with their environment. Usually there is no centralized control structure dictating how the individual agents should behave, but local interactions between such agents often lead to the emergence of a global behavior. Examples of systems like this can be found in nature, including ant colonies, bird flocking, bee swarming, animal herding, bacteria molding and fish schooling.

3.1.1. Swarm Intelligence as an optimization method

The most successful swarm intelligence technique currently in existence is Ant Colony Optimization (ACO). ACO is a meta-heuristic optimization algorithm that can be used to find approximate solutions to difficult combinatorial optimization problems. In ACO artificial ants build solutions by moving on the problem graph and they, mimicking real ants, deposit artificial pheromone on the graph in such a way that future artificial ants can build better solutions. ACO has been applied successfully to an impressive number of optimization problems.

3.1.2. Use of Ant Colony Optimization technique

ACO is the primary optimization algorithm being used in SI [14]. The reason behind choosing ACO is its distributed nature and inherent randomness. The algorithm used for optimization is a purely a distributed algorithm. This holds serious implications for MANETs. Since in a MANETs the nodes are constrained by power, storage and processing power limitations, a purely distributed algorithm like ACO prevents per node computation load, reduces state information to be stored per node. The randomness built in the algorithm prevents convergence to local maxima/minima but has a tendency to search for global optimization of parameters.

3.1.3. Why Ant Colony Optimization is suitable for ad hoc network?

- The algorithm is fully distributed ⇒ there is no single point of failure;
- The operations to be performed in each node are very simple;
- The algorithm is based on an asynchronous and autonomous interaction of agents;
- It is self-organizing, thus robust and fault tolerant⇒ there is no need of defining path recovery algorithms;
- It is intrinsically traffic adaptive without any need for complex and yet inflexible metrics;
- It is inherently adaptive to all kinds of long-term variations in topology and traffic demand, which are difficult to be taken into account by deterministic approaches.

3.2. Overview

The ant based routing protocol solves the routing problem based on an artificial intelligence concept called swarm intelligence. We have seen that the problem of routing and especially the problem of finding a path between source and destination is very difficult in mobile ad-hoc networks due to the mobility of nodes.

Although not being aware we almost daily see applied swarm intelligence when we go to a park or on the street. Everybody knows bees or ants, but we would not regard them as intelligent since they are not capable of executing a complex task on their own. Nevertheless a swarm of bees or ants is intelligent since the insects influence each other's behavior e.g. through pheromones and in the end intelligent behavior emerges. For instance are ants capable of finding the shortest way back to their nest through going in the direction of the highest pheromone concentration. Thus you can say they take the approved path. In computer science this kind of intelligence can be used to solve different problems.

The first algorithm which presented a detailed scheme for MANET routing based on ant colony principles is Ant Colony Routing Algorithm (ARA)[15] .The algorithm has its roots in ABC [12] and Ant Net [16] routing algorithms for fixed networks, which are inspired by the pheromone laying behavior of ant colonies. The algorithm floods ants to the destinations while establishing reverse links to the source nodes of the ants. Nodes launch ant agents in a reactive manner in order to limit the overhead caused by them. AntHocNet is a hybrid algorithm having both reactive and proactive components. The algorithm tries to keep most of the features of the original Ant Net over AODV. In this algorithm, no special agents are needed for updating the routing tables rather data packets are delegated this task. Each data packet follows the pheromone or its destination and leaves the pheromone for its source. Pheromone is a quality metric representing the goodness of a link. The data packets are biased toward the paths that have higher pheromone values. Exponential pheromone decay is introduced as a mean of a negative feedback to prevent old routes from remaining in the routing tables.

Fig. 3.1. (a) Searching for destination (b) Pheromone leads to destination (c) shortest Path is most reinforced

Out of all the techniques inspired by the behavior of social insects, ant colony optimization (ACO) algorithms [17] have evolved as a promising solution for efficient routing in MANETs. In recent years, models of collective intelligence of ants have been transformed into useful optimization and control algorithms [21]. ACO [18] is a novel evolutionary stochastic algorithm with characteristics such as positive feedback, distributive computing and the use of a constructive heuristic, etc., to match the demands of network optimization. ACO is based on the foraging behaviors of ants [18, 22]. Ants indirectly communicate by dropping a chemical substance called pheromone. Ants moving back and forth between nest and food deposit pheromone, and preferentially move towards areas of higher pheromone intensity as shown in Fig.3.1. The probability that the ant chooses a certain trail is proportional to the amount of pheromone on that trail. If it follows the trail, the ant's own pheromone reinforces the existing trail thus increasing the probability of the subsequent ants selecting the path. Shorter paths can be completed quickly and more frequently by the ants, and are therefore marked with more pheromone. This directly influences the selection probability for the next ant leaving the nest, which in turn increases the pheromone level on shorter path. Thus longer paths are less reinforced and ultimately abandoned. This distributed reinforcement learning process eventually allows the majority of the ants to converge onto the shortest. Pheromone evaporates over time, and if that path is not used for long time, pheromone will eventually vanish. Ants in network routing applications are simple agents(parallel processes) embodying intelligence and moving around in the network from one node to the other, updating the routing tables of the nodes that they visit with what they have learned in their traversal(history of visited nodes) so far. To reduce overhead history size has to be carefully chosen. This behavior is summarized in table 3.1.

Table 3.1: Ants food searching behavior

Steps	Description
Step1	Go (avoiding obstacles) in direction of highest pheromone concentration
Step2	If no pheromone noticed walk in random directions
Step3	If carry food, deposit pheromone in constant rates
Step4	If do not carry food and find food: take food and carry it
Step5	If carry food and inside nest: take off food

Although ACO has been successfully used to find high quality approximate solutions in many combinatorial optimization problems such as the Traveling Salesmen Problem (TSP) [26, 28, 31, 41], job-shop scheduling [30], electronic market, computer maps, traffic planning, computer games, connection-oriented and connection-less network routing problems [24, 25], graph coloring., sequential ordering problem and dynamic vehicle routing problem[23], this report focus entirely on discussion of ACO approaches in MANET routing. For last many years, ant based algorithms have captivated the researchers for solving routing problem in MANETs. Many algorithms have been proposed by researchers in last few years and many more are in pipeline. Existing algorithms have been able to address MANET routing issues up to some extent and those left untouched are being worked upon.

3.3. Existing Ant Based Algorithms in MANETs

The idea that ant-like agents can be used for routing in telecommunications network was introduced in mid 1990s, and has since then undergone many changes [29]. Year wise development of ant algorithms for routing in MANETs is presented in Table3.2.

Table3.2.Ant Algorithms for routing in MANETs

S. No.	Name of algorithm [reference]	Year
1	ARA [20,30]	2002
2	Ant AODV [31]	2002
3	PERA [32]	2003
4	AntHocNet [19]	2004
5	ADRA [33]	2004
6	ARAAI [34]	2005
7	ADDA [36]	2006
8	MAARA [35]	2007
9	HOPNET [37]	2008
10	PACONET [38]	2008

3.3.1. Ant colony based routing algorithm (ARA) [20, 30]

This algorithm uses two mobile agents FANT and BANT. FANT agent having unique sequence number and source address is broadcasted by the sender and will be relayed by the neighbors of the sender. A node receiving a FANT for the first time creates a record (destination address, next hop, pheromone value). The node interprets the source address of the FANT as destination address, the address of the previous node as the next hop, and the number of hops the FANT needed to reach this node decides the pheromone value. Thus FANT creates the pheromone track to the source node. The destination node extracts the information of the FANT, destroys it, and creates BANT which establishes the pheromone track to the destination node. The sender starts data transmission after receiving BANT.

No special packets are needed for route maintenance. Subsequent date packets are used instead to maintain the route. For a node *A* sending a data packet to destination *D* through a neighbor *B* pheromone value of the entry (*D*, *B*, *m*) is increased by Δm, thus strengthening the path to destination. Also, the next hop *B* increases the pheromone value of the entry (*S*, *A*, *m*) by Δm, thus strengthening the path to source node S. On receiving a duplicate packet, node sets the DUPLICATE ERROR flag, sends the packet back to the previous node refraining that node from sending more data packets in this direction, and hence preventing loops. When route failure occurs, node deactivates that path by reducing pheromone value to 0 in corresponding route table entry. Either the node sends the packet through alternate path if it exists, or the node try to transmit this packet through its neighbors. If packet still not reaches the destination, source initiates a new route discovery phase.

ARA fulfills the requirements of distributed operation, loop-freeness, on demand operation and sleep period operation (that is, nodes are able to sleep when their amount of pheromone reaches a threshold). Moreover, routing entries and statistic information are local to each node; several paths are maintained to reach a certain destination and, in a node with sleep mode on, only packets destined to it are processed.

Route Discovery

Route discovery is responsible for creating new routes. The creation of new routes requires the use of a FANT which is initiated at the source and a BANT which is initiated at the destination. The FANT is a small packet with a unique sequence number. A forward ant is broadcasted by the sender and will be relayed by the neighbors of the sender as shown in Figure 3.2.

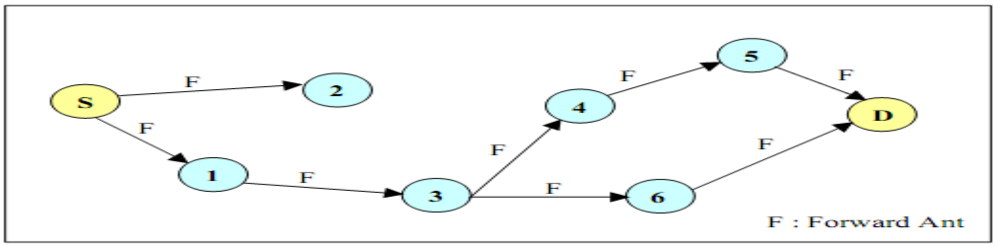

Figure 3.2. Route discovery phase by forward Ant.

A node receiving a FANT for the first time creates a record in its routing table. A record in the routing table consists of (destination address, next hop, pheromone value). The node interprets the source address of the FANT as destination address, the address of the previous node as the next hop and computes the pheromone value depending on the number of hops the FANT needed to reach the node. The node forwards the FANT to its neighbors. If the FANT reaches the destination, a BANT will be returned to the source.

