Efficient Scheduler Based On Swarm Intelligence to Improve Quality of Service for MANETs

Dissertation

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

Masters of Technology

in

Computer Science and Applications

Submitted By
Priya
(Roll No. 601203018)

Under the supervision of
Dr. Rajesh Kumar
Associate Professor, SMCA

School of Mathematics and Computer Applications
Thapar University
Patiala –147004

July 2014

Certificate

I hereby certify that the work which is presented in the dissertation entitled, "Efficient Scheduler Based On Swarm Intelligence To Improve Quality Of Service For MANETs", in partial fulfillment of the requirements for the award of degree of the **Master of Technology** in **Computer Science and Applications**, submitted in School of mathematics and Computer Applications of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Rajesh Kumar and refers others researcher's work which are duly listed in the reference section.

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

Priya

601203018

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

17.7.2014

Dr. Rajesh Kumar

Associate Professor

School of Mathematics and Computer Applications

Countersigned by

Dr. Rajesh Kumar

Head

School of Mathematics and Computer Applications

Thapar University

Patiala

Dr. S. K. Mohpatra

Dean Academic Affairs

Thapar University

Patiala

Acknowledgement

First of all I would like to thank the Almighty, who has always guided me to work on the right path of the life.

This work would not have been possible without the encouragement and able guidance of my supervisor **Dr. Rajesh Kumar**. I thank my supervisor for his time, patience, discussions and valuable comments. Their enthusiasm and optimism made this experience both rewarding and enjoyable.

I am also thankful to **Vishal Sharma** for discussing his valuable ideas and comments with us.

I will be failing in my duty if I don't express my gratitude to **Dr. S. K. Mohapatra**, Senior Professor and Dean of Academic Affairs of University, for making provisions of infrastructure such as library facilities, immensely useful for the learners to equip themselves with the latest field.

I am also thankful to the entire faculty and staff members of School of Mathematics and Computer Applications Department for their direct-indirect help, cooperation, love and affection, which made my stay at Thapar University memorable.

Last but not least, I would like to thank my parents for their wonderful love and encouragement, without their blessings none of this would have been possible. I would also like to thank my close friends for their constant support.

Abstract

Mobile Ad Hoc Network (MANET) is a collection of wireless mobile nodes making a temporary network without using centralized access points, infrastructure, or centralized administration. It is a dynamic multi hop wireless network which is established by a set of mobile nodes on a shared wireless channel. For an efficient network formation, mobile nodes should be capable enough to utilize the available query resources optimally. One of the major issues in MANETs is routing due to the mobility of the nodes. Routing is the act of moving information across the network from a source to a destination. When it comes to MANETs, the complexity increases due to various characteristics like dynamic topology, time varying QoS requirements, limited resources and energy etc. The primary challenge in building MANETs is to find the optimal path between the end points for handling user's QoS requirements.

In this dissertation, to utilize the available query resources optimally, queue scheduling is proposed. Scheduling is a way by which multiple threads and processes get access to system resources. Scheduling algorithm improves the quality by allocation of packets to queues. Scheduling algorithms improves the quality by allocation of packets to queues. So for this, a scheduling algorithm based on swarm intelligence is proposed to improve QoS for MANETs. The approach combines the idea of ACO and BCO with the scheduling algorithm that schedules the packets transferring from individual nodes. Results showed that by integrating scheduling algorithm with ACO and BCO, the performance of AODV and AOMDV protocols improves as compare to that using only AODV and AOMDV. Better Packet Delivery Ratio, minimum packet loss and better throughput is obtained.

List of Figures

List of snapshots

List of Tables

Table of Contents

Chapter 1

Introduction

1.1 Overview of MANETs

The collection of wireless mobile nodes is called Mobile Ad-Hoc Networks (MANETs) that make a momentary network not using any centralized access points or centralized administration. MANETs comprise of mobile nodes that have capabilities of communicating with each other using packet radios over a shared wireless medium. It should be easy for nodes in the network to sense and discover nearby nodes [3]. But due to small range of wireless network, it needs multiple network hops to exchange data and information with other nodes over the network. This network is useful because of having a lot of advantages in terms of cost and flexibility as compare to wired network [6]. The main application areas of MANETs are data collection, seismic activities, medical applications, military applications, rescue operations, wearable devices and in other places where pre-installed infrastructure is not possible. It is hard to use multimedia and other advance techniques in dynamic network without quality-of- service (QoS) constraints. QoS is a set of services such as end-to-end delay, bandwidth, probability of packet loss, energy and jitter etc that are required to be satisfied during sending packets from source to sink. The QoS metrics are categorized in three types as additive, concave and multiplicative. The bandwidth and energy are concave because these two are calculated as minimum for all the links during the path, But cost, jitter and delay are additive in nature because these constraints are sum of all individuals along the path [7, 4]. Under multiplicative constraint reliability and availability of link are based on some criteria such as link break probability.

1.2 Issues related to quality for MANETs

MANETs should take care of stringent time and reliability-sensitive service requirements that multimedia applications need the most. MANETs should fulfill QoS requirements. Earlier techniques cannot guarantee today's requirements. So many solutions are proposed to fulfill quality metrics. It is a big challenge to provide better quality of service because of the following:

- *Node Mobility:* Nodes in MANETs are mobile; it makes the topology in dynamic nature. It means a limited life is there for topology information and for sending packets from source to sink its frequent updating is necessary. This updating increases the routing overhead. It affects the end to end delay and packet loss is also increased.

- *Lack of Central Control:* The advantage of MANETs is that it is deployed without any planning and can be changed dynamically. But it is difficult to have any centralized control. So the control activities are distributed all over the nodes that require a lot of information exchange, which adds overhead to the routing.

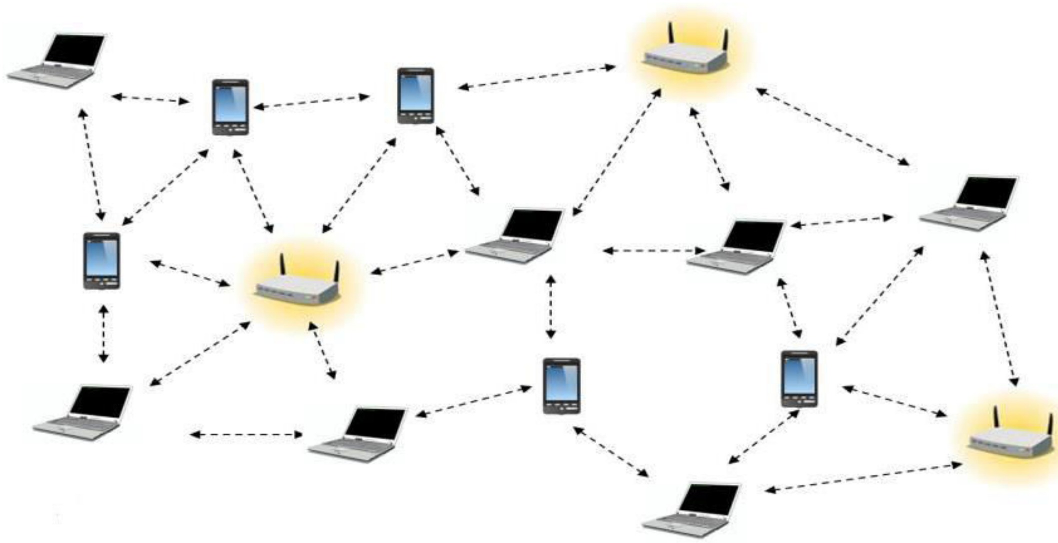


Figure1.1 view of MANETs [3]

There are already many routing protocols like AODV, TORA, DSR etc, but still there is need of improvement. With increase in the growth of internet, the demand for quality of service is also increasing.

1.3 Types of protocols for MANETs

Number of protocols is there in MANETs for routing process. These protocols are divided into two categories:

- Proactive.
- Reactive.

Proactive: In proactive routing protocol, each node keeps the record of entire topology of the network, for this a table is maintained that keeps information of connectivity of nodes *i.e.* which node is connected to which other nodes. The table is updated regularly so as to maintain up to date routing information. For this the information must be shared among the nodes on a regular routine, which leads to a high overhead over the network. On the other side, the routes are made available on request [27]. So a fresh list of destinations and routes is maintained by periodical distribution of routing tables throughout the network. Some of the proactive protocols are Distance Vector Routing (DSR), Destination Sequence Distance Vector Routing (DSDV), etc. These kinds of protocols are having following disadvantages:

- Overhead for data maintenance.
- Slow processing speed due to restructuring and maintenance.

Reactive: Reactive protocols are on demand protocols. They create a connection on request. In this type of protocols if a node has no route to the desired node even then the node can establish a connection with that node. This is done by broadcasting. The source node broadcasts its packets and the node receiving these packets updates its routing information. Reactive Protocol has lower overhead because routes are determined on demand [19]. MANETs multipath routing protocols have large ability to decrease the path finding frequency among the on-demand routing protocols. In on-demand multi-path routing protocols multiple paths are discovered from source to destination in a single route discovery [27]. This is the main advantage of reactive protocols because when all paths fail, only then a new discovery is needed. Some of the examples of reactive routing protocols are Ad hoc on-demand distance vector routing (AODV), Ad hoc on demand multiple distance vector routing (AOMDV) etc.

1.3.1 Ad hoc on-demand distance vector routing (AODV)

Ad hoc On-Demand Distance Vector (AODV) [33] Routing is a routing protocol for Mobile Ad Hoc Networks (MANETs) and other wireless ad hoc networks. In AODV, the network remains silent until a connection is desired. Then the requesting node broadcasts a request for connection. The nodes receiving the requests forward the messages to the other nodes so as to create temporary routes and then send back response to the needy node. If the node has already route to the desired node, it will send back messages to the needy node. When a node receives such a message from a node that already has a route to the desired node, it sends a message backwards through a temporary route to the requesting node. The requesting node then starts using the route that has the least number of hops through other nodes. The nodes which are not used are recycled after some time. If any link fails, it passes an error message to the sending node, so that process can be repeated. The complexity of this protocol is to lower number of messages to conserve the capacity of network. A sequence number is assigned to each request. The nodes keep this sequence number so that their route requests are not repeated with the ones that are already passed on. Another such feature is that the route requests have a "time to live" number that limits how many times they can be retransmitted. If a route request fails, another route request may not be sent until twice as much time has passed as the timeout of the previous route request [3]. The AODV overcomes disadvantages of proactive protocols as it creates no extra traffic for communication along existing links. The distance vector routing is simple enough to set, and doesn't require much memory or calculation [7]. But with many advantages there are some disadvantages of AODV that are:

- It requires more time to establish a connection.
- The initial communication to establish a route is heavier than some other approaches.

1.3.2 Ad hoc on demand multiple distance vector routing (AOMDV)

AOMDV is the advance form of AODV for computing multiple disjoint loop free paths in a route discovery. In this, each node has a unique number called unique identifier like IP address. All the links in MANETs are bidirectional in nature. AOMDV can be applied even in the presence of unidirectional links with additional techniques to help discover bidirectional paths. AOMDV possesses similar characteristics to AODV. The main concept here is distance between two nodes and uses hop-by-hop routing approach. Same as in AODV, AOMDV too calculates distant path on demand using route discovery method [33]. But there is one thing which differentiates AOMDV from AODV is the route discovery. In AODV if a source wants to make a connection with the destination node then it can be done by broadcasting a route request packet to the destination via network. If any delicacy is found in request then that request is discarded. These duplicate requests are detected by a unique sequence number. The halfway nodes which are receiving request packets firstly create a reverse connection if it founds a path to the destination node. If the route is valid then the intermediate node generates a route reply to the previous node otherwise it discards the message. The node then updates its routing table and sends the RREP to the previous node only if a RREP has a larger destination sequence number or if the route found is shortest. But in AOMDV, the RREQ is broadcasted from the route to the destination making multiple reverse paths both at intermediate nodes as well as the destination. Multiple RREPs traverse these reverse paths back to form multiple forward paths to the destination at the source and intermediate nodes [33]. The core concept of the AOMDV protocol is to find multiple paths from source to destination. It ensures that the multiple paths which are discovered should be loop disjoint and free. Multiple routes are discovered by multipath routing protocol during one route discovery procedure. When the primary route fails, the multiple paths are used for back up recovery. The paths should be found efficiently using flood-based route discovery. Route update rules in AOMDV are applied on each node and play a key role for maintenance of loop-freedom and disjoint properties. AOMDV is much useful as compare to other protocols because AOMDV mostly relies on the routing information that is already available in the underlying AODV protocol, so there is very limited overhead for discovering multiple paths. In general, it does not employ any special control packets. But additional overhead as compare to that of AODV is because of extra RREPs and RERRs for discovering multipath and maintaining along with a few extra fields in routing control.

There are many routing algorithms based on swarm intelligence to optimize the path of MANET nodes, two of them are explained below.

1.4 Ant Colony Optimization

The concept of ACO is inspired by observation of behavior of real ants while roaming for food. While moving from source to destination the ant lays a certain amount of pheromone trail (an odorous substance) along the path traversed. This pheromone serves dual purpose to the ant. Firstly, it helps other ants in searching food source *second* it helps ants to come back to their original source. Now, whenever a moving ant senses pheromone fragments it is likely that it will follow the same path and hence by mechanism of positive feedback this new ant lays more pheromone on that path. The amount of pheromone laid by the new ant will intensify the amount of already present pheromone trail on a specific path. The path having highest amount of pheromone trail will be selected by the new arbitrarily moving ant. The pheromone laid evaporates at a constant rate. This evaporation also helps in avoiding the convergence to a local optimal solution [29]. If an ant finds the shortest path to the destination randomly that ant will reach to the destination in minimum time and will be the first to reach to destination. Now in this forth and back traversal of the ant on the shortest path it will lay more pheromone trail, This path becomes an obvious choice for other ants after making a few hit and trials and hence quickly converging to the shorter path. This kind of behavior of ants from its colony to food source and back to its colony is called “ant colony optimization” [23].

To explain let us consider the experiment shown in Figure 1.2. There are a set of ants moves along a straight line from their nest C to a food source D. At a given moment, an obstacle AB is put across this way. The ants will thus have to decide which direction to choose either A or B. Firstly all the ants will follow random paths. The first ones will choose a random direction and will deposit pheromone along their way. Some of them will take CAD and some will follow CBD. They deposit pheromone on the path and find the shortest path [4]. Thus ants will follow the path having high deposit of pheromone. This quantity of pheromone is a function of the found route quality. After reaching to the destination evaporation of pheromone starts, so the path having high amount is selected by other ants so CBD is followed which is the shortest path. Because this path is shorter than the other one and the evaporation rate in this route is less than the other one so the coming ants can smell pheromone and follow this path.