The BANT will take the same path as the FANT but in the opposite direction as indicated in Figure 3.3. When the sender receives the BANT from the destination node, the path is established and data packets can be sent. Figures 3.2 and 3.3 schematically depict the route discovery phase.

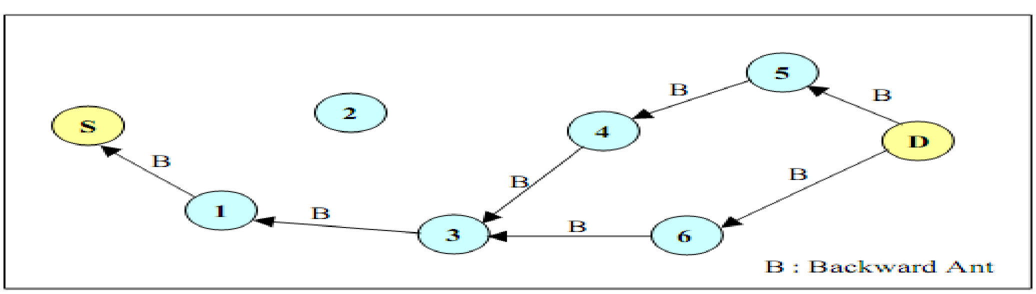

Figure 3.3. Route discovery phase by backward Ant.

Properties of ARA

According to [27] a routing algorithm for mobile ad-hoc networks should fulfill the following requirements:

- **Distributed operation**: In ARA, each node owns a set of pheromone counter $\phi_{i,j}$ in its routing table for a link between node v_i and v_j . Each node controls the pheromone counter independently, when ants visit the node on route searches.

- **Loop-free**: The nodes register the unique sequence number of route finding packets, FANT and BANT, so they do not generate loops.

- **Demand-based operation**: Routes are established by manipulating the pheromone counter $\phi_{i,j}$ in the nodes. Over time, the amount of pheromone decreases to zero when ants do not visit this node. A route finding process is only run, when a sender demands.

- **Sleep period operation:** Nodes are able to sleep when their amount of pheromone reaches a threshold. Other nodes will then not consider this node.

Additionally, ARA has the following properties:

- **Locality:** The routing table and the statistic information block of a node are local and they are not transmitted to any other node.

- **Multi-path**: Each node maintains several paths to a certain destination. The choice of a certain route depends on the environment, e.g., link quality to the relay node.

- **Sleep mode**: In the sleep mode a node snoops, only packets which are destined to it are processed, thus saving power.

Overhead of ARA

The expected overhead of ARA is very small, because there are no routing tables which are interchanged between the nodes. Unlike other routing algorithms, the FANT and BANT packets do not transmit much routing information. Only a unique sequence number is transmitted in the routing packets. Most route maintenance is performed through data packets, thus they do not have to transmit additional routing information. ARA only needs the information in the IP header of the data packets.

3.3.2. AntHocNet [19]

It is a hybrid algorithm, which combines reactive path setup with proactive path probing, maintenance and improvement. Multiple routes are set on demand at the start of a data session by broadcasting reactive FANT. For any FANT, its corresponding BANT retraces its path back to the source and updates the pheromone value. When a node receives several ants of same generation, it will compare the route of each new ant to that of the previously received ants of this generation only, if its number of hops and travel time are both within an acceptance factor of that of the best ant of the generation, the ant will be forwarded. Thus overhead is reduced by removing ants which follow bad paths. Routes are set up in the form of pheromone tables indicating their respective quality and then data packets are routed stochastically over these paths. Routes are monitored, maintained and improved by proactive FANTS. In the rest of this section we describe each of these functions in detail.

Reactive Path Setup

- Source s broadcast a reactive forward ant F_d^s.
- At each node, an ant is either unicast or broadcast, according to whether or not the node has routing information for d. The routing information of a node i is represented in its pheromone table T^i.
- If no pheromone is available for d, the ant is broadcast. Due to these broadcasting, ants can proliferate quickly over the network, following different paths to the destination.

- When a node receives several ants of the same generation, it compares the path travelled by each ant to that of the previously received ants of this generation, only if its number of hops and travel time are both within an acceptance factor a1 of that of the best ant of the generation, it will forward the ant.

Proactive path probing, maintenance and exploration

- While a data session is running, the source node sends out proactive forward ants according to the data sending rate.
- They are normally unicast, choosing the next hop according to the pheromone values using the same formula as reactive forward ants, but also have a small probability at each node of being broadcast.
- This way they serve two purposes. If a forward ant reaches the destination without a single broadcast it probes an existing path. If, on the other hand, the ant got broadcast at any point, it leaves the currently known paths and explores new ones.
- If the proactive ant does not find routing information within hops, it is killed. The effect of this is that the search for new paths is concentrated around the current paths, so that we are looking for path improvements and variations.

Link failures are dealt with either a local route repair using ants. Or by unicasting a warning back to preceding nodes on the paths, thus removing the wrong routing information.

3.3.3. Ant-ad-hoc on demand distance vector routing (ANT AODV) [31]

In Ant-AODV independent ant agents implement increased node connectivity (the number of destinations for which a node has un-expired routes), thus reducing the amount of route discoveries. Even if node launches on-demand route discovery for replacing stale route entries with fresh ones, there is high probability of receiving quick response from its neighbors due to the increased connectivity, hence resulting in reduced route discovery latency. Due to constant route updates, a source node can switch from a longer (and stale) route to a newer and shorter route, thus decreasing the average end-to-end delay by considerable amount. This makes Ant-AODV hybrid routing protocol suitable for real-time data and multimedia communication.

Ant-AODV uses route error messages (RERR) to inform upstream nodes of a local link failure. Ants randomly choose next hop (avoiding the previously visited node) from the list of neighbors in the routing table. Frequent HELLO broadcasts are used to maintain a neighbor table. If a node does not have a fresh route to destination, it will have to keep the data packets in its send buffer till an ant comes and tells it the route to that destination. In absence of local connectivity maintenance, even for a broken route the source will keep on sending data packets unaware of the link breakage, thus losing packets.

3.3.4. Probabilistic emergent routing algorithm (PERA) [32]

This algorithm exploits the inherent broadcast capability of wireless networks to reach a better performance. In this we approach, multiple routes are discovered by two small (variable) size agents FANT and BANT. These agents create and adjust a probability distribution at each node for the node's neighbors. The probability associated with a neighbor reflects the relative likelihood of that neighbor forwarding and eventually delivering the packet. Neighbor discovery is done using 'HELLO' broadcast messages. Routing table is initialized by adding all the neighbors (having initial equal probability value) in the table. The routing tables are then modified to attach higher probability to the node that the backward ant just came from, establishing a path toward the destination. To discover a route the node broadcasts FANT after pushing its own IP address and creation time of FANT on to the stack of the FANT. Subsequent nodes (which are not destination) adds their own address and the receiving time of ant on that node in FANT stack and decrement the hop count field of the FANT by 1. Duplicate FANTS are avoided by discarding all the ants with sequence number less than the number already stored in node's routing table for FANTs from that same previous node. If the sequence number is greater than that previously recorded by the node, the node updates this value. At the destination, information extracted from FANT is used to create BANT, and then FANT is killed. BANT retraces the path of the corresponding FANT that triggered its creation. The BANT travels in unicast fashion on high priority queues back to the source node.

Using the top address of FANT stack, the node forwards the BANT to the correct next hop. The data packets can be routed on the basis of the highest probability for the next hop at a node for the data packet's eventual destination.

3.3.5. Ant based distributed routing algorithm for ad-hoc networks (ADRA) [33]

To find new route source broadcasts FANT. The congested intermediate node ***i*** discards the FANTs received from node ***i-I.*** When node ***i*** has enough capacity to allow new route to he built, new route entry will be created for loop-free ants and those inducing loops will be discarded. In absence of any route from node ***i*** to node ***s*** via next hop ***i-I,*** a new pheromone entry including destination node **s,** next hop node ***i-I,*** the probability of going from node ***i*** to the source node ***s*** via next hop node ***i-l*** is created. Also all the existing entries whose destination is node ***s*** and next hop is one of the neighbors of node ***i*** are updated by node ***i***. At destination FANT is used to create BANT. Then FANT is killed and BANT is send towards source, updating the probability route tables of intermediate nodes on that route and finally of source node. Data can then be sent to its destination following the maximum probability path. This algorithm has been able to balance load on network up to some extent.

The process of route maintenance deals with the congestion problem, the link-break problem and the shortcut problem.

When congestion on an intermediate node exceeds its predefined congestion threshold, node will send a congestion control anti-ant to its upstream neighbor nodes in order to modify their probability route tables by reducing the strength of pheromone. More the congestion more will be the lifetime of anti-ant. After that, node ***i*** uses other routes with relatively large probability of pheromone to forward packets to the same destination. In case no alternate route is available and source node receives the anti-ant, it will initiate a new route discovery process. When the intermediate node ***i*** discovers link break, it will delete the corresponding entry in its probability route table. If alternate routes to the same destination are available, one with maximum pheromone strength is used for subsequent data transmission. Also, the probability of each probability route table entry is updated. Otherwise, the link break message is subsequently passed to the upstream nodes. Finally when source node receives link break messages and no redundant route is available, it will initiate a new route discovery process to regain a new route for subsequent data transmission.

Due to frequent topology changes, shorter route can come into existence. In this case the new route will not be discovered easily, because the old trails are so strong that almost all the ants choose them. When a new node boots, it will listen to its neighbor for a while, record the received neighbor node list, and then broadcast it to its one-hop neighbors. Nodes with data to send can let the new entered neighbor or neighbor nodes in the pheromone table with low pheromone probability anticipate in predefined probability, forwarding certain amount of packets, and determine whether the shortcut occurs or not based on whether the enforce-ants come or not.

3.3.6. Ant routing algorithm based on adaptive improvement (ARAAI) [34]

When source wants to communicate with destination and no routes are available in routing table, source will initiate route discovery process. Firstly, some FANTS are flooded into the network to search new routes, and collect paths' information and intermediate nodes' local information as they travel. To avoid loops, ants keep track of already visited nodes and further search only unvisited nodes. With passage of time pheromone on each link keeps on evaporating. At destination, BANTS are created from FANTS and are sent back towards source on the reverse path of corresponding FANT, updating the corresponding pheromone values. Heuristic value information is taken into consideration for selecting one among many available routes. Ant colony algorithm has the problem of slow astringency and tending to stagnation behavior, leading to constrained searching ability of FANTS especially for large-scale MANETs. ARAAI deals efficiently with this problem by using adaptive coordination of the parameters to improve FANT's global searching ability.