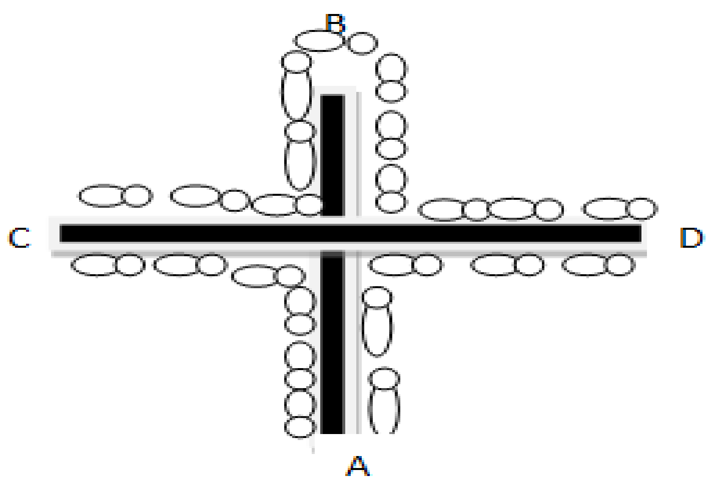


Figure 1.2 Bridge problem in ACO [4]

1.4.1 Ant colony optimization for travelling sales man problem

Here we have five test cases means vertices A, B, C, D, and E. Each node has an ant to start a path from that position. Because the ants are blind, they cannot see the path they can only smell. Firstly the ants move randomly in any direction while moving on the path the ants leave a chemical substance called pheromone trail, which the other following ants smell and follow the same path. The path having maximum quantity of pheromone trail will have the maximum number of ants and that path will be the final and shortest path. The process of finding the shortest path is to be completed in 4 iterations which are explained below.

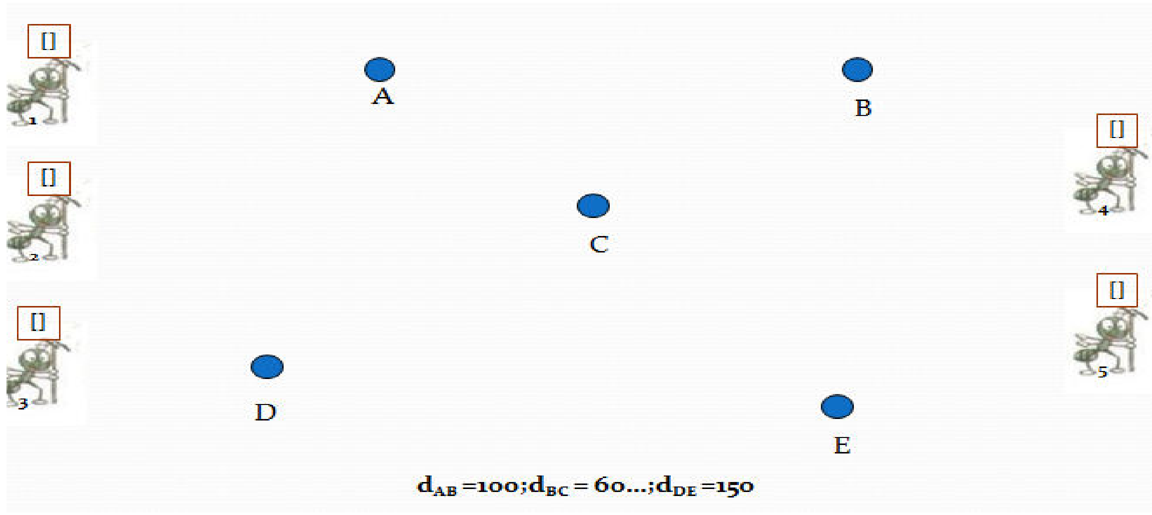


Fig 1.3 Travelling salesman problem

First iteration: In this example there is an ant on every node. In first step each ant moves randomly to any node. Let us consider the ant at A moves from node A to E, and similarly ant of B moves from B to C, and so the other ants as shown in the first iteration. While moving, the ants lie a chemical substance named pheromone on their way so that other ants can find their way by smelling.

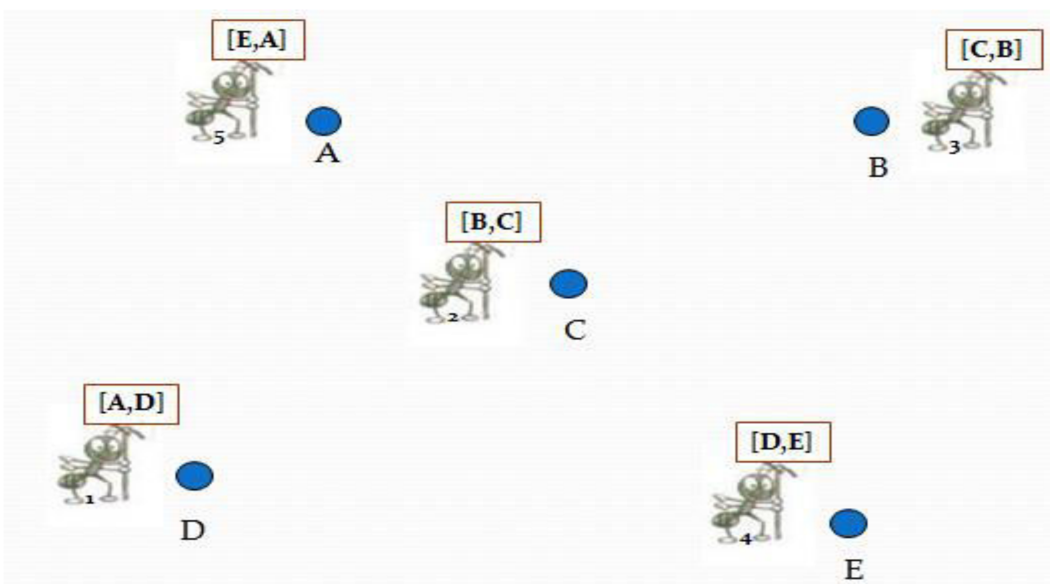


Fig 1.4 First iteration of travelling salesman problem

Second iteration: Similar to first iteration, movement of ants in 2nd iteration is as shown in figure 1.5 as the ant which was at E is moved to A and the ant which was at A is moved to B and so on same leaving pheromone on their way.

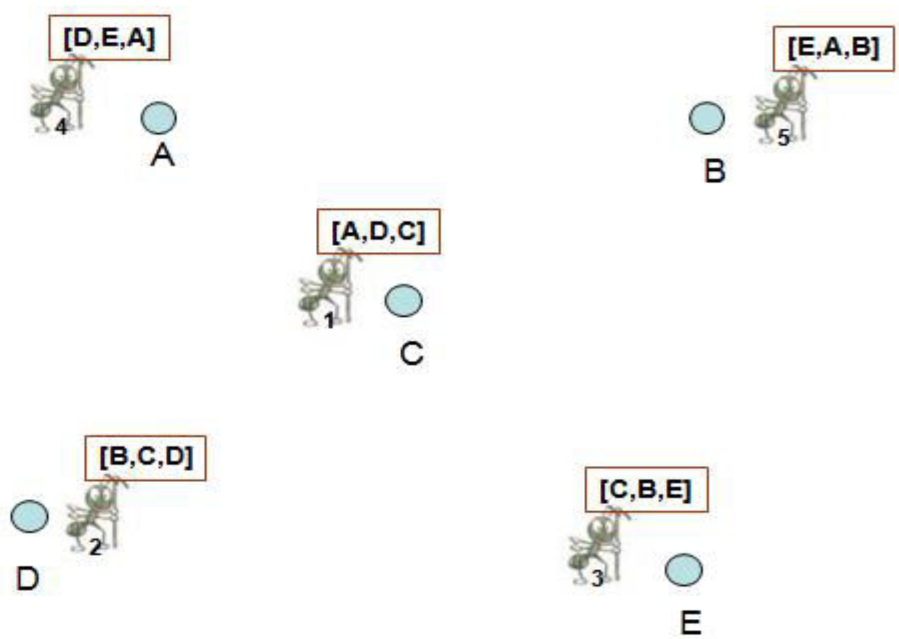


Figure 1.5 2nd iteration of travelling salesman problem

3th iteration: Same as above now ants move to the forth node having highest probability ,The ant at node D moves to A, ant at node B moves to C and so on as shown in figure 1.6.

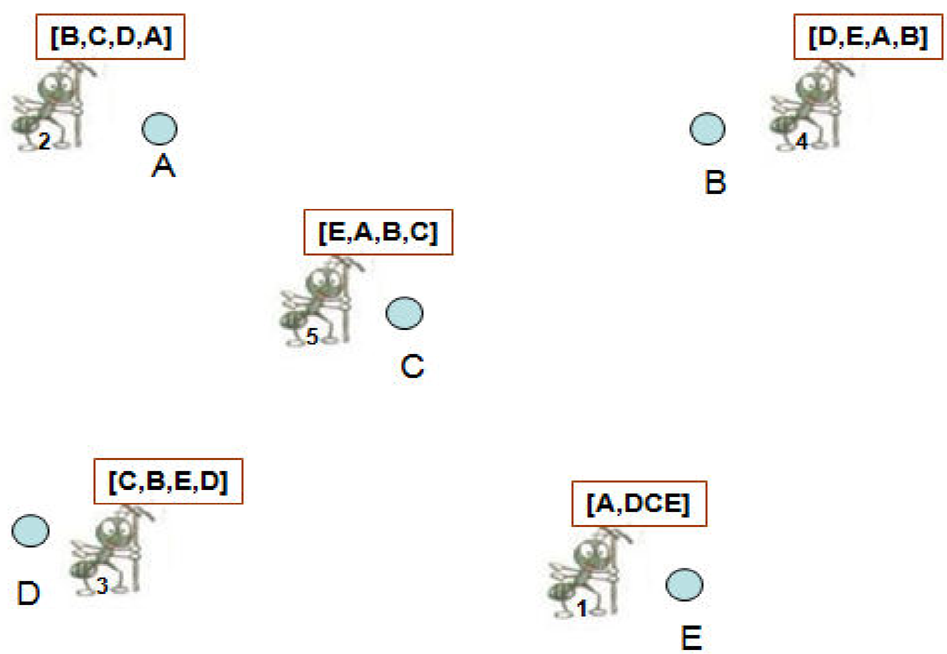


Figure1.6 third iteration of travelling salesman problem

4th iteration: Similarly at the end each ant has a path from source to destination. But the shortest path is the one having highest quantity of pheromone

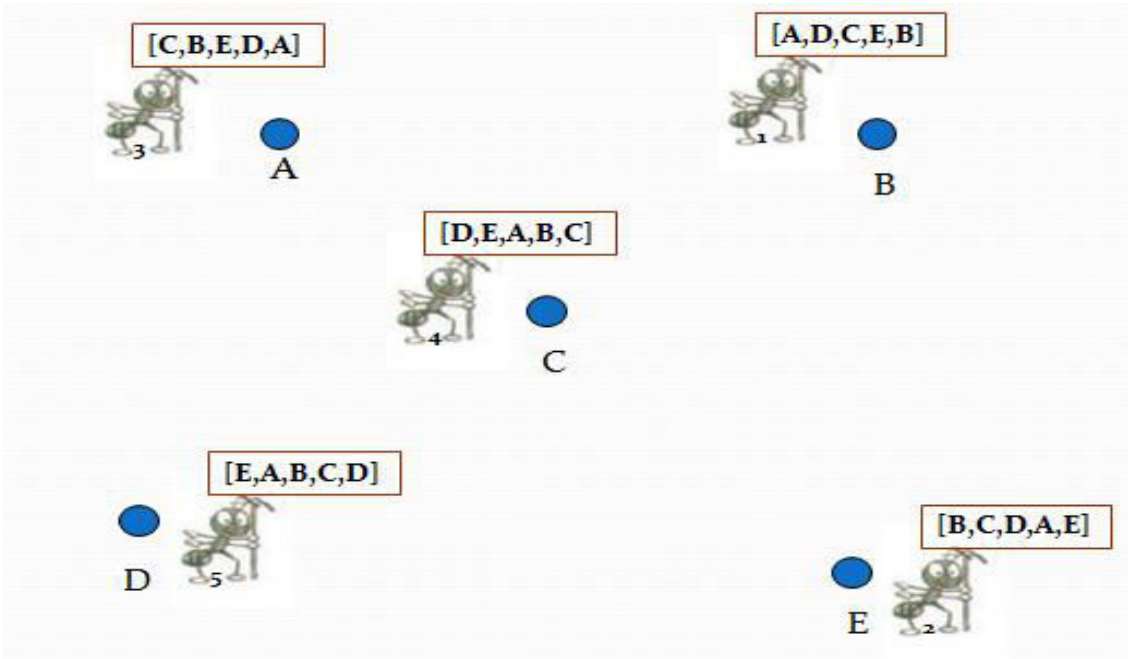


Figure 1.7 Last iteration of travelling salesman problem

1.4.2 Working of Artificial ants

1.4.2.1 Path selection formula

The ants are able to find shortest path on the basis of (i) the pheromone laid on ground by other ants of same colony which represents the experience of the colony and (ii) memory and heuristic information which represents useful knowledge about the particular problem the ants are solving[8, 7]. Therefore, probability of next path chosen by kth ant is given by Marco Derigo [29]:

$$p_k(r,s) = \begin{cases} \frac{[\tau(r,s)].[\eta(r,s)]^{\beta}}{\sum_{u \neq J_k(r)}[\tau(r,u)].[\eta(r,u)]^{\beta}} & \text{if } s \in j_k(r) \\ 0, & \text{otherwise} \end{cases} \quad \ldots (1)$$

where,

- τ is pheromone level on path (i, j) of graph,
- η is heuristic function,
- J(r) is array list (memory of ant) containing all the trails that the ant has already passed and must not be chosen again,

- and η heuristic function is used to search for optimal path with the help of useful information, and the array list stores the previous paths visited in the memory.

1.4.2.2 Updating of pheromone trail

Updating the trail is performed when ants complete their search of food source. The updating of pheromone is achieved by using

$$\tau(r,s) \leftarrow (1-\rho).\tau(r,s) + \sum_{k=1}^{m} \Delta\tau_k(r,s) \qquad \ldots (2)$$

where the value of ρ lies between 0 to 1, and

$$\Delta\tau_k(r,s) = \begin{cases} \frac{1}{L}, & \text{if}(r,s) \in \text{tour taken by ant k} \\ 0, & \text{otherwise} \end{cases}$$

L is the distance travelled by the best ant [29].

1.5 BEE COLONY OPTIMIZATION (BCO)

BCO is also based on swarm intelligence. This algorithm gets inspiration from the behavior of real bees that dynamically allocate the tasks. The honey bees have pictorial memories, space-age sensory and navigation systems, possibly even insight skills, group decision making process which help them in selecting new sites. They perform tasks such as queen and brood tending, storing, retrieving and distributing honey and pollen, communication and foraging. These types of characteristics are the core concepts for modeling the smart behaviors of bees. For explaining this algorithm firstly it is needed to note the behavior of bees. Bees are social insects living as colonies [6].

Types of bees in the colony are:

- Queen
- Drones and
- Workers.

Queen bee: The life of Queen Bee is several years. She is the mother of all other members because only she has the egg laying property. For only once in its life queen mates and fertilizes for two or more years by the sperms stored in the mating [24]. After consumption of the sperms, she produces unfertilized eggs and then only one daughter of them is selected for the post of queen for keeping egg-laying. A laid egg hatches into larva, pupate, adult bee, respectively. New eggs are produced by the queen when the colony is lack of food sources. If the colony becomes too crowded, it stops laying. 2,000 eggs a day are laid by a healthy queen bee.

Drones: The unfertilized eggs produce male bees called Drones They are called father of colony. Fertilized eggs produce queen bees and worker bees which are fed differently as larva. The drone's age is not more than 6 months [24]. Their strength increases in summer times. The main job of a drone is to fertilize a new queen. Drone's life ends as they mate with the queen.

Workers: The main task of worker bees is to find food source, collect food from them and then come back to the source. Workers used to make the wax cells for the queen into which it lays its eggs and feed the larvae, drones and queen by special material or secretion of their salivary glands. The tasks of worker bees go till their age and the needs of the colony. In the second part of its life, it works as a forager by initially leaving the hive for short flights in order to find the position of the hive and the surroundings topology. The life of worker bees is 6 weeks during summer times and 4–9 months during the winter times [24].

Following are working of Bees:

- Foraging
- Dance
- Nest Site Selection

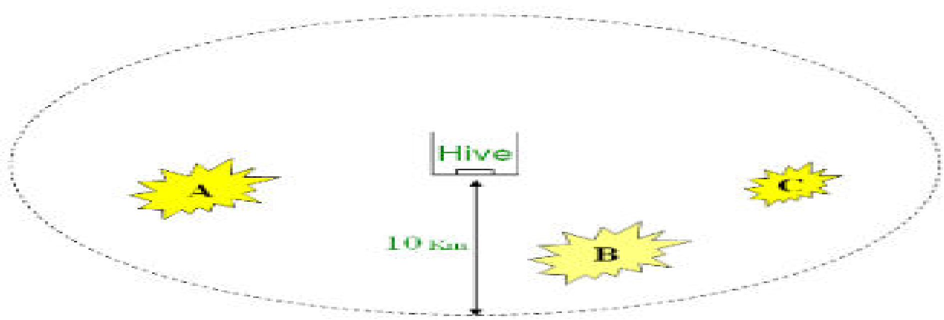


Figure1.8 Colony of Bee

Foraging: The most significant job in the hive is foraging. This course starts when the foragers leave the hive for searching a new food source to collect nectar. After discovering a flower for herself, the bee collects the nectar in her stomach [30]. On the basis of surroundings such as wealth of the flower and the distance of the flower to the net, the bee fill her stomach in about 30–120min and in their honey making procedure which starts with the emission of an enzyme on the nectar in her stomach. , the bee unloads the nectar to empty honeycomb cells after coming back to the hive and some extra substances are added in order to avoid the fermentation and the bacterial attacks. Filled cells with the honey and enzymes are covered by wax.

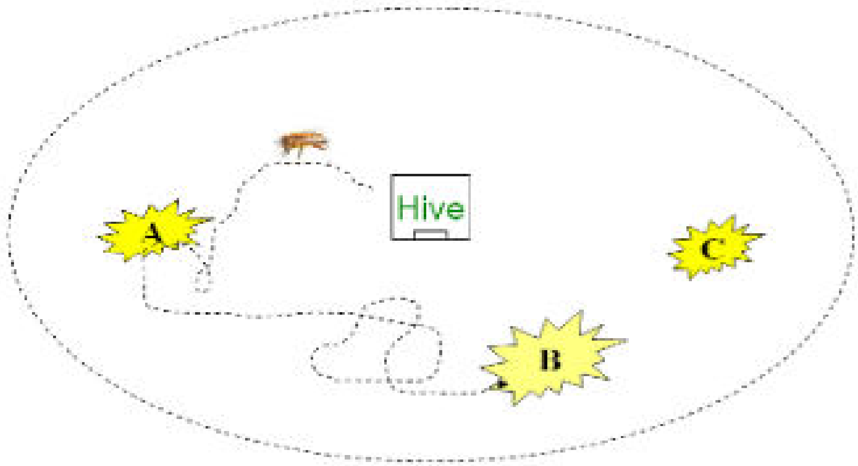


Figure 1.9 Foraging of bees

Dance: The forager bees perform a special kind of dance called waggle dance after unloading their nectar on the area of the comb so as to share its knowledge about the food source that is, what amount of nectar is, in which direction it and what is the distance between source to destination. After this they recruit other bees for searching the rich food source. Three types of dance forms are used in the net by the forager bees that are waggle dance, tremble dance and round dance. It is round dance if the distance from source to sink is not more than 100m. But for long distances waggle dance is performed. Waggle dance also gives the information about the direction of the food source but the round dance does not give this kind of information. For long delay in unloading nectar perceived by foraging bees, tremble dance is performed.

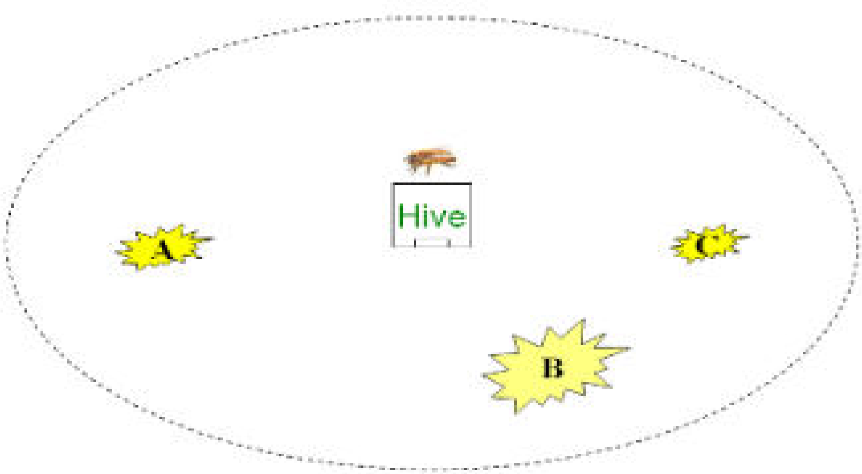


Figure1.10 Bee gets back to hive

Bees return to the hive and perform waggle dance which gives the direction in which the food source is available.

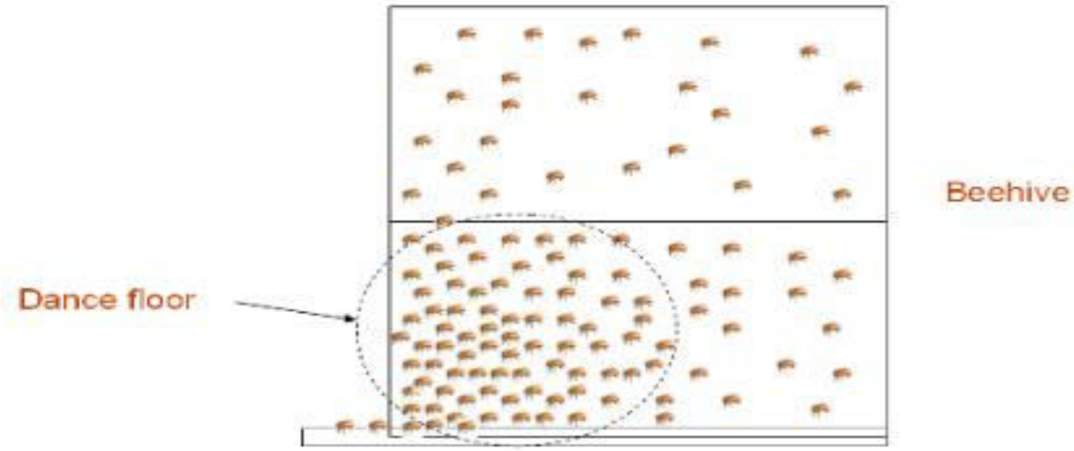


Figure 1.11 Waggle dance

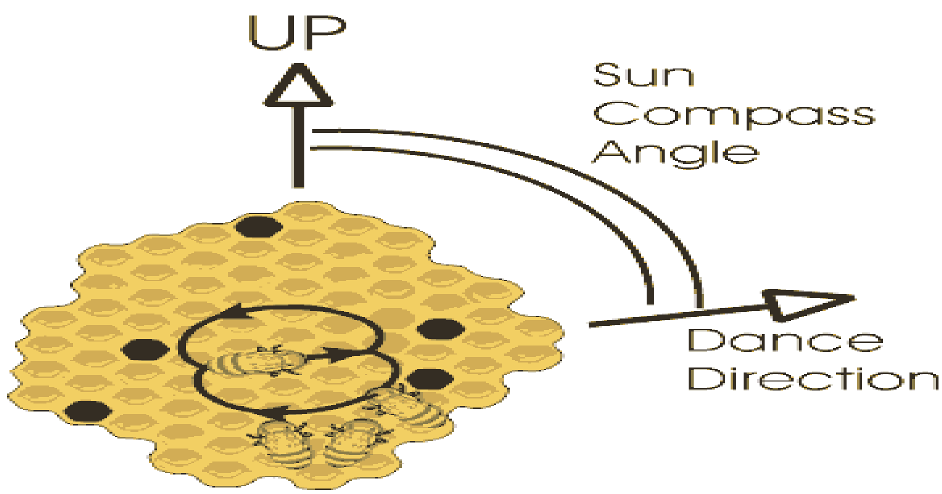


Figure 1.12 Direction of dance

Nest site selection: According to the factors like size of cavity to hold combs, tightness of the cavity, weather conditions and the construction, the bees select the next site to visit. For performing this task, many scout bees work in parallel for exploring the rich nectar nest sites and give this information to the other scout bees by showing their dance. The best bee is selected on the basis of dance of which the strength is proportional to the site quality. The scout bee moves to the other food source if that food source's bee's performance is better than the earlier one. The forager bees use a map-like organization of spatial memory for homing, food source search flights. This association is based on the formulation of two experienced vectors, or on viewpoints and landmarks. First one is that bees use stimuli obtained during their flights. The second one is they encode the spatial information in their dances into their map-like spatial memory

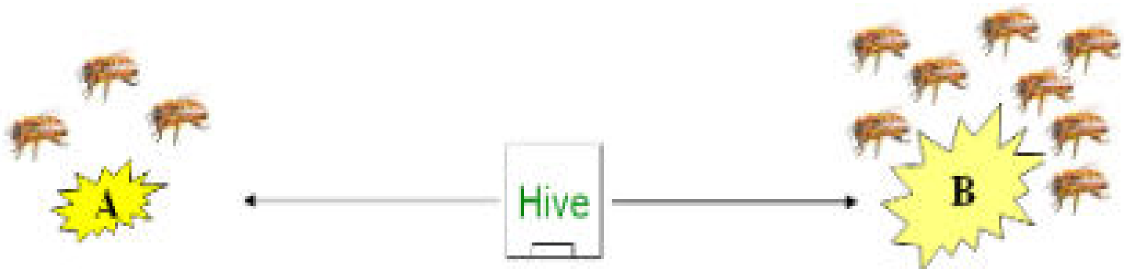


Figure1.13 All bees follow the same food source

In this dissertation scheduling is discussed to schedule the packets transferring from one node to other which is explained below.

1.6 Scheduling

Scheduling is a way by which multiple threads and processes get access to system resources like process time, communication bandwidth etc. Scheduling is mainly done to load balance and to share system resources effectively. It is concerned with latency, throughput and response time

An ideal scheduler should have the following properties [9, 34, 22].

- The implementation should be easy.
- It should be fair in resource sharing.
- Provide complete protection against starvation.
- Provide guaranteed performance bounds.

The factors under consideration are:

- *Throughput* – It is total number of processes that complete their execution per unit time.
- *Turnaround time* - Complete time between submission of a process and its completion.
- *Response time* – Difference of time it takes from when a request was submitted until the first response is produced.
- *Fairness* - Equal CPU time is distributed to each process
- *Waiting Time* - The time for which the process remains in the ready queue.

In real, these quality parameters often clash (e.g. throughput versus latency) with each other, so the scheduler will implement a suitable compromise. Depending upon user's requirements and objectives, preference is given to any one of the above mentioned parameters. In real-time processes such as embedded systems for automatic control in industry (for example robotics), the scheduler need to make sure that processes can meet the deadlines; it is very critical for keeping the system secure. Scheduled tasks are distributed to remote devices across the network and are handled by an administrative back end.

In this dissertation we trying to improve the quality of MANETs by applying scheduling algorithm based on priority [10] with swarm intelligence over MANETs and got improved results which are explained later in chapter 4.

Chapter 2

Literature Survey

Derigo firstly introduced ACO based on the nature of real ants. By noticing the nature of the ants i.e. how they search food and go to destination, Marco developed an algorithm which is used in MANETs for searching optimal path from source to sink so that quality of service of MANETs is improved [29].

Garg *et al.* proposed an Ant Colony Optimization Routing for Mobile Ad-hoc Networks (MANETs); in this a new protocol based on ACO is designed named Ant Colony Optimization routing (AOCR) protocol for mobile multi-hop ad-hoc networks. Basically this protocol is based on Ant Colony part of swarm intelligence and used the procedure of ants by which they solve complicated problems using coordination. The authors provide a probabilistic technique in the form of AOCR for solving computational problems which are reduced to find good paths by graphs. Route maintenance is done periodically in this algorithm through data packets to retain optimal path. This routing protocol is well adaptive, efficient and scalable. The main focus of this protocol is on reducing overhead for routing [6].

Roy *et al.* proposed an approach named Ant Colony based Routing for Mobile Ad-Hoc Networks towards Improved Quality of Services; an algorithm for improving QoS in mobile ad hoc network is generated in this paper. The main motive of this algorithm is to minimize the complexity in mobile ad hoc network and to maintain the QoS features in the presence of dynamic topology, absence of centralized authority, time varying QoS requirements etc. For this they used both reactive and proactive routing protocols for path searching along with Ant Colony Optimization. This algorithm takes the concepts of both ACO and Optimized Link State Routing (OLSR) protocol and uses them to recognize multiple stable paths from starting to end nodes. This used the concept of ant-like mobile agents to establish multiple stable paths between source and destination nodes. The concept of forward ants and backward ants is also introduced in this algorithm. These ant agents are used to select multiple nodes and these nodes use ant agents to establish connectivity with intermediate nodes. This algorithm is mainly useful for multimedia communications in MANETs based on ACO framework [7].

Thivyalakshmi *et al.* designed an efficient algorithm for improving Qos in MANETs. In this they proposed Ant Colony Optimization Technique along with Swarm Intelligence Mechanism (PSO) to improve the Quality of Service parameters and also to enhance the MANET security [1].

Venkata *et al*. proposed multipath routing for mobile ad hoc networks based on Ant Colony for quality of service of MANETs. The authors proposed a quality of service enabled ant colony based multipath routing (QAMR) algorithm based on the foraging behavior of ant colony for selecting path and transmitting data. In this approach, the path is selected based on the stability of the nodes and the path preference probability. The authors have considered bandwidth, delay and hop count as the QoS parameters along with the stability of node, number of hops and path preference probability factors. QAMR uses bandwidth, hop count and delay to calculate multiple disjoint paths between source and destination to satisfy given QoS constraints. They also used NHA(next hope availability) which considers both mobility and energy factor to check for the availability of the next hop. The network lifetime is increased as the battery of node is also considered for path computation. ACO used during route discovery phase optimizes the path selection. The best path gets strengthened by deposition of pheromone substances on that link. Thus as the routes are selected completely satisfying stability and QoS constraints, it fully complies with QoS objectives [13].

Jianping Wang *et al*. designed a hybrid ant colony optimization routing algorithm for mobile ad hoc network called HOPNET. In this the proposed algorithm is based on nature of ants, consisting of local proactive route discovery within a node's neighborhood and reactive communication among the neighborhood nodes. The features of this algorithm are extracted from ZRP and DSR protocols and for simulation it used GlomoSim. This algorithm is compared with AODV routing protocol. The algorithm is also comparable with the well known hybrid routing algorithm, AntHocNet, which is not based on zone routing framework. From the results it is proved that HOPNET is highly scalable and adaptable for large networks compared to AntHocNet. The results also indicate that the selection of the zone radius has considerable impact on the delivery packet ratio and HOPNET performs significantly better than AntHocNet for high and low mobility.ZRP and DSR are the base protocols for this approach [25].