Route maintenance phase deals with topology changes and broken routes in a network. The "time" item in neighbor table describes the valid period of each link, and exchange of hello message tells about link existence after the specific time. In case of broken route, either one among the alternate routes (if available) is chosen, or a ROUTE_ERROR message is forwarded to the upper nodes to find an alternative. At the same time, the link pheromone value is set to 0.

3.3.7. ADDA [36]

ADAA is an on demand multi-path algorithm. For considering the diversity and validity of solution, it updates the pheromone table by local and global updating rules to update pheromone of link and the probability route table. In ADAA, a node collects its neighbors' information by flooding HELLO messages, and creates its probability table to replace the traditional node route table from the feedback coming form each destination node. The bottleneck width of the FANT denotes the available bandwidth of the link between previous node and current node. ADDA selects route and balances traffic flows by bottleneck width and probability route table, as a result, the overheads of route discovery and route conversion are decreased largely.

To find route to destination, source node creates and broadcasts a FANT, containing its own address, sequence number and time of creation of FANT. Then all the nodes (other than destination)receiving this FANT for the first time records their own address and time of receiving FANT in route record of FANT and passes it to next node. Duplicate FANTs are discarded. The priority of both FANT and data packets are same, and they suffers the same delay and congestion. The relevant route information from the received FANT is used to create BANT which retrace the path of the corresponding FANT to the source node by unicasting. For reflecting the network status accurately, BANT is assigned the higher priority. When BANT arrives at the source node, the source node will wait for more BANTs to get more route information, and updates the probability route table of each node.

ADAA can reduce the frequency of the congestion effectively by selecting the path by its bottleneck width and distribute the traffic by multiple link-disjoint routes. The congested node will retrace the path of FANT to inform the source *S* to change route. *S* reduces the pheromone value of the route and updates the relevant probability table to increase the probability of other usable routes being selected. ADAA recognizes the route failure through a missing acknowledgement. For a broken link node will deactivates the link by setting its pheromone value to zero, and then informing the source *S* to select other available route to send data packets.

3.3.8. Multi agent ants based routing algorithm (MAARA) [35]

To find new route source sends FANT with unique sequence number towards destination. Intermediate nodes will update the route information in FANT. The destination node extracts the information of the FANT, creates a BANT and returns it to the source node.

The second phase of the routing algorithm is called route maintenance. MARA does not need any special packets for that purpose. Regular data packets are used to maintain the path by increasing and decreasing the pheromone values of links.

The third and last phase of MARA handles route failures which are recognized through a missing acknowledgement on the MAC layer. The pheromone value for broken link is set to 0.In case of no alternate route, source initiates new route discovery.

3.3.9. Hybrid ant colony optimization routing algorithm for mobile ad hoc network (HOPNET) [37]

The network is divided into zones. The HOPNET algorithm consists of the local proactive route discovery within a node's neighborhood and reactive communication between the zones. The size of the zone is determined by the radius length measured in hops. A node may be within multiple overlapping zones and zones could vary in size. The nodes can be categorized as interior (at distance less than radius), boundary/peripheral (at radius) nodes and exterior nodes (outside the zone). Zone is constructed by neighbor detection process based on exchange of hello messages between nodes.

There are five types of ants. This is indicated by particular value in the Type field- 0 for internal FANT (sent within zone), 1 for external FANT (sent between zones), 2 for BANT, 3 for notification ant, and 4 for error ant. Each node has two routing tables. Intrazone Routing Table (IntraRT) and Interzone Routing Table (InterRT). The IntraRT is proactively maintained by periodically sending internal FANTS towards destination within zone to collect information such as nodes entry or deletion in zone and link failure.

The rows in IntraRT indicate the neighbors of nodes which are one hop away and the column represents all the nodes in the zone Corresponding BANT change the concentration of the pheromone value on their journey back to the source. In this algorithm, an ant chooses a node that produces the best path from the node to the destination. To do so, an ant selects an unvisited node and explores its adjacent links before moving to next hop. This exploration strategy guarantees that no link is missed and thereby no good path is missed. The InterRT reactively maintains path to a node outside its zone. When a node fails to find the destination within its zone in the IntraRT table, the peripheral nodes of the zone are used to find routes between zones. The table consists of four fields: destination, sequence no, path and expire. The path (stored in the path field) to a destination (destination) is stored in the table for a certain period of time as indicated in the expire field. The external FANTs are first sent by the node to its peripheral or border nodes. At the boundary, the peripheral nodes check whether the destination is within its zone by searching for the destination or path in its IntraRT table or InterRT table. If not then the ants are forwarded to the next zone via the other peripheral nodes within its zone. This process continues until the destination is found.

Route maintenance is done in two ways. Due to proactive maintenance within a zone, Intrazone invalid route will automatically recover after a short duration. For an interzone invalid route, either the upstream node of the broken link will try to find alternative path to the destination which can then be used to send buffered packets, or an error ant will be sent to the source node. Then the source node will initiate a new FANT to find new route to that destination. In HOPNET, we can adjust the evaporation rate. Too slow evaporation rate (say 10%) leads to cycle. The evaporation rate of 20–25% was found to be suitable for this algorithm.

3.3.10. Improved ant colony optimization routing algorithm for mobile ad hoc Networks (PACONET) [38]

PACONET is a routing protocol for mobile ad hoc networks inspired by the foraging behavior of ants. It uses the principles of ACO routing to develop a suitable problem solution. To find routes from source to destination FANT explores the network in a restricted broadcast manner. The BANT retraces the path of FANT and updating the pheromone value at same time. Data packets are stochastically transmitted towards nodes having higher pheromone concentration along the path to the destination.

Each node in the network has a routing table whose size is the degree of the node times all the nodes in the network. The rows of the routing table of node N represent the neighbors of node N and the columns represent all the nodes in the network. Each pair (row, column) in the routing table has two values namely a binary value indicating if the node has been visited and the pheromone concentration.

For discovering route from S to D, when FANT from a source S arrives at a node N, the FANT can select best path from neighboring node to D by looking at the rows against the column D in the routing table. For all unvisited neighbors in column D, FANT will choose the node with the highest pheromone as the next hop. At destination BANT created from FANT, travels towards source updating the pheromone concentration for the destination column

HELLO messages are used to detect and monitor links to neighbors. In case of not receiving several HELLO messages from a neighbor, a link breakage is detected and its routing table can be updated by deleting the entries in the routing table for that neighbor. In PACONET, a node would wait several seconds to confirm a failed HELLO packet before modifying its routing table. Data packets going through this node towards the newly broken link will end up being lost due to the delay in updating the routing table.

Table 3.3. Comparison of various ant based algorithms for MANETs

Parameters	ARA	AntAODV	PERA	AntHocNet	ADRA	ARAAI	ADDA	MARA	HOPNET	PACONET
Routing overhead	less	more	Increase with increase in mobility	more	less	less	less	less	less but increase with increasing zone radius	More
Scalability	very less	-	-	less	more	less	-	-	more	-
Multiple paths	yes	No	yes	yes	yes	yes	yes	yes	yes	-
Route discovery latency	-	less	-	-	-	more	-	less	-	-
Average end-to-end delay	-	less	-	less	less	-	less	less	less	more but decreases with dearease in node's speed
Packet delivery ratio	less	less	-	Reduces with increase in MANET size	-	less	more and increases with increasing pause times	more and is not effected by network size	more than AntHocNet for large MANETS	less but increases with increase in Pause time
Routing approach (Reactive or proactive)	reactive	reactive	proactive	both	both	both	reactive	both	both	-
Loops/cycles	yes	no	-	no	no	no	-	-	no	No
Load balancing	no	no	-	-	yes	-	yes	-	no	No

CHAPTER 4

PROBLEM STATEMENT & OBJECTIVE

4.1. Problem Description

Current MANET routing protocols have a flat approach to routing needs. Though MANET nodes play various roles, current routing protocols have little or no consideration for that. The problem statement can be formalized as:

Out of all the techniques inspired by the behavior of the ants, which routing protocol gives the best performance on the basis of simulation parameters like delivery ratio, routing overhead, average end to end delay, etc., with varying pause time?

In this thesis report, the behavior of routing protocols based upon swarm intelligence was analyzed, simulated, evaluated and finally, one routing protocol was suggested as the best out of the evaluated routing protocols.

Most current MANETs routing protocols assume that the wireless network in benign and every node in the network strictly follow the routing behavior and is willing to forward packets to/for other nodes. Most of these protocols cope well with the dynamically changing topology. However, they do not address the problems when misbehavior nodes are present in the network.

A commonly observed misbehaviour is packet dropping. Practically, in a MANETs, most devices have limited computing and battery power while packet forwarding consumes a lot of such resources. The design of routing protocols for MANETs must consider the power and resource limitation of the network nodes, the time varying quality of wireless channels and possibility of packet loss and delay.

The main problem in mobile ad-hoc networks is still, the finding of a route between the communication end-points, which is aggravated through the node mobility. In the literature one can find many different approaches which try to handle this problem, but there is no routing algorithm which fits in all cases. In this thesis we present a new approach for an on-demand ad-hoc routing algorithm, which is based on swarm intelligence. Ant colony algorithms are a subset of swarm intelligence and consider the ability of simple ants to solve complex problems by cooperation. The interesting point is, that the ants do not need any direct communication for the solution process, instead they communicate by stigmergy. The notion of stigmergy means the indirect communication of individuals through modifying their environment. Several algorithms which are based on ant colony problems were introduced in recent years to solve different problems, e.g. optimization problems. To address these design requirements several design strategies for MANETs have been proposed and each having its fair share of advantages and limitations. Ad-Hoc routing algorithms should be self-configured, self-built and distributed routing algorithm.

4.2. Objective and Sub-tasks

In order to improve throughput in MANETs, routing algorithm should be chosen carefully. The throughput of different routing protocols with different topologies has been evaluated with different simulation time. The primary objective of this thesis is to improve the throughput in MANETs which is achieved by the following manner:

- To analyze, simulate and evaluate Ant Colony Based Routing protocol (ARA).
- To analyze, simulate and evaluate AntHocNet Routing protocol (AntHocNet).
- A comparison is being performed between the two protocols. In order to evaluate the performance of the network, the delivery ratio, average end to end delay and routing over head with different topologies was analyzed.