Karaboga has shown a comprehensive review of Artificial Bee Colony to provide a comprehensive survey of research on ABC. Karaboga proposed a population-based evolutionary and stochastic method named Artificial Bee Colony (ABC) algorithm in the year 2005. ABC algorithm is simple and very flexible as compared to other swarm based algorithms. Because of its good convergence properties, this method has become very popular and is widely used. A system of comparisons and descriptions is used to designate the importance of ABC algorithm, its enhancement, hybrid approaches and applications. A number of algorithms, applications based on ABC are compared in this algorithm [8].

Younes *et al.* proposed an improvement of the ACO meta-heuristic by using the method Artificial Bees Colony (ABC)" In this an ABC algorithm is united with ACO is developed to solve the Economic Power Dispatch (EPD) problem. In this algorithm Bee colony is used for finding best values and these best values are used to improve the ACO. A hybrid method for ABC-ACO suggested in this paper automatically and dynamically adjust the values of the pheromone factor weight and the weight of heuristic factor that have a strong influence on the articulation intensification diversification of research in ACO. The main purpose of this algorithm is to free the user of feeling difficulty in setting parameters in ACO along with improving performance of overall cost function. The feasibility of the proposed approach was also tested on IEEE 57-bus system and it gave better results. [11].

Wedde *et al*. generated An Energy Efficient Routing Algorithm for Mobile Ad Hoc Networks Inspired by Bee Behavior named BeeAdHoc. In this, a new energy efficient routing algorithm in MANETs is evolved. This algorithm got inspiration from the foraging principles of honey bees so named BeeAdHoc. Two types of bees are utilized in this named scouts and foragers for routing in MANETs. BeeAdHoc acts as reactive source routing algorithm and the consumption of energy is less as compared to existing state-of-the art routing algorithms. This algorithm is simple enough and mainly needs two types of messages for routing: scouts, which searches new routes on-demand to the destinations and forgers do the work of transporting data packets and along with evaluate the quality of the discovered routes. It has been proved by many simulations, which represent a extensive spectrum of network conditions, that BeeAdHoc delivers the better performance as compared to that of state-of-the-art algorithms (DSR, AODV, and DSDV etc) in terms of packet delivery ratio, delay and throughput etc but at a significantly smaller energy expenditure. [30].

Hoolimath *et al.* proposed Optimized TERMITE: A Bio-inspired Routing Algorithm for MANET's. In this a routing protocol called Optimized-Termite (Opt-Termite), based on the behavior of real termites is proposed. Opt-Termite used the concept of stigmergy for self-organization, so reduce the control packet overhead. The main concern of this algorithm is to balance load. In this technique less loaded mobile nodes are selected in terms of traffic to reach final destination. The movement of packets affect the routing information at each node gets and the routing table is updated accordingly [12].

Suguna *et al.* did a comparative analysis of Bee-Ant Colony Optimized Routing (BACOR) with existing routing protocols for scalable Mobile Ad Hoc Networks (MANETs) based on Pause Time. In this the authors proposed an approach based on swarm intelligence for on demand routing protocols. This algorithm takes the advantages of both Ant Colony and Bee Colony Optimization for foraging of bees to calculate the fitness function and ability of ants to solve complex problems and named this algorithm as Bee-Ant Optimized Routing (BAOCR). Based on the estimated delay, signal strength and residual energy of the nearby nodes BACOR finds the efficient node and sends the data packets through that node. The results of BACOR are compared with the other mobile routing tactics, and this algorithm provides better results in terms of quality parameters as delay signal strength and energy [14].

Shah *et al.* proposed an approach for improving Quality of Services in Mobile Ad Hoc Networks (MANETs) using Evolutionary Algorithms. They used the algorithms based on Evolutionary Algorithm as GA, ABC, PSO and ACO, and principle of Swarm Intelligence for solving computational and complex problems for improving QOS. Some known examples of swarms are bird flocks, fish schools and the colony of social insects such as terminates ants and bees. Use of these insect's intelligent – Evolutionary Algorithms (GA, PSO, ACO, ABC, etc) are developed to solve a computational and complex problem in different area [2].

Sharvani *et al.* used ant colony optimization for efficient stagnation avoidance for MANETs with local repair strategy In this they developed of an efficient routing algorithm called "Modified Termite algorithms" (MTA) for MANETs. MTA is developed by adopting efficient pheromone evaporation technique addresses to load balancing problems. By including QoS, efficient route maintenance, local repair strategy by prediction of node failures, the MTA is expected to enhance the performance of the network in terms of throughput, and reduction of End-to-end delay and routing overheads. The stagnation problems overcame by fine tuning of the Pheromone concentration based on node stability factor. The MTA implemented on MANET with fine tuning of pheromone concentration shows significant increase in the throughput ~76% with increase of Faulty Nodes as compared to termite algorithms [9].

Wang *et al.* proposed Packet Fair Queuing Algorithms for Wireless Networks. In this paper they introduced a concept of fair queuing in wireless networks, discussed the problems and difficulties to apply them to wireless networks, and provided a comprehensive survey of recent research on wireless fair queuing algorithms by noticing the rapid growth of wireless data services, the issues of providing different quality of services and fair channel access that have become more and more crucial. In wireless networks, there are many mature fair queuing algorithms that can provide fairness and bounded delay properties. However, they cannot be directly applied to wireless networks because of the busty and location-dependent error characteristics of wireless channels. So many fair queuing algorithms designed for wireless networks are proposed. This paper provided a comprehensive survey on current research in wireless fair queuing. [31]

Banchs *et al.* proposed a Distributed Weighted Fair Queuing in 802.11 Wireless LAN with Weighted Fair Queuing. They proposed a weighted fair queuing algorithm using advance DCF function of IEEE 802.11 in Wireless LAN. The simulation results of this algorithm showed that the proposed scheme is better to provide the desired bandwidth distribution independent of the flows aggressiveness and their willingness to transmit. It proposed the DWFQ (Distributed Weighted Fair Queuing) architecture for providing weighted fair queuing in wireless LAN that provides a flow with an average bandwidth proportional to its *weight*, but does not give any assurity of individual packets. The main aim of DWFQ is to keep the MAC protocol fully distributed, to minimize the migration effort from the current standard, and to provide backwards compatibility. This helps in getting preferred bandwidth distribution free from flows aggressiveness [32].

Kusyk *et al.* give a review of applications of Game Theory to Mobile Ad Hoc Networks: Node Spreading Potential Game. They presented a dispersed game theory for getting an even node distribution among the MANET nodes over a particular area. It used Nash Equilibrium for setting players. Nash equilibrium (NE) is a set of all players' strategies in which no individual player has an incentive in unilaterally changing her own strategy, assuming that all other players' strategies stay the same. This paper gives an idea of a node spreading potential game for uniform distribution of MANET nodes over an unknown geographical territory. The simulation results show that the game is effective in providing an acceptable level of area coverage while utilizing only the local information among the neighbors of each player. [26].

Zhao *et al.* proposed an approach using incompletely cooperative game theory in Mobile Ad Hoc Networks. In this, they proposed an algorithm based on incompletely cooperative game theory and used it to get better performance of wireless LANs. In this game, firstly, the current state of the game is detected by the channel by each node and then each node get changed its equilibrium strategy by changing its local contention parameters based on the estimated game state. And at last, the game is repeated again and again to get the best performance. Its simulation results show that the incompletely cooperative game increases system throughput, and decrease delay, jitter and packet-loss-rate. [28].

Jaramillo *et al.* proposed Optimal Scheduling for Fair Resource Allocation in Ad Hoc Networks with Elastic and Inelastic Traffic. After studding the problems of congestion control and scheduling in ad hoc wireless networks that have to support a mixture of best-effort and real-time traffic, It suggests a model for incorporating with the OS metrics of packets in the optimization framework that provides the solution in the form of both congestion control as well as scheduling algorithm for allocating resources fairly to achieve the fairness goal of both elastic and inelastic flows, and per-packet delay requirements of inelastic flows. Optimization and stochastic network theory have been successful in designing architectures for fair resource allocation to meet long-term throughput demands. Using a dual function approach it presented a decomposition of the problem into an online algorithm that is able to make optimal decisions while keeping the network stable and fulfilling the inelastic flow's QoS Constraints. [22].

Singh *et al.* proposed an approach for improving quality of services of Ad Hoc Network by working over link budgeting and delay elimination. In this approach they proposed the concept of link budget for MANETs. MANETs are networks capable of communicating in a set of small, low cost, low power sensing devices. A wireless sensor networks is totally based on the limiting factor i.e. energy consumption. A wireless sensor network consists of large number of sensor nodes distributed or scattered in particular network region. MANETs consist of node that is highly mobile, so in particular the range of the nodes is very important. Each device in a MANETs is free to move independently in any direction, and will therefore change its links to other devices frequently. The energy and the bandwidth of such path are of major concern. They provide formula and techniques based on quality parameters given in this paper that can be used to solve the problem regarding the transmission, bandwidth and energy. The lifetime of the network depends upon these parameters [15].

Chapter 3

Problem Formulation

The mobile adhoc networks are the infrastructure less networks constructed without any fixed infrastructure such as base station, tower, redirection switches and routers. Mobile adhoc networks are the temporary wireless networks. All the routing information is managed by the node itself. The main task of MANET nodes is to send the data and information from source to destination. The performance of MANETs is calculated in terms of how efficiently the nodes send data to the other nodes without any loss.

Many routing protocols are there to improve the efficiency such as reactive and proactive protocols but still there is need of improvement. The performance results of routing protocols AODV and AOMDV are very weak in terms of quality of service.

To overcome this problem a new approach is suggested based on swarm intelligence along with scheduling algorithm. This approach firstly calculates the best path for MANET nodes and then schedule packets at each node to avoid load and congestion. This approach is combined with the routing protocols i.e. with AODV and AOMDV and compared the results with the earlier ones. We get better throughput, packet delivery ratio and minimum packet loss than before.

Chapter 4

Proposed Work

In this dissertation a Swarm Intelligence-Based Scheduler is proposed for improving QoS of MANETs. Because the requirements for various applications changes time to time, the QoS routing approach may not be proactive. To fulfill the requirement of improvement of quality, the designed approach is divided into two phases. The first phase contains the routing algorithm to find the optimized distance from source to sink [25]. The second phase is scheduling of packets or data that are to be transferred from one node to another so that data loss should be minimized. This algorithm is different from others because the other algorithms focus only on routing decisions but here scheduling algorithm is also applied with routing that schedules the packets.

4.1 Phase 1: Routing algorithms

In MANETS the network routing is a critical problem since the network characteristics such as traffic load and network topology may vary rapidly and in a time varying nature. The given network can be represented as a construction graph where the vertices correspond to set of routers and links corresponds to the connectivity among routers in that network.

Swarm intelligence based Ant Colony Optimization (ACO) and Bee Colony Optimization (BCO) routing algorithms come under phase 1 that provide optimal path form source to destination.

4.1.1 Ant Colony Optimization for MANETs

Mobile ad hoc network routing is a difficult problem because network characteristics such as traffic load and network topology may vary stochastically and in a time varying nature. The distributed nature of network routing is well matched by the multi agent nature of ACO algorithms. The given network can be represented as a construction graph where the vertices correspond to set of routers and the links correspond to the connectivity among routers in that network. Here, network routing problem is finding a set of minimum cost path between nodes present in the corresponding graph representation which can be done easily by the ant colony algorithm.

The simple ant colony optimization (ACO) metaheuristic can be used to find the shortest path between a source node v_i and a destination node v_j on the graph G. Let $G = (V, E)$ be a connected graph with $n = |V|$ nodes. The pheromone concentration is an indication of the usage of the edge i, j. An ant located in node v_i uses pheromone τ_{ij} of node v_j and N_i to compute the probability of node v_j as next hop. Ni is the set of one-step neighbors of node *vi*. An ant located in node v_i uses pheromone $\tau_{i,j}$ of node v_j.

To find out routing path each ant selects the next node based on their probability of selection. The node having highest probability is selected for the next move. The probability of each node is calculated from equation *Eq* (1) in first chapter.

As discussed earlier each ant releases a chemical substance pheromone on their way. But after some time it starts evaporating. The evaporation rate of pheromone is calculated according to equation *Eq* (2). After this the path having highest pheromone value is selected which is the shortest path from staring node to destination. Because of some reasons ACO is used in MANETs

Reasons for ACO ensembles to ad hoc Networks:

I. *Dynamic topology:* This property is responsible for the bad performance of several routing algorithms in mobile multi-hop ad-hoc networks. The ant colony optimization meta-heuristic is based on agent systems and works with individual ants. This allows a high adaptation to the current topology of the network.

II. Local work: In contrast to other routing approaches, the ant colony optimization meta heuristic is based only on local information, i.e., no routing tables or other information blocks have to be transmitted to neighbors or to all nodes of the network. Ant Colony Optimized Routing for Mobile Ad hoc Networks (MANETs).

III. *Link quality:* It is possible to integrate the connection link quality into the computation of the pheromone concentration, especially into the evaporation process. This will improve the decision process with respect to the link quality. It is here important to notice, that the approach has to be modified so that nodes can also manipulate the pheromone concentration independent of the ants, i.e. data packets, for this a node has to monitor the link quality.

IV. *Support for multi-path:* Each node has a routing table with entries for all its neighbors, which contains also the pheromone concentration. The decision rule, to select the next node, is based on the pheromone concentration on the current node, which is provided for each possible link. Thus, the approach supports multipath routing.

4.1.2 Bee Colony Optimization for MANETs

In ABC algorithm, the actual position of a food source corresponds to a probable solution to the optimization problem and the amount of nectar found in a food source represents the quality (fitness) of the associated solution. The number of employed bees is equal to the number of onlooker bees and the number of employed bees is equal to the number of food sources. Each employed bee is assigned to an individual food source. This algorithm is an iterative process, starts by initializing all employed bees with randomly generated food sources i.e., solution represented by a D-dimensional real valued vector. For each iteration every employed bee finds a food source in the neighborhood of its current food source and evaluates its nectar amount i.e., fitness.BCO algorithm starts by associating bees randomly to all the food sources. All employed bees determine the food source and calculates its fitness value i.e. nectar amount. The food source can be represented by *A* means the position *ith* food source is represented as A_i = $(a_{i1},a_{i2},a_{i3}, \ ...a_{in})$. The nectar amount of position *i* is referred as $f\,(A_i)$. As the onlooker bees get information from employed bees, they move to the food source with probability *P* as in paper [24]

$$Pi= \frac{fit_i}{\sum_{k=1} fit_k} \qquad \dots (3)$$

where

$$fit_i = \frac{1}{1+f(A_i)} \qquad \dots (4)$$

The onlooker bees go on searching a food source in the locality of Ai by this

$$Ai(t+1) = Ai(t) + xij * u \qquad \dots (5)$$

where xij is the patch size of neighborhood for *jth* parameter of *ith* food source. Its value is calculated from the formula

$$xij \ = \ xij \ - \ xkj \qquad \dots (6)$$

where k is any random number, which is different from *i*. It is the solution in the neighborhood of i. The value of u lies between -1 to 1 and j is the dimension of the problem considered. The best fitness value is achieved by comparing new fitness value with the previous one; if the new fitness value is more than the earlier one then the bee will move to the latest food source and forgets the older one else it stays at its old position [25]. After completing this process all employed bees share their knowledge with onlooker bees, which select a food source having maximum probability P. By following this process the good food source will get maximum onlookers. Each bee searches food for a limited no of turns and that bee can become scout bee if the fitness value is not improved.