CHAPTER 5

SIMULATION ENVIRONMENT

5.1. Fedora 11

Fedora Core [43] is a free operating system base on Linux. Red Hat being developed by the open source community and the Red Hat engineers sponsor the development of Fedora. Fedora 11 (F11) and F12 are the release of the Fedora Project. Fedora 11, codenamed Leonidas, was released on June 9, 2009 [39]. This was the first release whose artwork is determined by the name instead of by users voting on themes. Some primary features of F11 are extensive performance improvements, support for Intel-based Macs ,a new Graphical User Interface (GUI) virtualization ,ext4(journaling file system for Linux, developed as the successor to ext3.) as the default file system(method of storing and organizing computer files and their data.),experimental Btrfs(GPL-licensed copy-on-write file system for Linux.) activated by I can’t believe it’s not BTR command line option at boot up [40] ,Faster Boot up aimed at 20 seconds.GCC 4.4,GNOME 2.26,KDE 4.2,2.6.29 kernel, Eclipse 3.4.2 ,Net beans 6.5,OpenOffice 3.1,Python 2.6,Xfce to 4.6,X server 1.6 and fprint-support for systems with fingerprint readers. Fedora **(http://Fedora-Commons.org)** is a robust, modular repository system for the management and dissemination of digital content. It is especially suited for digital libraries and archives, both for access and preservation. It is also used to provide specialized access to very large and complex digital collections of historic and cultural materials as well as scientific data. Fedora’s flexibility enables it to integrate gracefully with many types of enterprise and web-based systems, offering scalability (e.g., millions of objects) and durability (e.g., all of the information is maintained in files with no software dependency, from which the complete repository can be rebuilt at any time). It also provides the ability to express rich sets of relationships among digital resources and to query the repository using the semantic web’s SPARQL query language. Fedora has a worldwide installed user base that includes academic and cultural heritage organizations, universities, research institutions, university libraries, national libraries, and government agencies. In figure 5.1 given complete architecture of Linux.

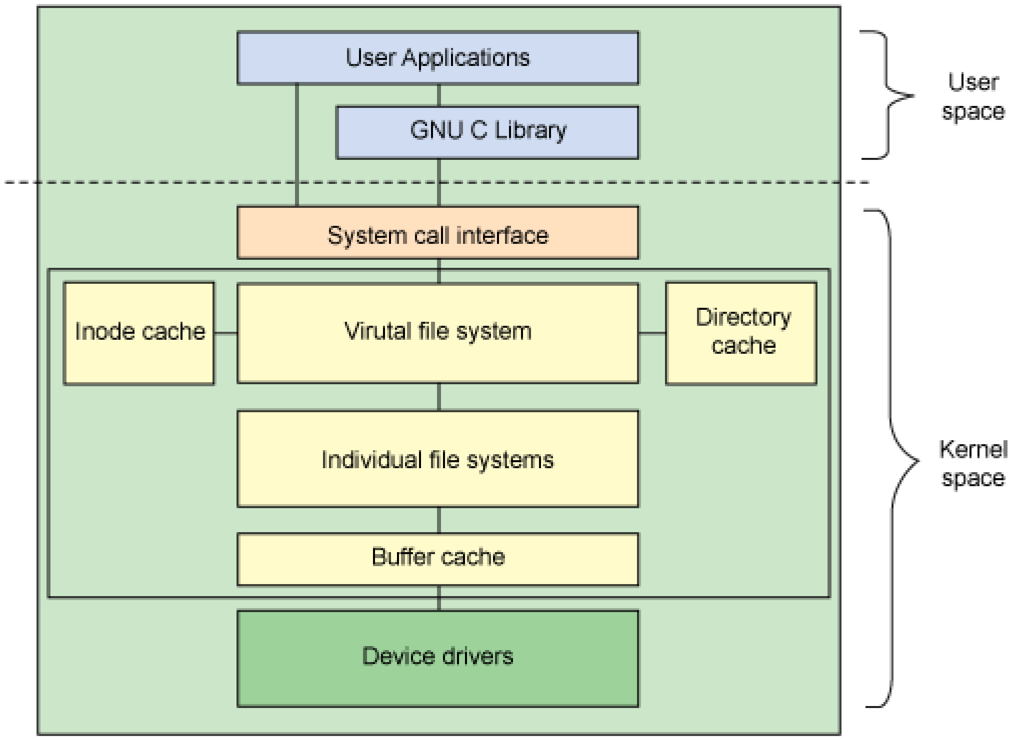

Fig 5.1.Architecture of Linux [41]

5.2. The Network Simulator (NS2)

Simulation [44] can be defined as "Imitating or estimating how events might occur in a real situation". It can involve complex mathematical modeling, role-playing without the aid of technology, or combinations. The importance of simulation lies in the consideration of realistic conditions that change as a result of behavior of others involved and thus we can anticipate the sequence of events or the final outcome. Different simulators such as ns2, GloMoSim, OPNET etc., are being used by researchers in order to evaluate the routing protocols. We have used ns2 for the evaluation of the proposed routing protocol as the same is an open source, freely available and the programming languages used are C++, Tcl / Tk and OTcl.

NS is a public domain simulator boasting a rich set of Internet Protocols, including terrestrial, wireless and satellite networks. NS is the most popular choice of simulator used in research papers appearing in select conferences like Sigcomm. NS is constantly maintained and updated by its large user base and a small group of developers at ISI.

5.2.1. NS2 Overview

NS [42] is an event driven network simulator developed at University of California at Berkeley, USA, as a REAL network simulator projects in 1989 and was developed at with cooperation of several organizations. Now, it is a VINT project supported by DARPA.NS is not a finished tool that can manage all kinds of network model. It is actually still an on-going effort of research and development. The users are responsible to verify that their network model simulation does not contain any bugs and the community should share their discovery with all. There is a manual called NS manual for user guidance.

NS (version 2) is an object-oriented, discrete event driven network simulator. NS is primarily useful for simulating local and wide area networks. Although NS is fairly easy to use once you get to know the simulator, it is quite difficult for a first time user, because there are few user-friendly manuals. Even though there is a lot of documentation written by the developers which has in depth explanation of the simulator, it is written with the depth of a skilled NS user. The purpose of this project is to give a new user some basic idea of how the simulator works, how to setup simulation networks, where to look for further information about network components in simulator codes, how to create new network components, etc., mainly by giving simple examples and brief explanations based on our experiences. Although all the usage of the simulator or possible network simulation setups may not be covered in this project, the project should help a new user to get started quickly. NS is a discrete event network simulator where the timing of events is maintained by a scheduler and able to simulate various types of network such as LAN and WPAN according to the programming scripts written by the user. Besides that, it also implements variety of applications, protocols such as TCP and UDP, network elements such as signal strength, traffic models such as FTP and CBR, router queue management mechanisms such as Drop Tail and many more. Currently, ns (version 2) written in C++ and OTcl (Tcl script language with Object-oriented extensions developed at MIT) is available. The compiled C++ programming hierarchy makes the simulation efficient and execution times faster. The OTcl script which written by the users the network models with their own specific topology, protocols and all requirements need. The form of output produce by the simulator also can be set using OTcl. The OTcl script is written which creating an event scheduler objects and network component object with network setup helping modules.

To setup a simulation network, an OTcl script is written and to simulate it the script is executed which initiates an event scheduler and the network topology is setup using the network objects, controlling the traffic sources and the time to start and stop the transmitting of packets.

Figure 5.2 shows the general architecture of ns. In this figure, a general user (not an ns developer) can be thought of standing at the left bottom corner, designing and running simulations in Tcl using the simulator objects in the OTcl library. The event schedulers and most of the network components are implemented in C++ and available to OTcl through the OTcl linkage that is implemented using tclcl. The whole thing together makes the ns, which is an OO extended Tcl interpreter with network simulator libraries.

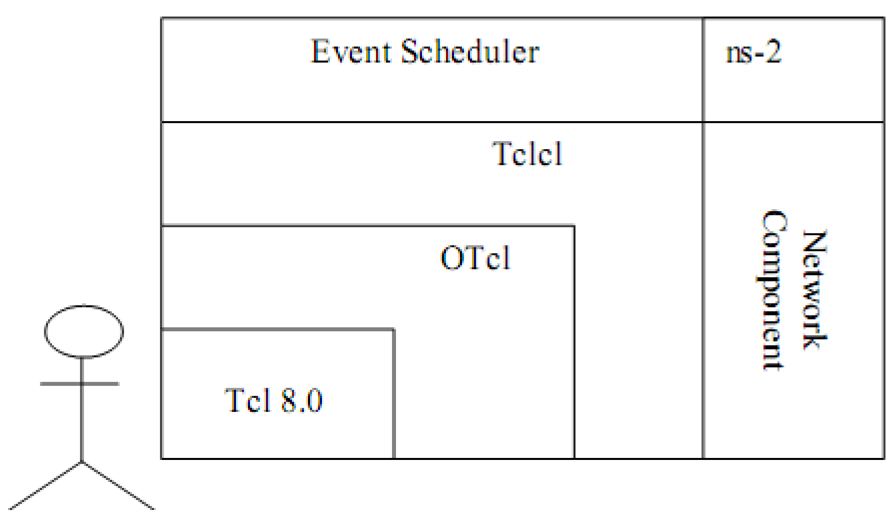

Figure 5.2. Architectural view of ns.

The simulation results produce after running the scripts can be use either for simulation analysis or as an input to graphical software called Network Animation (NAM).

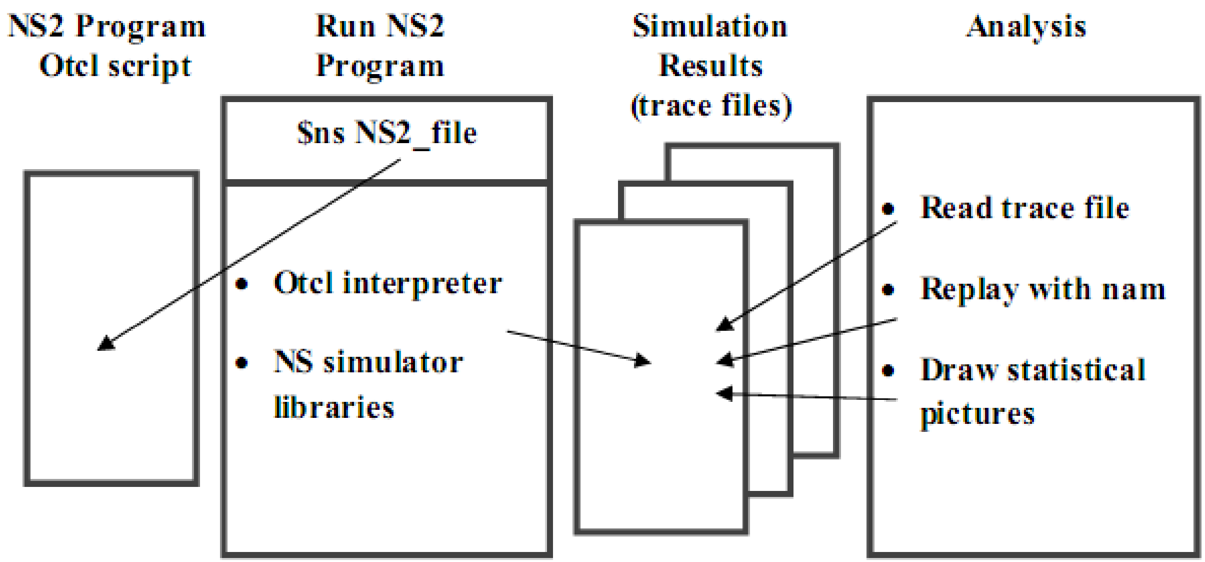

Fig. 5.3. Running NS2 Program [42]

NS2 is an event driven network simulator, which can be implemented in Linux-based platform. This report will explain on how to install NS2 in Fedora Core platform. The NS2 files (recommended to download a piece of file which includes all the needed files called ns-allinone-2.xx from http://www.isi.edu/nsnam/ns/ must be downloaded into any media storage, most preferred is inside the computer itself where the NS2 is going to be installed. Since, we are using NS 2.34. It is not recommend logging in as a root because installation at root may interfere with any important Linux files.

5.2.2. Tool Command Language/Tee-Kay (Tcl/Tk)

Short for Tool Command Language, Tcl [45] is a powerful interpreted programming language developed by John Ouster out at the University of California, Berkeley. Tcl is a very powerful and dynamic programming language. It has a wide range of usage, including web and desktop applications, networking, administration, testing etc. Tcl is a truly cross platform, easily deployed and highly extensible. The most significant advantage of Tcl language is that it is fully compatible with the C programming language and Tcl libraries can be interoperated directly into C programs.

Tk (pronounced "tee−Kay") is an extension to Tcl which provides the programmer with an interface to the X11 windowing system. Note that Tk has been successfully compiled under X11 R4, X11 R5, X11 R6, as well as Sun's NeWS/X11 environments. Many users will encounter Tcl/Tk via the "wish" command. Wish is a simple windowing shell which permits the user to write Tcl/Tk applications in a proto typing environment.

5.2.3. The Network Animator Tool (NAM)

NAM is a Tcl/Tk based animation tool for viewing the simulation. It is mainly designed to animate the behavior of the network. It supports topology layout, packet level animation, and various data inspection tools. It allows designers to take in large amounts of information quickly, visually identify and study patterns in communication, and better understand causality and interaction among nodes in the network. NAM benefits from a close relationship with ns. It takes in a nam file generated by NS during a simulation run of the network and animates the process. Ns together with its companion, nam, form a very powerful set of tools for teaching networking concepts. *ns* contains all the IP protocols typically covered in undergraduate and most graduate courses, and many experimental protocols contributed by its ever-expanding users base.

The biggest advantage of network animator (NAM) is that it provides a graphical user interface (GUI) for the different simulation environment according to the parameters specified by the user. The Xgraph utility generates the graphical output of the input data (or trace files).

5.2.4. The Trace File

The trace file is an ASCII code files and the trace is organized in 12 fields as in Figure 5.4. below.

Event	Time	From node	To node	Pkt type	Pkt size	Flags	Fid	Src addrs	Dst addrs	Seq num	Pkt id

Fig. 5.4. Fields of Trace File

The first field is the event type and given by one of four available symbols r, +, - and d which correspond respectively to receive, enqueued, dequeued and dropped. The second field is telling the time which the event occurs. The third and fourth fields are the input and output node of the link at which the events takes place. The fifth is the packet type such as continuous bit rate (CBR) or transmission control protocol (TCP). The sixth is the size of the packet and the next field is some kind of flags. The eighth field is the flow identities of IPv6, which can specify stream color of the NAM display and can be use for further, analyze purposes. The ninth and tenth fields are the source and destination address in the form of "node.Port". The eleventh is the network layer protocol's packet sequence number. NS keeps track of UDP packet sequence number for the analysis purposes. The twelfth, which is the last field, is the unique identity of the packet. Results of simulation are stored into trace file (*.tr). Trace Graph was used to analyze the trace file.

5.2.5. The Trace graph

Statistical and graphical results are presented using a tool called Trace Graph [46]. It reads the trace file generated during simulation in NS. It requires MATLAB 6.0 libraries to be fully operational. The latest version of Trace Graph (version 2.02) enables researchers to plot not only 2D but also 3D graphs.

5.3. Simulation of Routing Protocols

Simulation of different routing protocols (ARA and AntHocNet) has been carried over to evaluate the performance. Various parameters that are considered for simulation are listed in table 5.1.

Table 5.1. Simulation Parameters for simulating selected Ant routing protocols

Metric	Value
Simulator	NS2(ver2.34)
No' of nodes	10
Routing protocol	ARA, AntHocNet
Pause time	15,30,60,120,240,480 sec.
Traffic type	CBR(Constant Bite Rate)
Simulation time	900 sec.
Simulation area	500mx500m

5.3.1. Metrics used for Simulation

To analyze the performance of our solution, various contexts are created by varying the number of nodes and node mobility. The metrics used to evaluate the performance of the protocols. The performance metrics are purposely chosen to show the difference in performance among the different routing methods. These metrics are the most crucial and common yardstick to measure the overall performance of the network routing algorithms. Similar types of metrics were also used in many other comparison related work. The performance metrics are defined as the followings.

- **Route Acquisition Time:** The time it takes a source node to find a route to a destination node.

- **Average End-to-End Delay or Mean Overall Packet Latency:** This implies the delay a packet suffers between leaving the sender application and arriving at the receiver application.

- **Routing Overhead:** The total number of routing packets transmitted during the simulation. i.e. the sum of all transmissions of routing packets sent during the simulation. For packets transmitted over multiple hops, each transmission over one hop, counts as one transmission.

$$\text{Routing overhead} = \sum \text{Transmissions of routing packets}$$

  Routing overhead is important to compare the scalability of the routing protocols, the adoption to low-bandwidth environments and its efficiency in relation to node battery power (in that sending more routing packets consumes more power). Sending more routing packets also increases the probability of packet collision and can delay data packets in the queues.

- **Packet delivery ratio:** The packet delivery ratio in this simulation is defined as the ratio between the number of packets sent by constant bit rate sources (CBR,"application layer") and the number of received packets by the CBR sinks at destination.

$$\text{Packet delivery ratio} = \frac{\sum \text{CBR packets received by CBR sinks}}{\sum \text{CBR packets sent by CBR sources}}$$

It describes percentage of the packets which reach the destination.

CHAPTER 6

SIMULATION, EVALUATION & ANALYSIS

6.1. Simulation

We evaluate two swarm based selected algorithms on a number of simulation tests. We compare its performance between them (with route repair), a state-of-the-art MANET routing algorithm and a de facto standard. In next sections, we describe the simulation environment and discuss the results. The analysis is being done on the basis of the results of *.nam file and the *.tr file. We also evaluate the performance of the protocol. The analysis is being done on the basis of the results of *.nam file and the *.tr file. The scripts for the NAM is stored as *.nam and for trace graph *.tr is used. The two algorithms selected are given below:

- Simulation of Ant Colony Based Routing protocol(ARA) [20,30]
- Simulation of AntHocNet Routing protocol(AntHocNet) [19]

The comparison is done between ARA and AntHocNet is performed over the common factors like throughput of dropped packets, end-to-end delay ,packet routing overhead and packets delivery ratio in the network over different simulation time. There are two critical components that determine the performance of an ant-based routing algorithm. First, the performance depends on how the mobile agents search for the shortest path, which is referred to as network exploration. The mobile agents could also search independently of the routing tables. Second critical step determining the performance of ant-based routing algorithms is the routing table update. The mobile agents have to update the routing tables based on the paths that have been searched.

6.2. Simulation Environment for ARA

We have simulated ARA with ns-2 [19]. For our results we assumed 10 mobile nodes communicating via IEEE 802.11. The nodes move inside a simulation area of 500mx500m. The simulation time is 900 seconds. The nodes move with a maximal velocity of 10 m/s and according to varying pause time. Subsequently, the node drives to the selected point at constant speed. After arriving at the end point the node remains there for a certain time. Subsequently, the node repeats the operation by selecting a new end point and a new speed. We performed simulations with 7 different pause times of 0, 30, 60, 120, 300, 600 and 900 seconds. When the pause time is 0 seconds, the nodes move constantly. In contrast, when the pause time is 900 seconds the nodes do not move at all. As mentioned in the introduction our main goal was to reduce the routing overhead. So we will mainly discuss this aspect of ARA.

6.3. Simulation Results for ARA

In the presence of high mobility, link failure can happen very frequently. Link failures trigger new route discoveries in ARA since it has almost one route per destination in its routing table. Thus the frequently occurrences of route discoveries in ARA is directly proportional to the number of route breaks.