Steps for BCO

(i) Initialization.

(ii) Move the employed bees in search of food source and calculate the fitness value *i.e.* nectar amount.

(iii) Forward the onlookers onto the food source searched by employed bees and find their nectar amount.

(iv) Move the scout bees for finding the new food source.

(v) Store the best food source achieved so far.

(vi) If termination criterion or the no of turns are not completed then repeat from 2nd step.

(vii) Else stop the process and print the best food source obtained so far [11].

4.2 Phase 2: Scheduling Algorithm

In this scheduling algorithm an attempt has been made to fulfill all the QoS conditions. Here, main factor is weight [10]. We have prioritized queues of packets according to their weights which were assigned to them according to quantity and quality of packet bits they receive. The weight of each queue is calculated from the following method:

$$W_i = \frac{MaxWeight - S_i}{MaxWeight - MinWeight} \quad \dots (6)$$

where Si is the sum of all types of packets going to one queue. The different kinds of packets are acknowledgements, messages and data packets etc.

The probability of selection of queue is calculated by applying game theory. This can be done by calculating saddle point corresponding to each queue. Saddle point is the combination of strategies in which each player can find the highest possible payoff assuming the best possible play by the opponent. The saddle point helps in selecting the player [26] [28] Saddle points calculated for all queues are shown in table 1 where Q1,Q2,Q3 and Q4 are queues of which saddle points are to be calculated

Table 1: Saddle points

	Q1	Q2	Q3	Q4
Q1	0	0	1	-1
Q2	0	0	1	1
Q3	-1	-1	0	-1
Q4	1	-1	1	0

Here the saddle points are within {-1,0,1}. If a queue's weight is greater than the other one then its saddle point is 1, or if equal then 0 otherwise it will be -1. The weight of Q1 is equal to Q2 so saddle point is 0, weight of Q3 is greater than Q1 so 1 but the weight of Q4 is less than that of Q1 and Q3 so here the saddle point is -1 and so on. Next step is to calculate the sum of all individual rows. The row having maximum value will be the resulting queue to send packets first.

Steps of scheduling algorithm

(i) Take number of queues

(ii) Assign weight according to the type of packets.

(iii) Calculate probabilities of the queues using game theory by finding the saddle point.

(iv) Select one queue having high saddle point to send the packets to next node.

(v) Repeat the process until all queues send their packets.

The queue having highest probability will be selected for sending the packets. After this the queue having next highest probability will send the packets and so on. This is the way all the queues send their corresponding packets to the other nodes in a scheduled manner with minimum loss and congestion. This scheduler works at each node in MANETs.

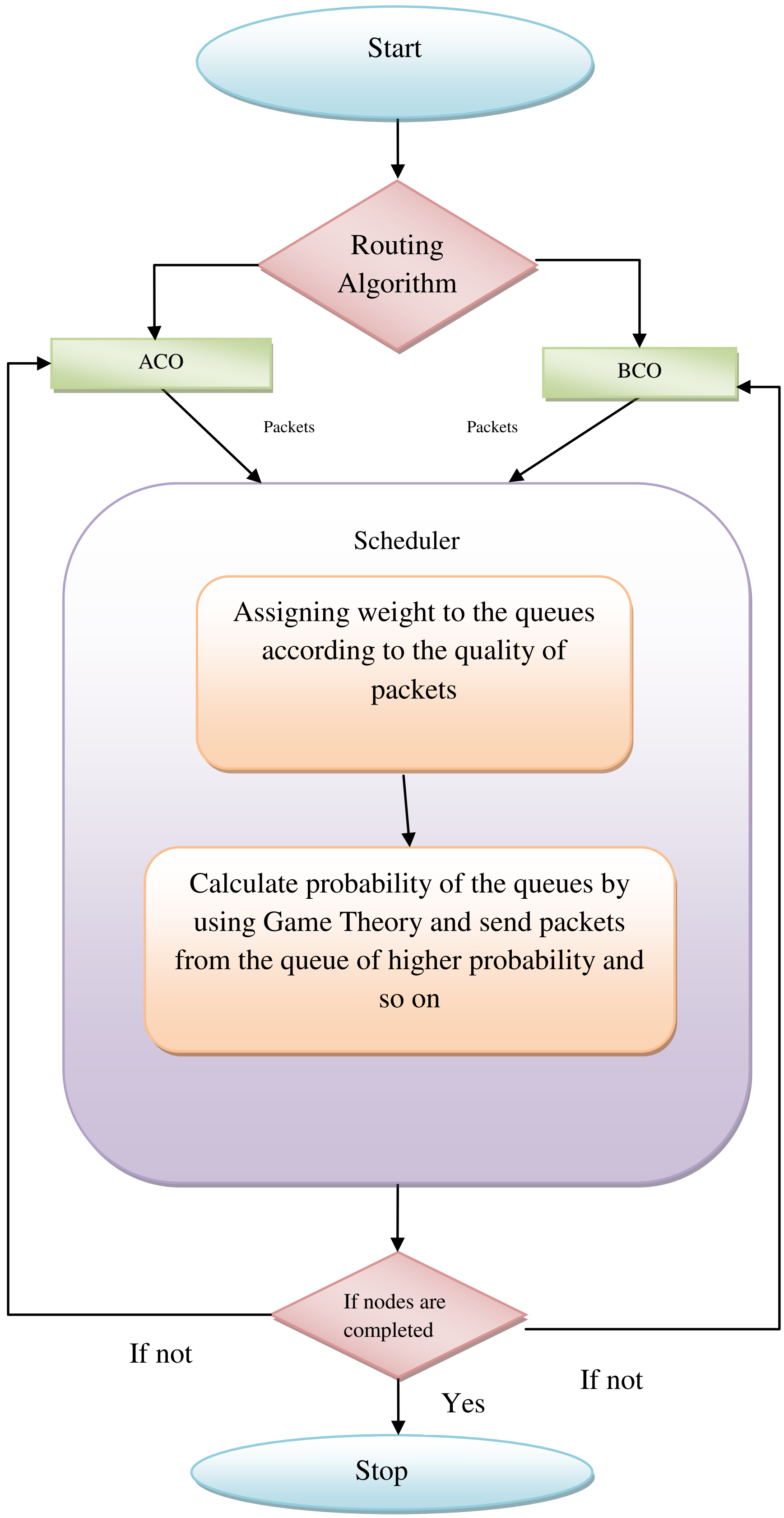


Figure 4.1 Flow chart of Routing Algorithm

4.3 Working with ns2

4.3.1 Introduction to Network Simulator

Network simulator is the standard simulator used for designing of new protocols as per considered as best one to give the correct results. NS-2 is the simulator that works in three phases. One is the TCL scripting and other two are the high level language based such as C++ or JAVA [35] Ns began as a variant of the REAL network simulator in 1989 and has evolved substantially over the past few years. Currently ns development is support through DARPA with SAMAN and through NSF with CONSER, both in collaboration with other researchers including ACIRI. Ns has always included substantial contributions from other researchers, including wireless code from the UCB Daedelus and CMU Monarch projects and Sun Microsystems.

4.3.2 Beginning with NS-2

The first step is to install the NS-2. I have installed NS-2 on RED HAT LINUX. The presence of gcc compiler is must for working of the NS-2. The reason for this is that cross compiler is required to interpret the coding performed in NS-2; this compiler is provided through the gcc environment. To check the presence of gcc use the following command:

cd> gcc –v

Now if the message showing the configuration of gcc is displayed, this means that the gcc is correctly installed on the machine, in case there is error message then there is requirement of installing gcc again in correct format. After this installation, the second step is to install NS-2. This can be done by using simple Linux commands and then exploring the .install file present in the NS-2 folder. Then the installation will start. One can then validate or use it without validation. The command for validation is:

cd> ./validate

After validation, the next step is to check for ns if installed correctly. To do this, go to ns-2.3x folder and then type:

cd/ns2.3x>./ns

If, % sign appears then the NS-2 is installed correctly else you need to reinstall it. Here x denotes the version of the NS-2 used.

For working on MANETs in ns2 we have to create traffic files and scenario files for data traffic and for defining the position of mobile nodes

4.3.3 Scenario Files

The concept of setting up of the scenario for the animator in NS-2 is one of the key concern for getting the automated setting of the nodal structure thus, allowing the researcher to make the scenario that can handle the multi-functioning of the node. The scenario configured by the user is required to be placed as a tcl script file in the SCENE folder present under the MOBILITY folder present in the TCL folder. The scenario available in the NS-2 can be set for both wired and wireless mediums. The concept of MANETs is totally wireless, thus the explanation given below is regarding the configuration of the wireless medium. The scenario available is in two different versions- version 1 and version 2. The difference between both these versions is because of the number of parameters one can the user is required to be placed as a tcl script file in the SCENE folder present under the MOBILITY folder present in the TCL folder. The scenario available in the NS-2 can be set for both wired and wireless mediums. To create a scenario file we have to write the following command

```
setdest [-n num_of_nodes] [-p pausetime] [-s maxspeed]
[-t simtime] \[-x maxx] [-y maxy] > [outdir/movement-
file]
```

For example in this proposed algorithm 10,15,20,25, 30,50 nodes with p pause time 2, speed 30m/s, simulation time 1000 and area 1500x1500 are taken.

./setdest -n 10 -p 2.0 -s 30.0 -t 1000 -x 1500 -y 1500 > scen1500x1500-10-0-1000

There is no need to run the output. It is easy to redirect the output by opening file scen1500x1500-10-0-1000. The file gives the complete description about total number nodes, with their initial positions and their movement form initial position to final position as shown bellow.

```
$ns_ at 2.000000000000 "$node_(0) setdest
90.441179033457 44.896095544010
1.373556960010"
```

This line depicts that at 2.0 second the 0^{th} node starts to move at destination (90.44, 44.89) with the speed of 1.37m/s. We can explicitly change the direction and movement by changing these commands.

Directives for GOD are present as well in node-movement file. The General Operations Director (GOD) object is used to store global information about the state of the environment, network, or nodes that an omniscient observer would have, but that should not be made known to any participant in the simulation.

Currently, the god object is used only to store an array of the shortest number of hops required to reach from one node to another. The god object does not calculate this on the fly during simulation runs, since it can be quite time consuming. The information is loaded into the god object from the movement pattern file where lines of the form

```
$ns_ at 899.642 "$god_ set-dist 23 46 2"
```

are used to load the god object with the knowledge that the shortest path between node 23 and node 46 changed to 2 hops at time 899.642.

4.3.4 Traffic Files

The traffic generator is the file that is actually required to generate the traffic from the source towards the destination. The scenario set in the scenario file is incomplete without this traffic generator file as it decides the movement of data from the source towards the destination and this configuration also leads to movement of nodes at particular location following the mobility factor and thus moves the nodes to new set of location. This requires mobile transmission of data and this can be achieved through this tcl scripting. Again, this can be achieved through coding by an individual also, as described in the earlier portion of the paper but this can also be generated as a single command in NS-2. The whole command is to be performed in the terminal if opened in LINUX or in the Cygwin terminal if being worked upon the window based NS-2. The terminal is to be opened at the location cmu-scen-gen as it was opened in scenario configuration. The sample code for configuration of the traffic generator file is as follows

```
ns cbrgen.tcl [-type cbr|tcp] [-nn nodes] [-seed seed] [-
mc connections][-rate rate] > cbr-file
```

For example, a cbr connection files of 10 nodes, having minimum 5 connections, with a seed value of 1.0 and rate of 0.5 m/s. So command type is:

```
ns cbrgen.tcl -type cbr -nn 10 -seed 1 -mc 5 -rate 0.5 >
cbr-10-5-2-1
```

Where nn is the number of nodes, seed is the uniqueness of node, mc is minimum connection required and rate is the rate at which packets are sent from one node to others. The output of cbr-10-5-2-1 is shown bellow. It shows the connections of 2nd node to 3rd at time 82.5s with UDP connection setup between these nodes. Total UDP sources are 5 and total number of connections is 8 respectively

```
#
# 2 connecting to 3 at time 82.557023746220864
#
set udp_(0) [new Agent/UDP]
$ns_ attach-agent $node_(2) $udp_(0)
set null_(0) [new Agent/Null]
$ns_ attach-agent $node_(3) $null_(0)
set cbr_(0) [new Application/Traffic/CBR]
$cbr_(0) set packetSize_ 512
$cbr_(0) set interval_ 0.25
$cbr_(0) set random_ 1
$cbr_(0) set maxpkts_ 10000
$cbr_(0) attach-agent $udp_(0)
$ns_ connect $udp_(0) $null_(0)
$ns_ at 82.557023746220864 "$cbr_(0) start"
```

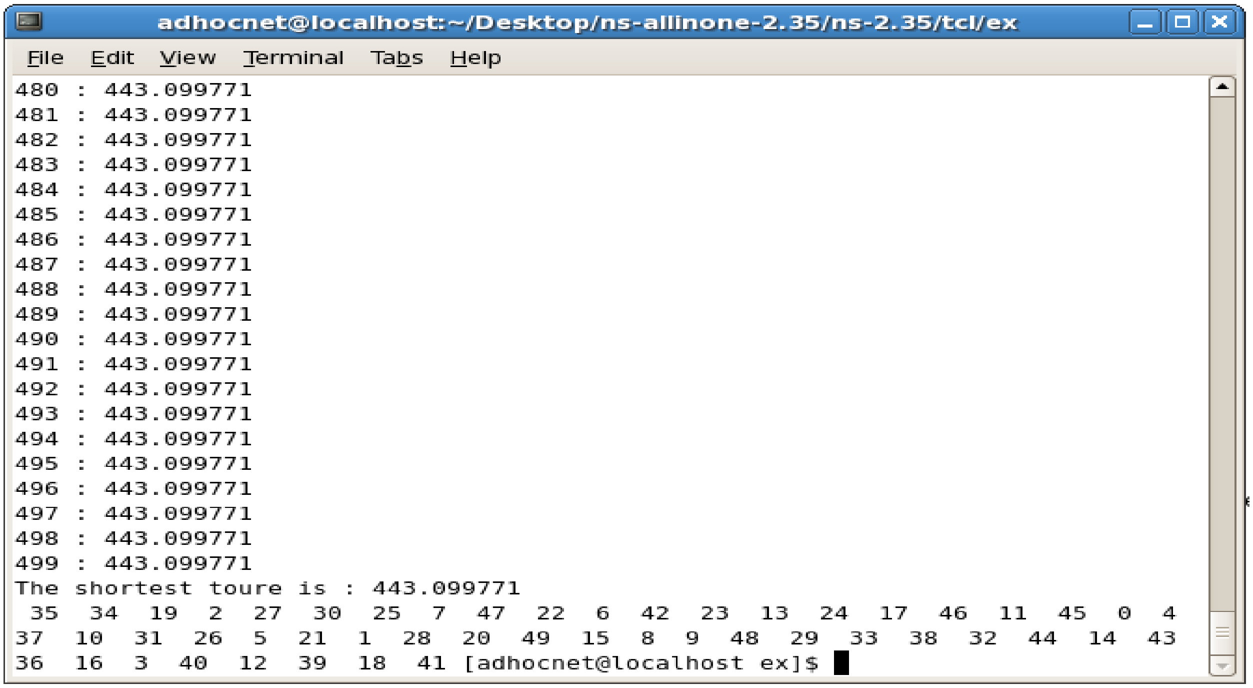
In proposed algorithm, cbr type traffic file is created having 10, 15, 20, 25, 30 and 50 nodes with mobility 10 and minimum connections 5 to 25 for each.