6.3.1. Packet delivery ratio

Figure 6.1 depicts the delivery rate of the current version of ARA with its extensions. Simulation was done with simulation time 900 sec, 10 mobile nodes with a varying s of pause time and with a maximum speed of 10m/s in a 500x500 region. After the simulation and analyzing the trace files, it have been obtained the figure6.1(graph) as presented below shows Packet Delivery Ratio

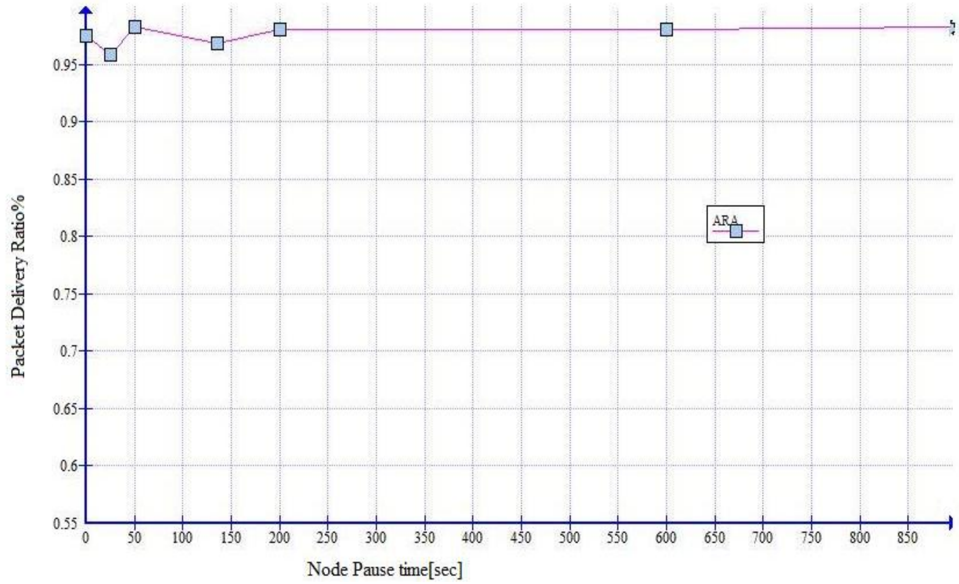

Figure 6.1. ARA by the fraction of successfully delivered packets as a function of pause time. Simulations with 10 CBR connections.

6.3.2. Average routing overhead

Simulation was done with simulation time 900 sec, 10 mobile nodes with a varying of pause time and with a maximum speed of 10m/s in a 500x500 region. After the simulation and analyzing the trace files, it have been obtained the figure6.2(graph) as presented below shows Average routing overhead (packets)

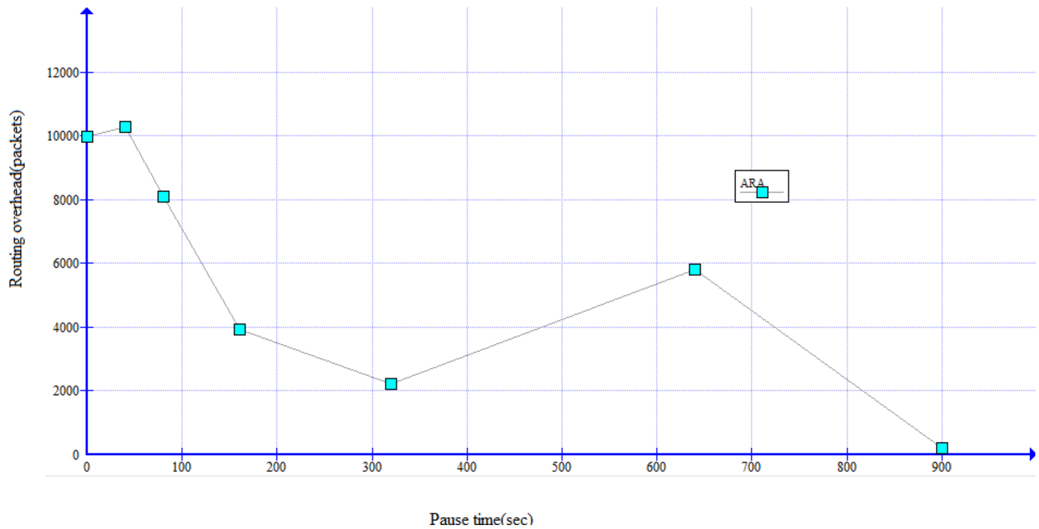

Fig6.2. ARA by the fraction of successfully average routing overhead as a function of pause time. Simulatiotions with 10 CBR connections.

6.4. Simulation Environment for AntHocNet

We have simulated ARA in ns-2 [19]. For our results we assumed 10 nodes move in a flat, rectangular area of 500 x500 m^2. Each simulation runs for 900 seconds. 10 different Constant Bit Rate sources start sending at a random time between 0 and 180 seconds and keep sending till the end. As measures of performance, we use the average end-to-end delay for data packets and the ratio of correctly delivered versus sent packets. These are standard measures of effectiveness in MANETs. We also report delay jitter, the average difference in inter-arrival time between packets. As measure of efficiency, we consider routing overhead, in terms of number of control packets forwarded per successfully delivered data packet.

We investigate AntHocNet performance for varying levels of pause time and node density, for different data traffic patterns. At the end of the section, we also show results reporting the performance of the algorithm at a smaller time scale: following the evolution of the end-to-end delay over the course of a simulation session while some disruptive events take place, we attempt to give an idea of the adaptivity of the algorithm.

6.5. Simulation Results for AntHocNet

AntHocNet algorithm was evaluated in terms of average end-to-end delay per packet and delivery ratio (the fraction of successfully delivered data packets), We also consider delay jitter and routing overhead. Delay jitter measures packet delay variation. It is calculated as the average of the difference of interracial time between subsequently received packets: the session's jitter is the arithmetic average of the values $(t_3-t_2)-(t_2-t_1)$ for all triplets of subsequently received packets, where t_1 is arrival time of the first packet and t_3 of the last. Routing overhead measures the algorithm's efficiency and is calculated as the total number of control packets sent divided by the number of data packets delivered successfully. The results are reported in figures 6.3, 6.4 and 6.5. AntHocNet shows good better effectiveness in terms of average delay, delivery ratio, and jitter. The bad performance for high pause times is due to the reduced connectivity.

6.5.1. Packet delivery ratio

Simulation was done with simulation time 900 sec, 10 mobile nodes with a varying of pause time and with a maximum speed of 10m/s in a 500x500 region. After the simulation and analyzing the trace files, it have been obtained the figure6.3(graph) as presented below shows Packet delivery ratio.

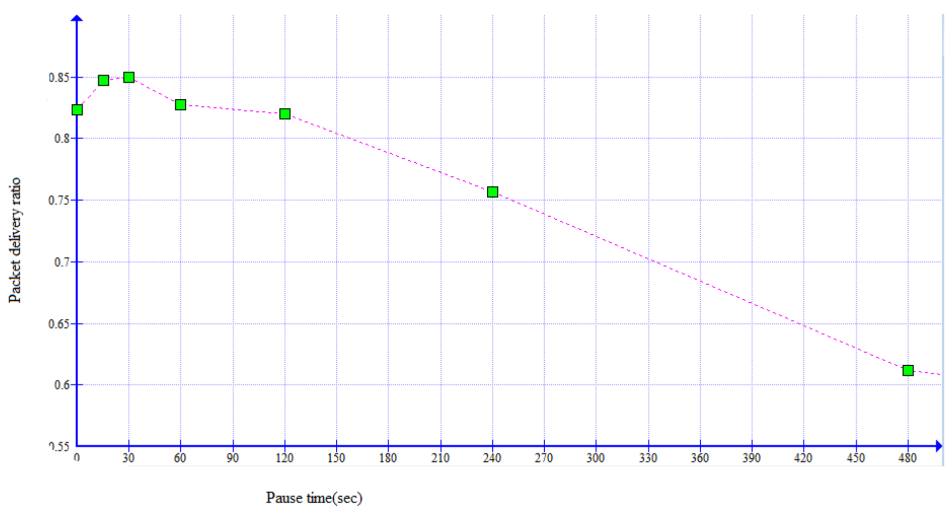

Figure 6.3. AntHocNet by the fraction of successfully packet delivery ratio as a function of pause time. Simulatiotions with 10 CBR connections

6.5.2. Average End-to-End Delay

Simulation was done with simulation time 900 sec, 10 mobile nodes with a varying of pause time and with a maximum speed of 10m/s in a 500x500 region. After the simulation and analyzing the trace files, it have been obtained the figure6.4(graph) as presented below shows Average End-to-End Delay.

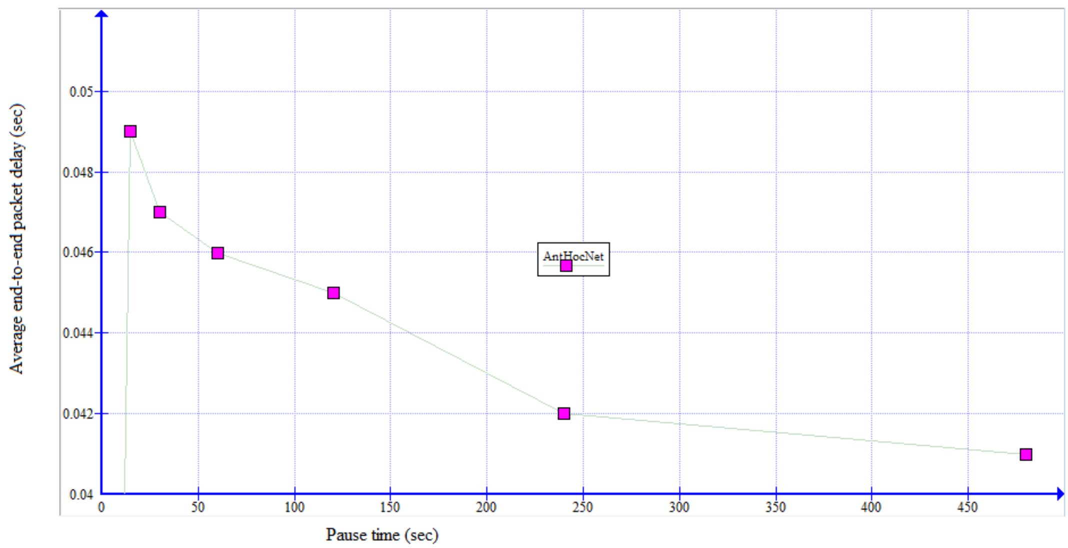

Figure 6.4. Average delay under various pause times for AntHocNet.

6.5.3. Average routing overhead

Simulation was done with simulation time 900 sec, 10 mobile nodes with a varying of pause time and with a maximum speed of 10m/s in a 500x500 region. After the simulation and analyzing the trace files, it have been obtained the figure6.5 (graph) as presented below shows Average routing overhead.

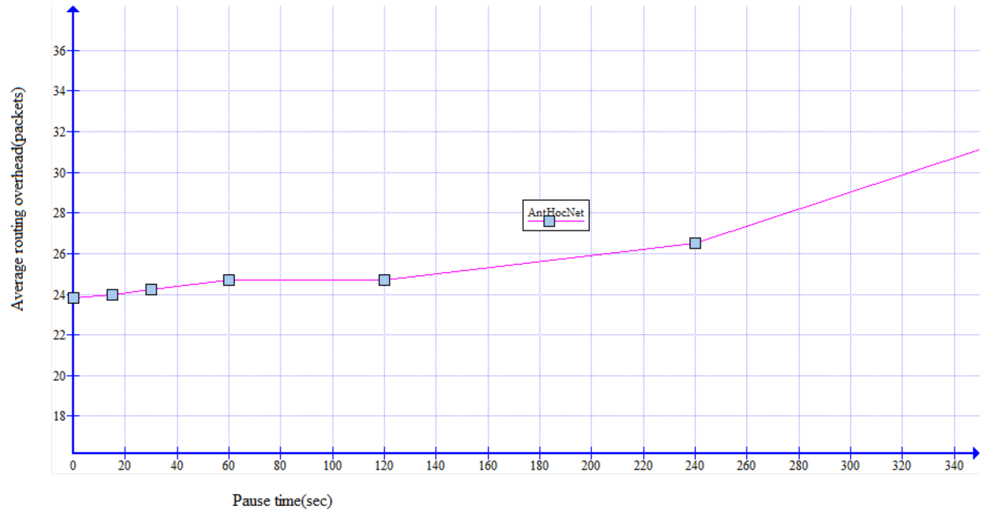

Figure 6.5. Average overhead under various pause times for AntHocNet.

6.6. Comparison of ARA with AntHocNet

We compared ARA, AntHocNet with respect to Delivered packet rate (i.e. the part of packets a certain routing protocol was able to deliver properly), overhead (fraction of successfully send bits and the needed bits and the number of needed routing packets). The performance improvement is based on the improved maintenance of the pheromone values which leads to stronger amplification of young paths and the exploitation of the multi-path capability of ARA, i.e. the load is balanced towards the destination.

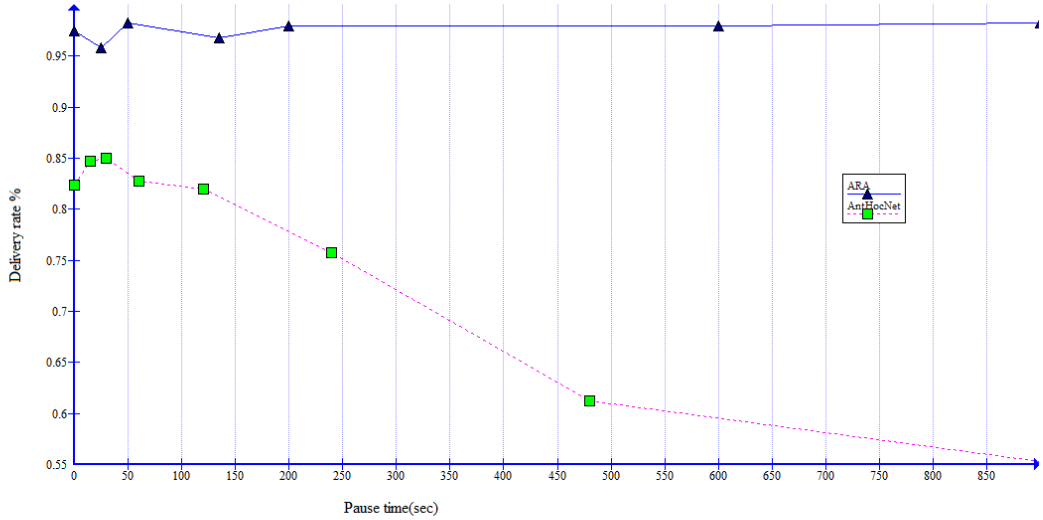

Figure 6.6.Comparision of two routing protocols by fraction of successful delivered packets as a function of pause time. Simulation with 10 CBR connections.

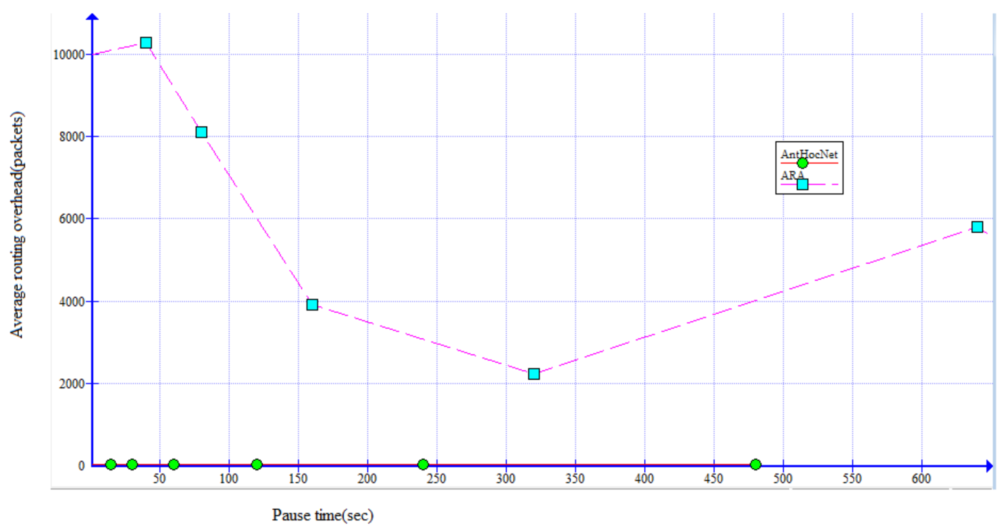

Figure6.7. Comparison of two protocols by the number of needed routing packets. Simulations with 10 CBR connections.

Fig 6.7 depicts the routing overhead of the two protocols. Again we observe advantages for AntHocNet, ARA in terms of average delay and delivery ratio. ARA overhead grows fast then AntHocNet. This is the important result which indicates that ARA is more scalable with respect to Mobility. In this case ARA shows the best performance.

Overhead in ARA is generated during the path discovery phase and the route failure handling by the FANT and BANT. The path discovery is performed for each new connection. With less mobility the number of route failure also decreases and therefore generate low overhead. ARA has better delivery ratio than AntHocNet, but higher average delay. As the environment becomes more difficult (high mobility), difference in delivery ratio is bigger, while the average delay of ARA becomes the better than of AntHocNet. The experiments described above show that ARA has clear advantage over AntHocNet. First of all, ARA gave a better delivery ratio than AntHocNet. The construction of multiple paths at route setup and the continuous search for new paths with proactive ants ensure that there are often alternate paths available in case of route failure, resulting in less packet loss. Second, ARA has higher average delay than AntHocNet. When we compare of these protocols by the fraction of successfully send bits and the needed bits, ARA is perform better than AntHocNet. And when we compare that needed packet over head then it perform better than AntHocNet.

CHAPTER 7
CONCLUSION & FUTURE SCOPE

This report first describes overview of Routing in MANETs and is an interesting research area that has been growing in recent years. Its difficulty is mainly because of the continuous changes in the network. There exist some traditional solutions such as proactive protocols, reactive protocols, and hybrid protocols with each one with its advantages and disadvantages. However, these solutions have to be improved to offer better performance. Researchers are designing new MANETs routing protocols, comparing and improving existing MANETs routing protocols. Routing is a significant issue in MANETs. The objectives listed in the problem statement have been carried out properly. Mobile multi-hop ad-hoc networks are flexible networks, which do not require pre-installed infrastructure. With upcoming wireless transmission technologies and highly sophisticated devices their application will increase. However the main challenge in mobile multi-hop ad-hoc networks is still the routing problem, which is aggravated by the node mobility. Various approaches were introduced in the recent years which try to handle the problems in this kind of networks, but no one fits best for all applications. In the presented work, we have discussed a comparison of two routing protocols for MANETS with varying pause times.

In this thesis we have described AntHocNet, ARA for MANETs which is inspired by ideas from Swarm Intelligence, and more specifically by the frame work of ACO. These algorithms combine reactive and proactive behavior to deal with the specific challenges of MANETs in an efficient way. Despite the improvements, the need to flood the network is a big disadvantage and not desired in mobile ad-hoc networks. A possibility to diminish this problem might by to use a flooding technique with less overhead. We will concentrate our further studies on the reduction of the overhead required and on the maintenance of the pheromone values. Pheromone diffusion provides an alternative way to learn pheromone information, it is used to help update pheromone on existing paths and to provide guidance to ants in search of new paths. We have evaluated these algorithms in an extensive set of simulation tests. With the results of trace graph, we can conclude that in the case of ARA delivery ratio of delivered packets is than the delivery ratio in the case of AntHocNet. Also, overhead is better in case of ARA.

Although ant algorithms listed in table3.1 are considered as one of the most promising approaches for optimizing routing problem in MANETs, still they pose many challenges for researchers. The successful test on the comparison of ARA and AntHocNet shows that our performance evaluation mechanism developed in our thesis is really effective for scalable performance test in NS-2. Firstly, computations must be performed so as to converge fast so as to cope with dynamic topology of MANETs. Secondly, overhead caused by proactive algorithms need to be reduced. Thirdly, route establishment in conventional ant-based routing techniques depend solely on the ants visiting the node and providing it with routes which may lead to ineffective routing in case of highly dynamic topology and shorter life of route. But still the dynamic component of these method leads to better adaptation to dynamic topology in MANETs. In particular, adding inter agent communication and intelligence to the ant agents in ANT-AODV along with the feedback such as node affinity and power levels etc. to the nodes could embed more intelligence in routing decisions. ARAAI can also be enhanced to give better efficiency, scalability, reduced route discovery delay. ADAA can be improved to optimize node energy, increase the lifetime of network and route. HOPNET is based on zone routing, which can be further explored to embed load balancing in this algorithm. PACONET can be upgraded to provide scalability and load balancing. Thus successful implementation of ant algorithms in MANETs poses great challenge before the researchers and needs to be tackled judiciously.