After setting of the scenario and the traffic generator for the MANETs animator, perform the following tcl scripting in the code file to include the scenario and the traffic generator file for working of the simulation.

```
set val(chan) Channel/WirelessChannel
set val(prop) Propagation/TwoRayGround
set val(netif) Phy/WirelessPhy
set val(mac) Mac/802_11
set val(ifq) Queue/DropTail/PriQueue
set val(ll) LL
set val(ant) Antenna/OmniAntenna
set val(x) 1500
set val(y) 1500
set val(ifqlen) 50
set val(seed) 1
set val(adhocRouting) AODV
set val(nn) 10
set val(cp) "../mobility/scene/ cbr-10-5-2-1"
set val(sc) "../mobility/scene/ scen-10-1500x1500-10-0-
1000"
set val(stop) 1000.0
```

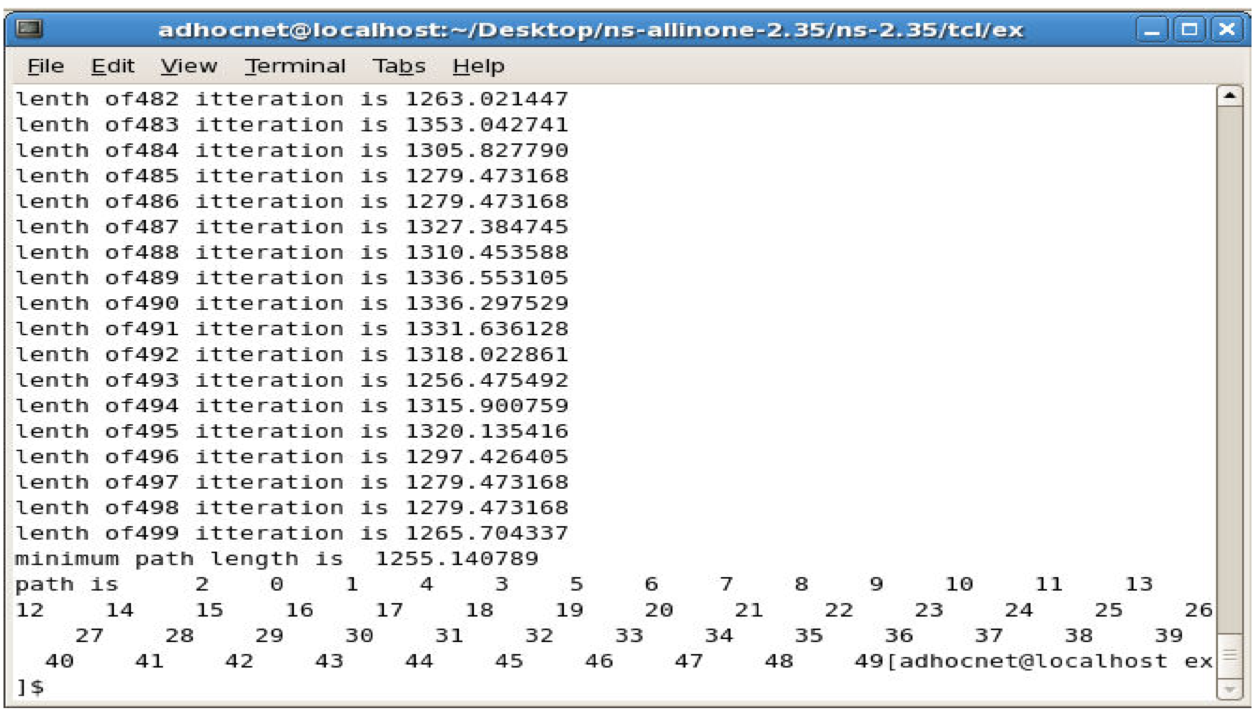
The codes of ACO and BCO with scheduling algorithm are integrated in this tcl script. After this, results are get by running this tcl script, Results are in the form of final output along with an animated nam file and trace file, which provide information about data packets.

Output of Ant Colony Optimization program with scheduling of packet



Snapshot 4.1 Output of ACO using scheduling

Output of Bee Colony Optimization program with scheduling of packets

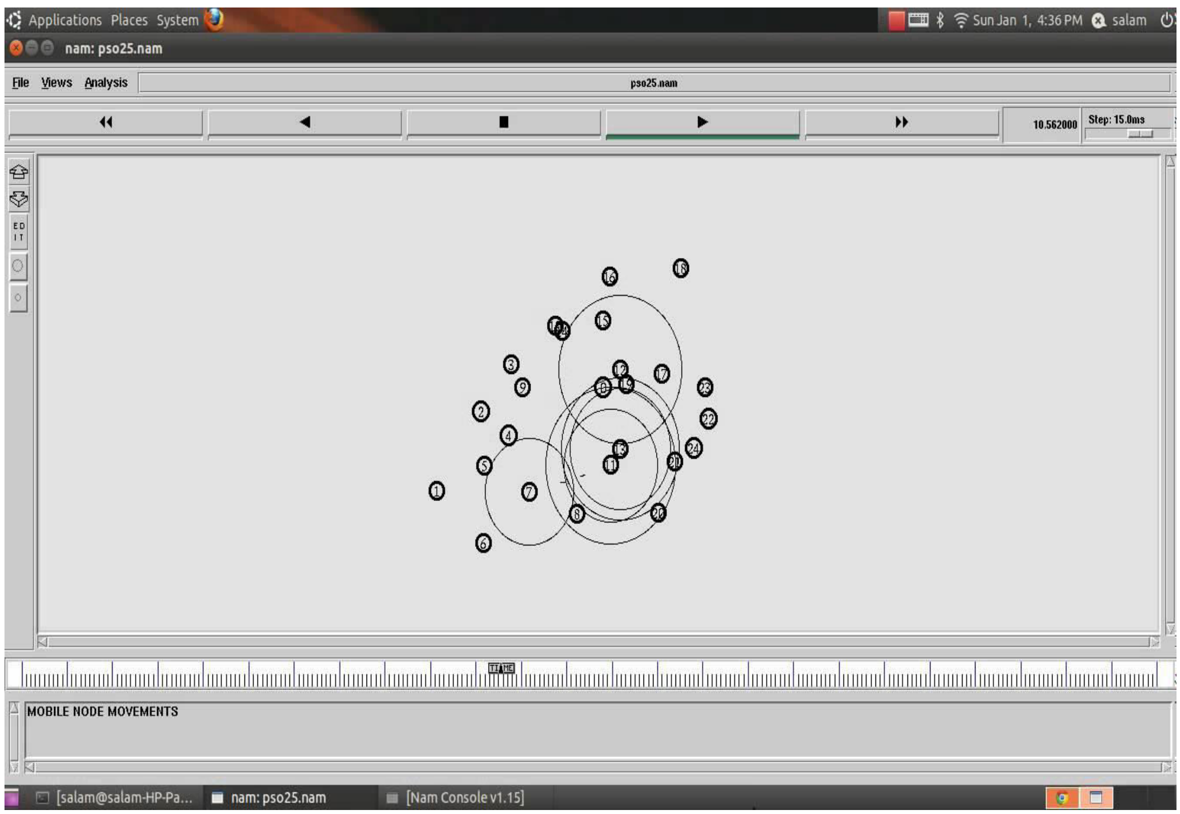


Snapshot 4.2 Output of BCO using scheduling

Nam file- Nam file is the visualization of node movements which shows how nodes interact with each other and how they send and receive packets. It has the following features.

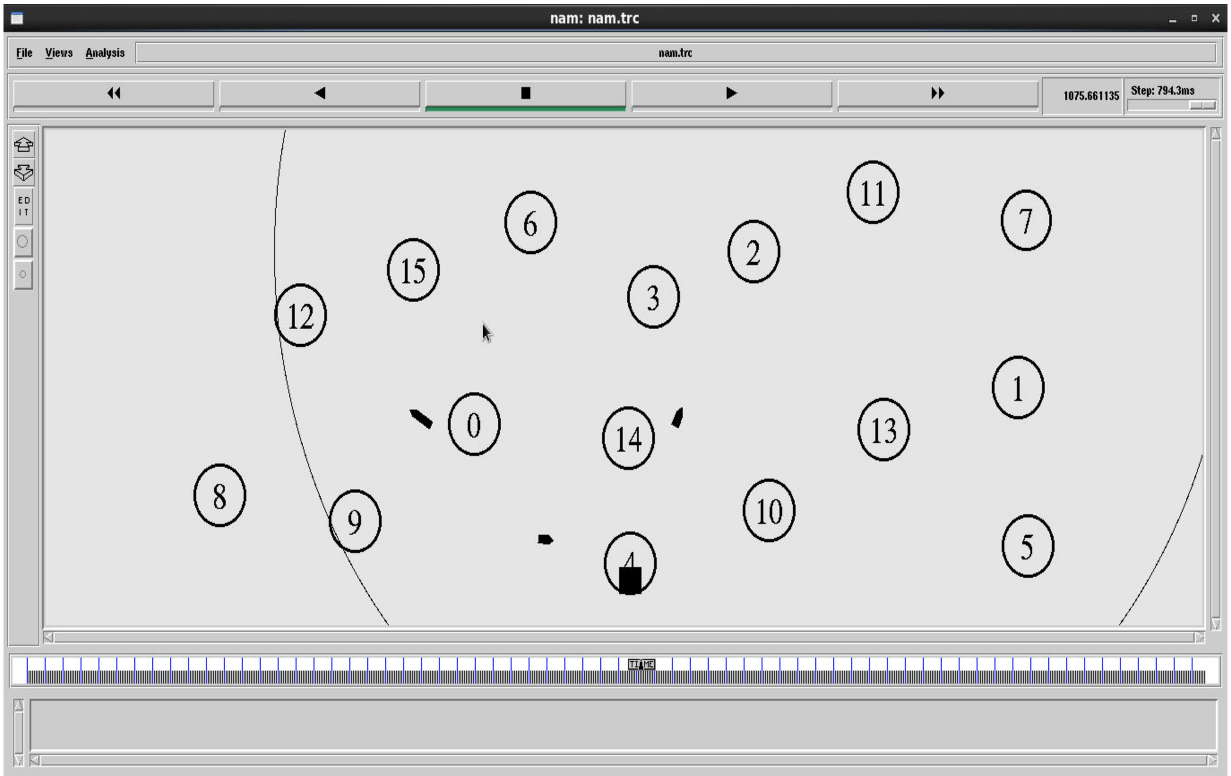
- Can be executed directly from a Tcl script
- Controls include play, stop ff, rw, pause, a display speed controller and a packet monitor facility.
- Presents information such as throughput, number packets on each link.
- Provides a drag and drop interface for creating topologies

Below snapshot 4.3 of nam file shows the movement of nodes. In this file the nodes broadcast requests to the other nodes to make a connection with them after which these nodes send their corresponding packets and data and get back the acknowledgement



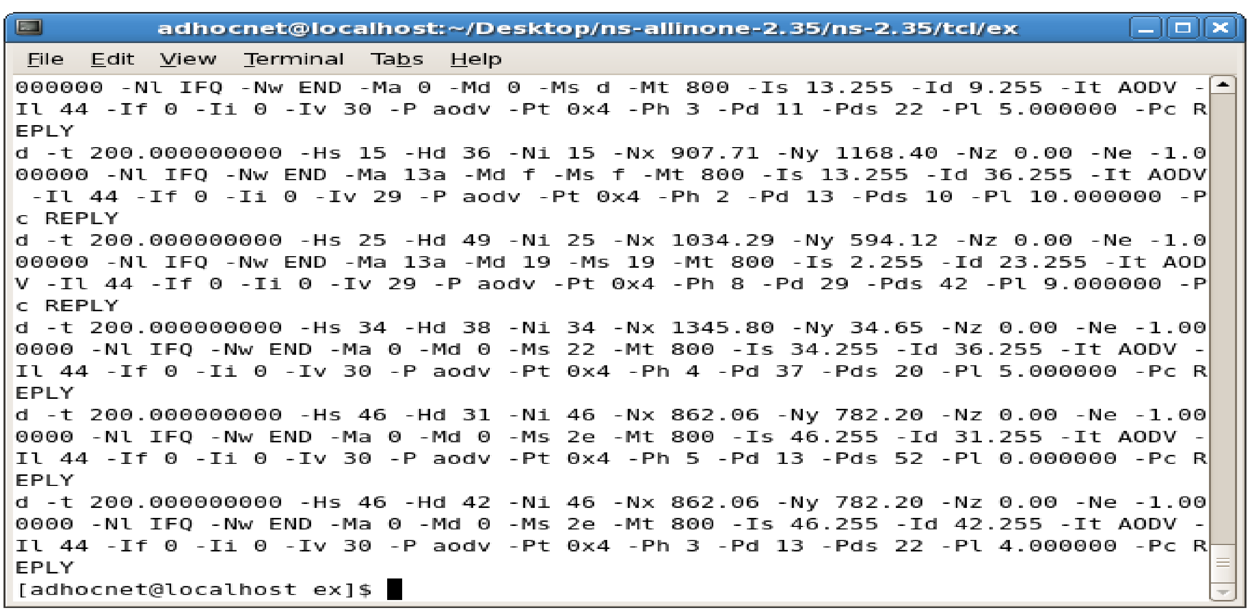
Snapshot 4.3 View1 of Nam animated file

The transfer of packets and acknowledgements at a particular moment are shown in snapshot 4.4. The nodes start sending packets after making connections through broadcasting. In following 4.4 snapshot the node 12 is sending packets to node 0 and get back acknowledge which is shown below. Similarly node 14 is sending packets to node 2 and so on.



Snapshot 4.3 View2 of Nam animated file

Trace File- Trace file is a .tr file same as nam file that gives the similar information of messages that are sent or received using command prompt. A trace file of of AODV protocol is shown below in snapshot 4.4.



```
adhocnet@localhost:~/Desktop/ns-allinone-2.35/ns-2.35/tcl/ex
File  Edit  View  Terminal  Tabs  Help
000000 -Nl IFQ -Nw END -Ma 0 -Md 0 -Ms d -Mt 800 -Is 13.255 -Id 9.255 -It AODV -
Il 44 -If 0 -Ii 0 -Iv 30 -P aodv -Pt 0x4 -Ph 3 -Pd 11 -Pds 22 -Pl 5.000000 -Pc R
EPLY
d -t 200.000000000 -Hs 15 -Hd 36 -Ni 15 -Nx 907.71 -Ny 1168.40 -Nz 0.00 -Ne -1.0
00000 -Nl IFQ -Nw END -Ma 13a -Md f -Ms f -Mt 800 -Is 13.255 -Id 36.255 -It AODV
 -Il 44 -If 0 -Ii 0 -Iv 29 -P aodv -Pt 0x4 -Ph 2 -Pd 13 -Pds 10 -Pl 10.000000 -P
c REPLY
d -t 200.000000000 -Hs 25 -Hd 49 -Ni 25 -Nx 1034.29 -Ny 594.12 -Nz 0.00 -Ne -1.0
00000 -Nl IFQ -Nw END -Ma 13a -Md 19 -Ms 19 -Mt 800 -Is 2.255 -Id 23.255 -It AOD
V -Il 44 -If 0 -Ii 0 -Iv 29 -P aodv -Pt 0x4 -Ph 8 -Pd 29 -Pds 42 -Pl 9.000000 -P
c REPLY
d -t 200.000000000 -Hs 34 -Hd 38 -Ni 34 -Nx 1345.80 -Ny 34.65 -Nz 0.00 -Ne -1.00
0000 -Nl IFQ -Nw END -Ma 0 -Md 0 -Ms 22 -Mt 800 -Is 34.255 -Id 36.255 -It AODV -
Il 44 -If 0 -Ii 0 -Iv 30 -P aodv -Pt 0x4 -Ph 4 -Pd 37 -Pds 20 -Pl 5.000000 -Pc R
EPLY
d -t 200.000000000 -Hs 46 -Hd 31 -Ni 46 -Nx 862.06 -Ny 782.20 -Nz 0.00 -Ne -1.00
0000 -Nl IFQ -Nw END -Ma 0 -Md 0 -Ms 2e -Mt 800 -Is 46.255 -Id 31.255 -It AODV -
Il 44 -If 0 -Ii 0 -Iv 30 -P aodv -Pt 0x4 -Ph 5 -Pd 13 -Pds 52 -Pl 0.000000 -Pc R
EPLY
d -t 200.000000000 -Hs 46 -Hd 42 -Ni 46 -Nx 862.06 -Ny 782.20 -Nz 0.00 -Ne -1.00
0000 -Nl IFQ -Nw END -Ma 0 -Md 0 -Ms 2e -Mt 800 -Is 46.255 -Id 42.255 -It AODV -
Il 44 -If 0 -Ii 0 -Iv 30 -P aodv -Pt 0x4 -Ph 3 -Pd 13 -Pds 22 -Pl 4.000000 -Pc R
EPLY
[adhocnet@localhost ex]$
```

Snapshot 4.4 Trace file

4.4. Simulations

4.4.1 Simulation and Performance Analysis

The performance of new routing algorithm has been compared with existing and known algorithms. The simulated traffic is Constant Bit Rate (cbr). Here, the proposed algorithm is applied on AODV and AOMDV routing protocols and comparison of results of Ant Colony and Bee Colony having scheduler has been shown. The proposed approach improves performance of quality metrics such as packet delay, packet delivery ratio and throughput. AODV and AOMDV perform best in terms of robustness when pause time network dynamics are down to 300s. As AODV and AOMDV use particular route during transmission with alteration in internet, the existing path will destroy that requires computation of new path which can cause network delay, But in the proposed approach this thing does not cause a problem because of different routes. All results are very close to the simulation scenarios. This shows that the algorithm creates much less routing overhead for all considered mobility scenarios. From results it is proved that the proposed algorithm is better in all aspects like PDR, throughput and packet loss etc.

Performance of the proposed approach is analyzed and compared with AODV and AOMDV by using the network simulator NS-2.35. For simulation, different numbers of nodes are considered with an area of 1500X 1500. The other parameters configured for simulation are shown in the table 2.

Table 2: Simulation Parameters

Parameters	Value
Version	2
Area	1500x1500
Nodes	10,15,20,25,30,50
Speed	30
Simulation time	1000
Minimum Connections	5-25
Connection Type	CBR
Mobility	10

Total no of Packets	15000
Protocols	AODV,AOMDV

4.2 EXPERIMENT RESULTS

The metrics considered for analysis are as follows.

- *PDR (Packet Delivery Ratio)*-It is the ratio of the no of packets delivered to the sum of all the packets. Calculated PDR for both AODV and AOMDV protocols using ACO and BCO with scheduler are shown the figures 4.1 and 4.2 below.

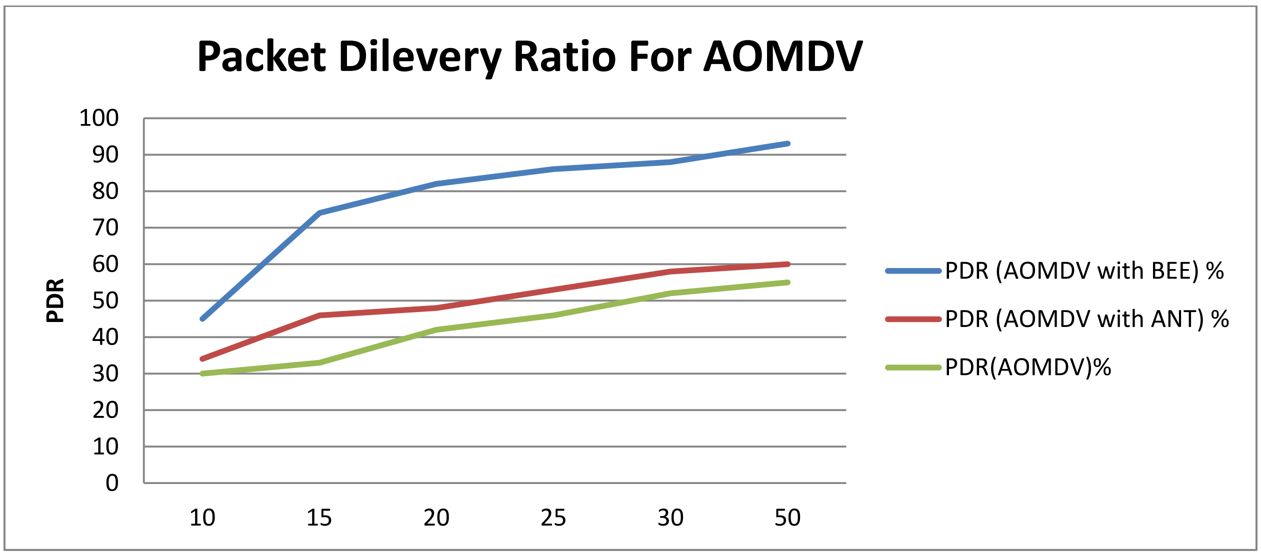


Figure 4.2 Packet Delivery Ratio(AOMDV- BEE Vs ANT)

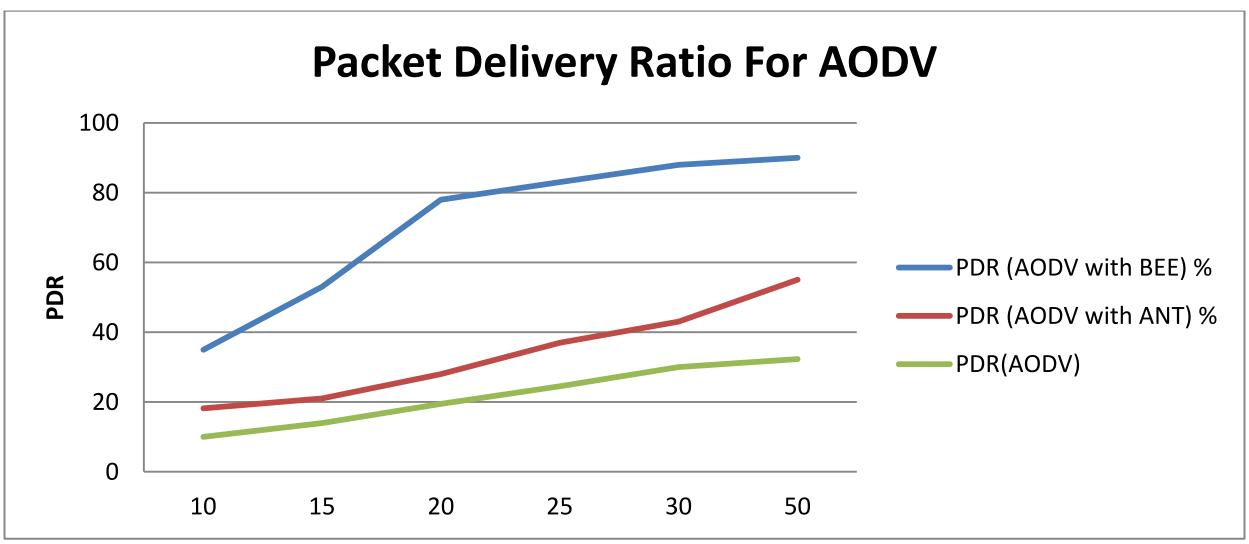


Figure 4.3 Packet Delivery Ratio (AODV – BEE Vs ANT)

Form these graphs it is clear that the packet delivery ratio is improved by using ACO and BCO because number of packets delivered in case of using ACO and BCO with the routing protocols AODV and AOMDV are much more as compare to when not using any algorithm with the routing protocols.

- *Packet Loss*- Packet loss is the total number of packets dropped during simulation. It is the difference of sent packets and received packets.

The no of packets lost in AODV and AOMDV protocols for different number of nodes are shown in figures 4.3 and 4.4

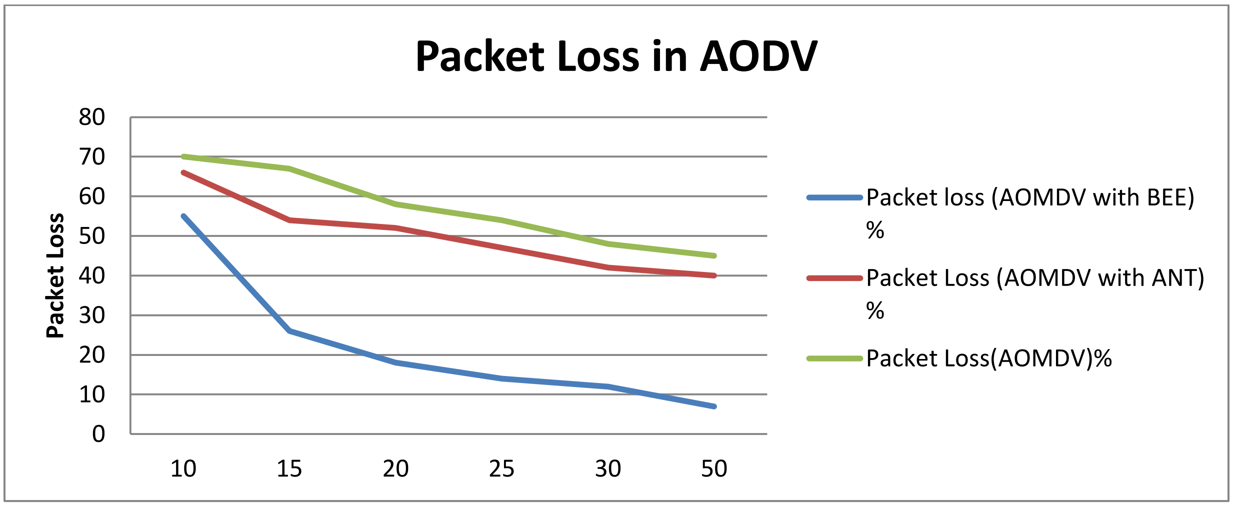


Figure4.4 packet Loss (AOMDV-BEE Vs ANT)

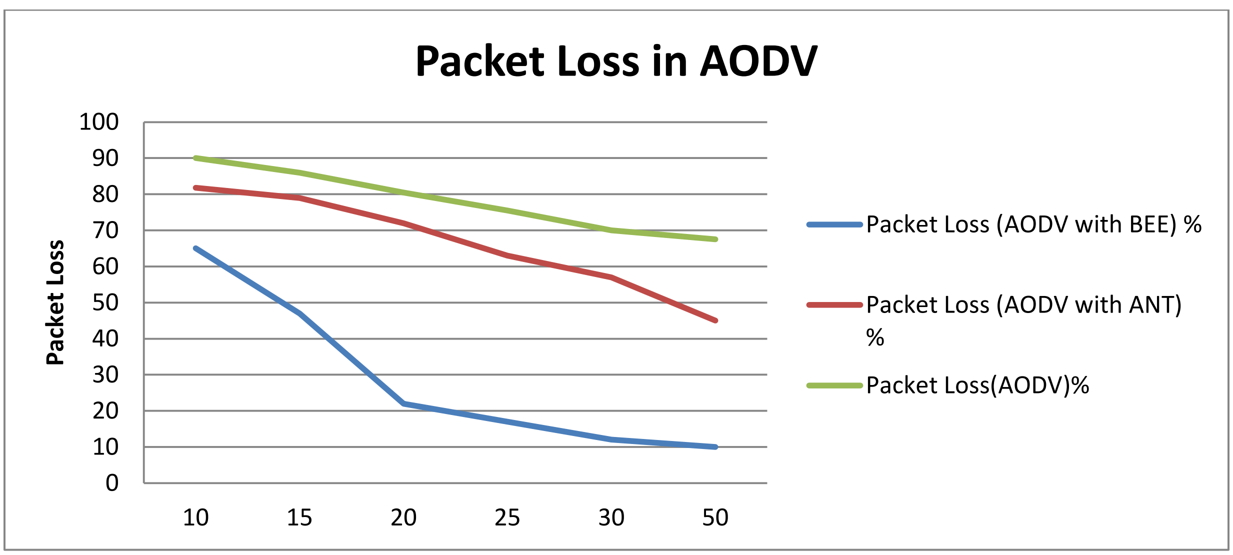


Figure 4.5 packet Loss (AODV-BEE Vs ANT)

Packet loss is minimum in case of Bee colony as compare to Ant, and maximum in case of simple using routing protocols AODV and AOMDV.