In future work we want to improve the exploratory working of proactive ants. By extending the concept of pheromone diffusion, more information about possible path improvements will be available in the nodes, and this information can guide proactive ants. This should lead to better results with less overhead. Also, we would like to try out a virtual circuit based approach. This could result in better control over paths, so that data delivery can be made more reliable.

[1] S. Corson and J. Macker, "Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations," RFC 2501, Jan. 1999.

[2] Stefano Basagni, Marco Conti, Silvia Giordano, Ivan Stojmenovic (Eds.), "Mobile Ad Hoc Networking," IEEE Press, Wiley Interscience. 2004.

[3] D. De Couto, J. Li, C. Blake, H. Lee, and R. Morris, "Capacity of Wireless Ad Hoc Wireless Networks," in Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom'01), Rome, Italy, pp. 61–69, July 2001.

[4] Vikram Goyal, Shyam K. Gupta, Anand Gupta, "Malafide Intension and its mapping to Privacy Policy Purposes for Masquerading," In Proc. 10th International Database Engineering and Applications Symposium (IDEAS 2006), pages 311-312, 2006.

[5] Mobile Ad-hoc Networks (MANET), http://www.ietf.org/html.charters/manetcharter.html.(2009-11-12).

[6] P. Michiardi and R. Molva, "Game Theoretic Analysis of Security in Mobile Ad Hoc Networks," Institut Eurecom Research Report RR-02-070, April 2002.

[7] Krishna Gorantala, ―Routing in Mobile Ad-hoc Networks‖, Umea University, Sweden, June-2006.

[8] S. Ramanathan and Streenstrup M., "A Survey of Routing Techniques for Mobile Communication Networks," ACM/Baltzer Mobile Networks and Applications, special issue on Routing in Mobile Communication Networks, Vol. 1, No. 2, pages 89-104, 1996.

[9] Geetha Jayakumar and Gopinath Ganapathy, ―Performance Comparison of Mobile Ad-hoc Network Routing Protocol‖, International Journal of Computer Science and Network Security (IJCSNS), VOL.7 No.11, pp. 77-84 November 2007.

[10] Robert Castaneda and Samir R. Das, "Query Localization Techniques for On-demand Routing Protocols in Ad Hoc Networks," In Proc. of the 5th International Conference on Mobile Computing and Networking (MobiCom'99). ACM, August 1999.

[11] Yih-Chun Hu and David B. Johnson, "Implicit Source Routing in On-Demand Ad Hoc Network Routing," In Proc. of the 2nd Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2001), pages 1-10, October 2001.

[12] Schoonderwoerd R, Holland O, Bruten J, Rothkrantz L. "Ant-Based load Balancing in telecommunications networks, Adaptive Behavior Hewlelt Packard Laboratories, Bristol-England, pp 162-207, 1996.

[13] Swarm Intelligence: From Natural to Artificial Systems (Santa Fe Institute Studies on the Sciences of Complexity).

[14] Internet Resource: Ant Colony Optimization and Swarm Intelligence: 4th International Workshop, ANTS 2004, Brussels, Belgium, September 5-8, 2004, Proceeding (Lecture Notes in Computer Science),

[15] I. Kassabalidis, A.K. Das, M.A. El-Sharkawi, R.J. Marks II,P. Arabshahi, A. Gray, " Intelligent routing and bandwidthallocation in wireless networks", Proc. NASA Earth ScienceTechnology Conf. College Park, MD, August 28-30, 2001.

[16] G. Di Caro and M. Dorigo. AntNet: Distributedstigmergetic control for communication networks. Journal of Artificial Intelligence, 9:317–365, Decembe1998.

[17] M. Dorigo and G. Di Caro, "The Ant Colony Optimization meta-heuristic," in *New Ideas in Optimization*, D. Corne et al., Eds., McGraw Hill, London, UK, pp. 11–32, 1999.

[18] M. Dorigo and T. St¨utzle. "Ant Colony Optimization," *MIT Press, Cambridge*, MA, 2004.

[19] G. Di Caro, F. Ducatelle, and L.M. Gambardella. "AntHocNet: an ant-based hybrid routing algorithm for mobile ad hoc networks*," Proceedings of Parallel Problem Solving from Nature (PPSN VIII),*Vol. 3242 of LNCS, pages 461-470. Springer -Verlag, 2004.

[20] M. G-unes, M. K-ahmer, I. Bouazizi. “Ant Routing Algorithm (ARA) for Mobile Multi-Hop Ad-Hoc Networks - New Features and Results,” *The Second Mediterranean Workshop on Ad Hoc Networks*, 2003.

[21] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant System: Optimization by a colony of cooperating agents,” *IEEE Transactions on Systems, Man, and Cybernetics—Part B,* vol. 26, no. 1, 1996.

[22] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. “Inspiration for Optimization from Social Insect Behaviour,” Nature, 406:39–42, July 2000.

[23] R. Montemanni, L.M. Gambardella, A.E. Rizzoli, and A.V. Donati, “Ant colony system for a dynamic vehicle routing problem,” *Journal of Combinatorial Optimization*, vol. 10, pp.327–343, 2005.

[24] G. Di Caro and M. Dorigo. “Mobile agents for adaptive routing,” In *Proceedings of the 31st Hawaii InternationalConference on Systems Science*, pages 74–83,January 1998.

[25] G. Di Caro and M. Dorigo. “AntNet: Distributed stigmergetic control for communications networks,” *Journalof Artificial Intelligence Research*, 9:317–365,1998.

[26] Marco Dorigo and Luca Maria Gambardella. “Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem,” *IEEE Transactions on Evolutionary Computation*, 1:53–66, 1997.

[27] Joseph P. Macker and M. Scott Corson. Mobile ad-hoc networking and the IETF. Mobile Computing and Communications Review, 2(1):9–14,1998.

[28] Eugene L. Lawler, Jan Karel Lenstra, Alexander Hendrik George Rinnooy Kan, and David B. Shmoys, editors. “The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization” *Wiley Series in Discrete Mathematics & Optimization*. *JohnWiley &Sons*, September 1985. ISBN 0471904139.

[29] D. Subramaniam, P. Druschel and J. Chen, “Ants and Reinforcement Learning: A Case Study in Routing in Dynamic Networks,” *Proc. of the Fifteenth International Joint Conference on Artificial Intelligence,* 1997.

[30] Gunes, M.,Sorges, U., Bouazizi, I. “ARA-the ant-colony based routing algorithm for MANETs,” *Parallel Processing Workshops,* 2002. *Proceedings. International Conference,* Page(s):79-85,2002.

[31] Marwaha S, Tham CK, Srinivasan D. “Mobile agents based routing protocol for mobile ad hoc networks,”*Proceedings of the IEEE Global Communications Conference (GlobeCom)*, 2002.

[32] John S. Baras and Harsh Mehta. “A Probabilistic Emergent Routing Algorithm for Mobile Ad hoc Networks,” *In WiOpt'03:Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, March 3-5, 2003.

[33] Xiangquan Zheng, Wei Guo, Renting Liu. “An ant-based distributed routing algorithm for ad-hoc networks,” *(ICCCAS) International Conference on Communications, Circuits and Systems,* Vol. 1, Page(s):412-417 ,2004.

[34] Zeng Yuan-yuan, He Yan-xiang. “Ant Routing Algorithm for Mobile Ad-hoc Networks Based on Adaptive Improvement,” *Proceedings of International conference on Wireless Communications, Networking and Mobile Computing, 2005,*Vol. 2, Page(s): 678 – 681, 2005.

[35] D.Sivakumar, R.S. Bhuvaneswaran, " Proposal on Multi agent Ants based Routing Algorithm for Mobile Ad-Hoc Networks", IJCSNS International Journal of Computer Science and Network Security, Vol7, No.6,261-268, July 2007.

[36] Xiang Yang, Li Layuan, Cheng Chuanhui . “Application Research Based Ant Colony optimization for MANET,”*Proceedings of IEEE International Conference on Wireless Communications, Networking and Mobile Computing, 2006. WiCOM* 2006.

[37] Jianping Wang, Eseosa Osagie, Parimala Thulasiraman, Ruppa K. Thulasiram. “HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network,” *Journal of ad hoc networks, Elsevier*,2008.

[38] Eseosa Osagie,Parimala Thulasiraman, Ruppa K. Thulasiram, “PACONET: imProved Ant Colony Optimization Routing Algorithm for Mobile Ad Hoc NETworks”,

22nd International Conference on Advanced Information Networking and Applications (aina 2008), pp. 204-211.

[39] "Red Hat Fedora 11 Focuses on the Linux Desktop". internetnews.com. April 28, 2009. http://www.internetnews.com/devnews/article.php/3817686/Red+Hat+Fedora+11+Focuses+on+the+Linux+Desktop.htm. Retrieved 2009-04-30.

[40] "Fedora 11 File systems". fedoraproject.org. http://docs.fedoraproject.org/releasenotes/f11/en-US/sect-Release_Notes-File_Systems.html. Retrieved 2009-07-04.

[41] Linux System Administrators' Guide 0.6, by Lars Wirzenius, http://www.iki.fi/liw/linux/sag/ or http://www.linuxhq.com/LDP/LDP/sag/index.html.

[42] S. McCanne and S. Floyd. Network Simulator. http://www.isi.edu/nsnam/ns/

[43] Fedora website. http://docs.fedoraproject.org/install-guide/fc4/en/

[44] A. Law and W. Kelton, "Simulation Modeling and Analysis," McGraw Hill, 2000.

[45] TCL Tutorial. http://www.tcl.tk/man/tcl8.5/tutorial/tcltutorial.html.

[46] Trace graph http://www.tracegraph.com/download.html.

List of Publications Annexure II

PUBLISHED

- P.kusakumar, S.G.Reddy, A.K.Verma, "Ant based Routing Algorithms for MANET's-A Review" in Proc. IISN 2010(pages133 -136).

- S.G.Reddy, P. Kusakumar, A .K .Verma, "Evaluation of Routing protocols for Mobile Ad hoc Networks" in Proc. of ICSCI-2010(pages 682-687).

COMMUNICATED

- P.kusakumar, A.K.Verma,"Performance Analysis of Swarm Based Routing Protocols for MANETs" in IEEE R10, the Second World Congress on Nature and Biologically Inspired Computing (NaBIC 2010).