- *Throughput*- Throughput is the number of bits transferred per second between sources to the destination. Throughput for both AODV and AOMDV using ACO and BCO with scheduler which are shown in figures 4.5 and 4.6

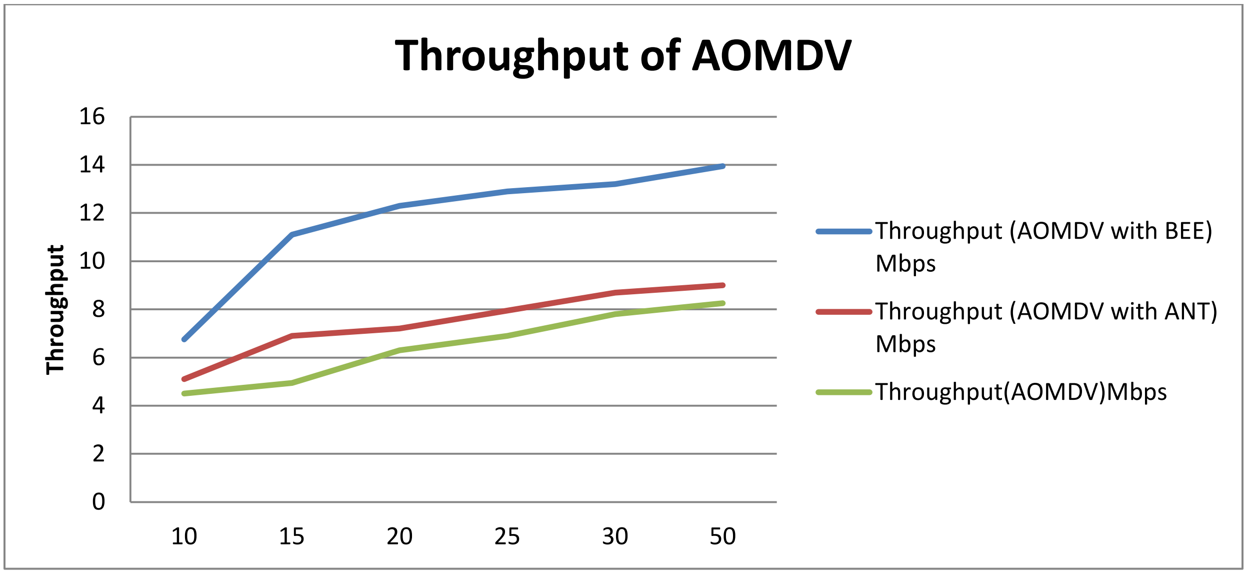


Figure 4.6 Throughput (AOMDV-BEE Vs ANT)

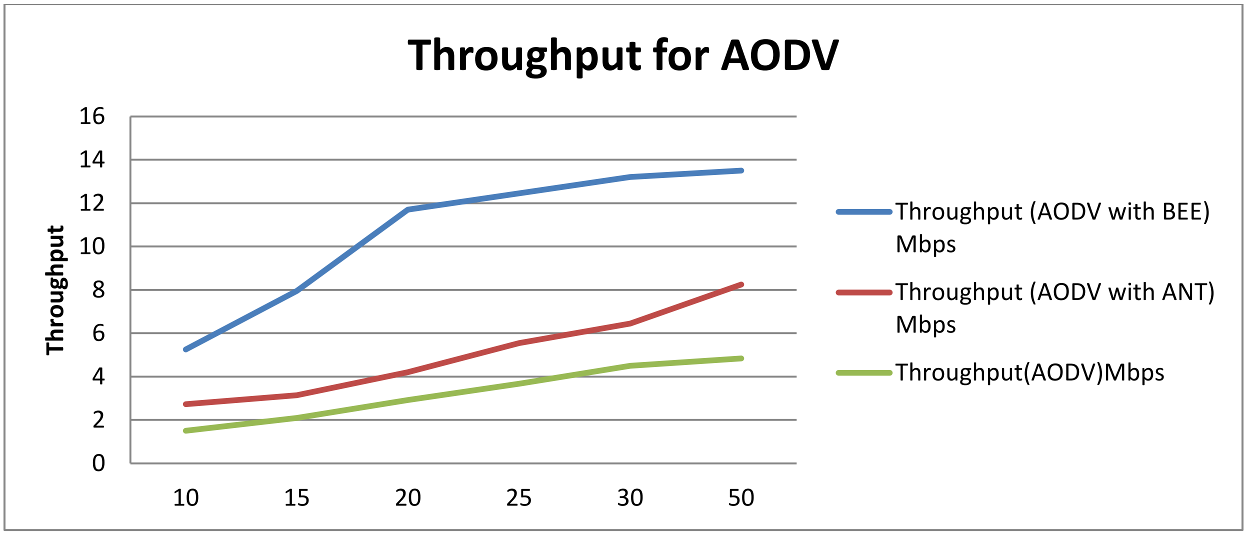


Figure 4.7 Throughput (AODV-BEE Vs ANT)

From the above graphs that the throughput is maximum in case of Bee colony as compare to Ant and minimum if using only AODV and AOMDV protocols

- *Running Time-* It is the time for which the program runs, we compared the running time of ACO and BCO and concluded that running time of BCO is better than ACO which is shown in figure 4.7.

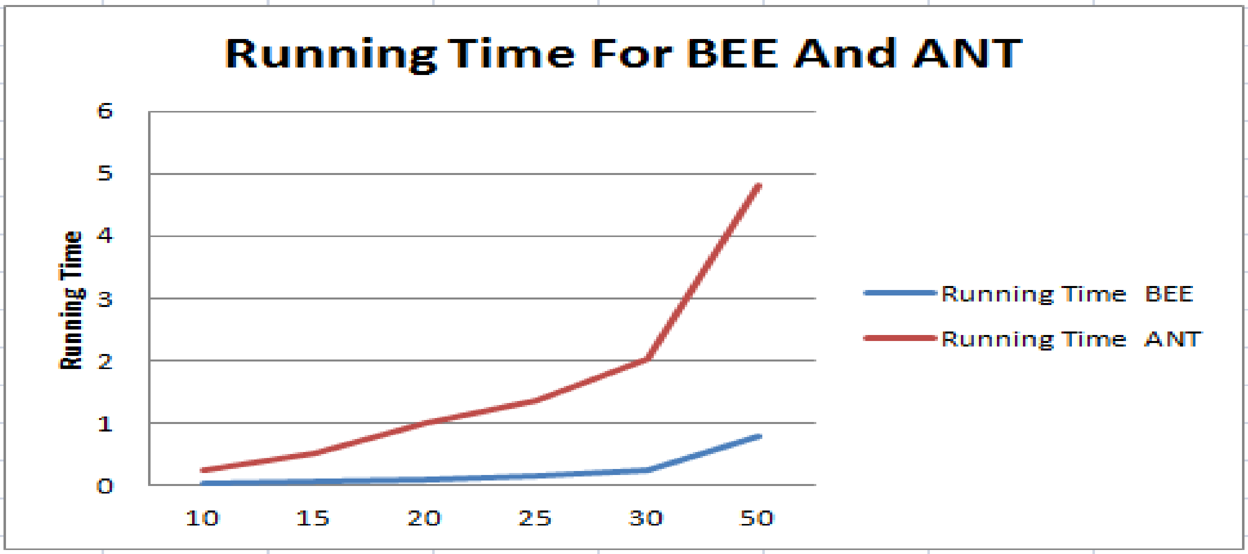


Figure 4.8 Running Time (BEE Vs ANT)

For different no. of nodes the running time of bee colony is less than that of Ant Colony , so from all above results, it is clear that Bee Colony Optimization is better than that of Ant Colony Optimization in every aspect whether it is PDR, throughput, packet loss or its running time.

Chapter 5

Conclusion and Future Scope

Quality of Service (QoS) support for Mobile Ad hoc Networks (MANETs) is an exigent task due to dynamic topology and limited resource. One of the major issues in MANET is routing due to the mobility of the nodes. For an efficient network formation, mobile nodes should be capable enough to utilize the available query resources.

For this, a scheduling algorithm with swarm intelligence to improve QoS of MANETs is proposed that means it combines the idea of both ACO and BCO with the scheduler that schedules the packets transfer from source to sink node. The routing algorithm satisfies more QoS requirements of the incoming traffic and at the same time reduces constrained resources consumption as much as possible. Results showed that by integrating scheduler with ACO and BCO, the performance of AODV and AOMDV protocols improves. Enhanced PDR, minimum packet loss and better throughput have been obtained.

The scheduling algorithm can individually be applied on network transmission process and its performance can be enhanced over other routing protocols also. The results showed that the proposed algorithm was capable of handle network latency and offers better network lifetime and it can promptly adapt to a changing network status.

The probability distribution changes after every iteration makes real time implementation of these algorithms a challenging task thus in future an efficient probability distribution model can be formulated that can optimally utilize the features of the proposed scheme in real time environment.

References

[1]. R.T.Thivyalakshmi, R.Srinivasan and G.S.Raj, "An efficient algorithm for improving quality of MANETs", International Journal of Scientific and Research Publications, Vol. 4 No. 2 (2014)

[2]. K. S. Shah and U. H. Soni "Improved Quality of Services in Mobile Ad Hoc Networks (MANETs) using Evolutionary Algorithms", IJSRD - International Journal for Scientific Research & Development Vol. 1 No.12 (2014)

[3]. D. Sensarma, and K Majumder. "An efficient ant based qos aware intelligent temporally ordered routing algorithm for MANETs" International Journal of Computer Networks & Communications (IJCNC) Vol.5 No.4 (2013)

[4]. Sivasakthi, D. Adhimuga, "Optimized Ant Routing For Improving Qos In MANETs." International Journal Of Engineering And Science Vol.3 No. 9 pp. 33-36 (2013)

[5]. J. Vats, R.K Singh and V. Sharma "Quality of Service Model for Load Balancing In Mobile Adhoc Networks", Proc. of the Intl. Conf. on Recent Trends in Computing and Communication Engineering (RTCCE) 2013

[6]. D. garg and P. Gohil, "Ant colony optimized routing for mobile adhoc networks (MANETs)", International Journal of Smart Sensors and Ad Hoc Networks (IJSSAN), Vol. 2 No. 3,4 (2012).

[7]. B. Roy, S. Banik, P. Dey, S. Sanyal and N. Chaki "Ant colony based routing for mobile ad-hoc networks towards improved quality of services." Journal of Emerging Trends in Computing and Information Sciences Vol. 3 No. 1 pp. 10-14 (2012)

[8]. D. Karaboga, B. Gorkemli, C. Ozturk and N. Karaboga "A comprehensive survey: artificial bee colony (ABC) algorithm and applications." Artificial Intelligence Review, pp. 1-37 (2012)

[9]. Sharvani, G. S., A. G. Ananth, and T. M. Rangaswamy. "Efficient stagnation avoidance for manets with local repair strategy using ant colony optimization" International Journal of Distributed and Parallel Systems (IJDPS) Vol. 3 No.5 pp. 123-137 (2012)

[10]. Ying, Lei, and Sanjay Shakkottai. "Scheduling in mobile ad hoc networks with topology and channel-state uncertainty" Automatic Control, IEEE Transactions on Scheduling in mobile ad hoc networks with topology and channel-state uncertainty Vol. 57 No. 10 pp. 2504-2517 (2012)

[11]. M. Younes and M. Maamar. "Improvement of the ACO meta-heuristic by using the method Artificial Bees Colony (ABC)" Przegląd Elektrotechniczny Selected full texts 88.5b, pp 174-178 (2012)

[12]. Hoolimath, P. K. G., M. Kiran, and GR Mohana Reddy. "Optimized Termite: A bio-inspired routing algorithm for MANET's." IEEE International Conference on Signal Processing and Communications (SPCOM), (2012)

[13]. P.V. Krishna, V Saritha, G Vedha, A Bhiwal and AS Chawla "Quality-of-service-enabled ant colony-based multipath routing for mobile ad hoc networks." IET communications Vol. 6 No.1 pp. 76-83 (2012)

[14]. Suguna, S. Kanimozhi, and S. Uma Maheswari. "Comparative Analysis of Bee-Ant Colony Optimized Routing (BACOR) with Existing Routing Protocols for Scalable Mobile Ad Hoc Networks (MANETs) based on Pause Time." IJCSNS International Journal of Computer Science and Network Security, Vol.12 No.4 (2012)

[15]. T. P. Singh, R.K. Singh, V. Sharma and M. Kaur, "Improving Quality of Services of Ad Hoc Network by Working over Link Budgeting and Delay Elimination." International Journal of Computer Applications, Vol. 40 No. 10, (2012)

[16]. V.Sharma, S.Amitoj, G. S.Sodhi, and T. Gupta, "An efficient approach for routing in wireless ad hoc network using artificial intelligence" International Journal on Computer Science and Engineering, Vol. 4, no. 2, (2012)

[17]. T. P. Singh, R. K. Singh, V. Sharma, and M. Kaur. "Link Budgeting over Selective Frequency Fading and Delay Elimination for Mobile Ad Hoc Networks" International Journal of Computer Networks and Wireless Communications (IJCNWC), Vol. 2 No.1 (2012)

[18]. T. P. Singh,., R. K. Singh and V. Sharma "Ad hoc on Demand delay Elimination (AODE) routing protocol for mobile Ad Hoc Networks and its performance comparison." Vol. 44 pp. 7519-7526 (2012)

[19]. A. K. Gupta, H. Sadawarti and A. K Verma., " MANET Routing Protocols Based on Ant Colony Optimization " International Journal of Modeling and Optimization, Vol. 2 No. 1 (2012)

[20]. M. Sailaja, R. K. Kumar, P. S. R. Murty and P E S N Krishna Prasad, "A Study on Routing lgorithms Derived From the Nature for MANETs", imacst, vol. 2 No. 1 (2011)

[21].T.P. Singh, R.K. Singh, and V. Sharma. "An Efficient Delay Elimination Protocol (EDEP) for mobile ad hoc networks", IEEE International Conference on Communication and Industrial Application (ICCIA), pp. 1-5 (2011)

[22]. Juan Jos´e and R. Srikant "Optimal scheduling for fair resource allocation in ad hoc networks with elastic and inelastic traffic." INFOCOM, Proceedings IEEE, (2010)

[23]. Shokrani, Hamideh, and S. Jabbehdari. "A novel ant-based QoS routing for mobile adhoc networks." First IEEE International Conference on Ubiquitous and Future Networks, ICUFN, (2009)

[24]. Karaboga, Dervis, and B. Akay. "A survey: algorithms simulating bee swarm intelligence." Artificial Intelligence Review, Vol.31 No.1 pp. 61-85 (2009)

[25]. J. Wang, E. Osagie, P. Thulasiraman and R. K. Thulasiram "HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network."Ad Hoc Networks, Vo. 7 No. 4, pp. 690-705 (2009)

[26]. J. Kusyk ,M. U. Uyar, E. Urrea, M. A. Fecko and S. Samtani "Applications of game theory to mobile ad hoc networks: node spreading potential game." Sarnoff Symposium IEEE, (2009)

[27]. Mbarushimana, Consolee, and A. Shahrabi. "Comparative study of reactive and proactive routing protocols performance in mobile ad hoc networks." 21st IEEE International Conference on Advanced Information Networking and Applications Workshops, 2007 AINAW'07. Vol. 2 (2007)

[28]. L. Zhao, J. Zhang, K. Yang and H. Zhang "Using incompletely cooperative game theory in mobile ad hoc networks." IEEE International Conference on Communications, 2007. ICC'07. IEEE, (2007)

[29]. M. Dorigo, M. Birattari and T. Stutzle “Biological Inspiration”, in ant colony optimization ,IEEE computational intelligence magazine, (2006)

[30]. H. F. Wedde, M. Farooq, T. Pannenbaecker, B. Vogel, C. Mueller, J. Meth and R. Jeruschkat, "BeeAdHoc: an energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior." Proceedings of the 2005 conference on Genetic and evolutionary computation ACM, (2005)

[31]. Wang, Y. Chiun, and Y. Tseng, "Packet Fair Queuing Algorithms for Wireless Networks." Design and analysis of wireless networks, Vol. 1 (2005)

[32]. Banchs, Albert, and X. Perez, "Distributed weighted fair queuing in 802.11 wireless LAN." ICC 2002. International Conference on IEEE, Vol. 5 (2002)

[33]. Marina, M. Kumar., and S. R. Das, "On-demand multipath distance vector routing in ad hoc networks." IEEE International Conference on Network Protocols, (2001)

[34]. http://en.wikipedia.org/wiki/Scheduling_(computing) accessed on 3-3-2014.

[35]. http://en.wikipedia.org/wiki/Ns_ (simulator) accessed on 11-10-2013.

Publications

Communicated

P. Sharma and R. Kumar, "Enhanced swarm intelligence-based scheduler to improve QoS for MANETs", 5th International Conference Confluence 2014 The Next Generation Information Technology Summit 25-26th September, 2014.