QoS Routing For Mobile

Ad Hoc Networks Using Genetic

Algorithm

by

Jiwa Abdullah

A Doctoral Thesis

Submitted in partial fulfillment of the requirements

for the award of

Doctor of Philosophy

Department of Electronic and Electrical Engineering of Loughborough University

· 2007

Abstract

Mobile Ad Hoc Networks (MANETs) are a class of infrastructure less network architecture which are formed by a collection of mobile nodes that communicate with each other using multihop wireless links. They eliminate the need for central management, hence each node must operate cooperatively to successfully maintain the network. Each node performs as a source, a sink and a router. Future applications of MANETs are expected to be based on all-IP architecture, carrying a multitude of real-time multimedia applications such as voice, video and data. It would be necessary for MANETs to have an efficient routing and quality of service (QoS) mechanism to support diverse applications. This thesis proposes a set of cooperative protocols that provide support for QoS routing. The first is the on-demand, Non-Disjoint Multiple Routes Discovery protocol (NDMRD). NDMRD allows the establishment of multiple paths with node non-disjoint between source and destination node. It returns to the source a collection of routes with the QoS parameters. The second part of the protocol is the Node State Monitoring protocol for the purpose of monitoring, acquisition, dissemination and accumulation of QoS route information. The third part of the protocol implements the QoS route selection based on a Genetic Algorithm. The GA is implemented online with predetermined initial population and weighted-sum fitness function which operates simultaneously on the node bandwidth, media access delay, end to end delay and the node connectivity index (*nci*). The term *node connectivity index* is a numerical value designed to predict comparatively the longest time a node-pair might be connected wirelessly.

Keywords:

MANET, QoS Routing, Non-Disjoint Multiple Routes Discovery Protocol, Node State Monitoring Protocol, QoS Route Selection, Genetic Algorithm, node bandwidth, media access delay, end to end delay and Node Connectivity Index.

Acknowledgements

Table of Contents

List of Figures

List of Tables

List of Symbols

m	number of valid routes from source to target
P_{ST}	a set of valid routes, $P_{ST} = \{ P_0, P_1, P_2, \ldots P_{m-1} \}$
P_i	a route, $P_i = [n_0, n_1, n_2, \ldots n_i, \ldots n_k]$
n_i	i-th mobile node
k	total number of node in the routes,
T	Route Accumulation Latency
i,j	subscripts of node position in a route
D	matrix of end-to-end delay where the element is Di,j indicating the ete delay from node i to node j
C	matrix of Node Connectivity Index where the element is $nci_{i,j}$ indicating the node pair connectivity index for the neighbour node of i and j
BW	matrix of node bandwidth where the element B_i indicates the i-th node Bandwidth
DM	matrix of medium access delay where the element is d_i indicating the i-th node MAC delay
t	time
L(i,j)	node connectivity between node i and j
v	node velocity
V_{max} or V_m	maximum velocity of a node
θ	the direction of arrival of node at the circular line of transmission range which is between $-\pi/2$ and $+\pi/2$
$f_v(v)$	node velocity probability density function(pdf)

$f_\theta(\theta)$	robability density function(pdf) of direction of node movement
X	distance traversed by a node from point TOD to point TOA on the circular radius equal to the transmission range of 250m
R	transmission range of the node
×	distance travel by node
$F_x(x)$	density function of x.
$Prob(X \le x)$	probability of X which is less than or equal to x.
T_{NCT}	node connectivity time for mobile node to travel from point TOD to point TOA
$P_{r,t}$	received and transmitted power
$G_{r,t}$	receiver and transmitter antenna gain
$P_{1,2}$	received power due to the first packet arrival, received power due to the second packet arrival
d	distance between transmitter and receiver
L	loss factor
λ	wavelength in meters
npem	node pair expansion metric
npcm	node pair contraction metric
nci	node pair connectivity index
G	undirected graph
E	set of all mobile nodes
$B_{AVA,i}$	amount of bandwidth available at a node i
D_{E2E}	end to end delay

D_{MAC}	MAC delay
B_{REQ}	bandwidth request by the source
$v(i,j)$	relative velocity of node i with its neighbour j
M_i	mobility characterisation
F_{RWP}	node movement function following Random Waypoint Model
F_{TRJ}	node movement function following Trajectory Model
T_{BUSY}	time when the node is active transmitting and receiving
$T_{SAMPLING}$	the length of time the protocol is sampling the node busy time
U_{INST}	instantaneous utilisation of wireless channel
U_{AVG}	average utilisation using using weighted moving average
a	smoothing factor for moving average of channel utilisation
$B_{CON,i}$	bandwidth consumed at a node
TMT	Theoretical Maximum Throughput
$C_{SUM(S,T)}$	sum of *nci* for all node pair from S to T
$F_{1,2,3}$	fitness function
a, β, γ	weighting coefficient for the fitness function
μ	fittest fraction for elitism selection method
P_c	probability of crossover
P_m	probability of mutation
C_{AMC}	average minimum cost

Glossary

ACK	Acknowledgement
AODV	Ad Hoc On Demand Distance Vector
AETED	Average End to End Delay
ANH	Average Number of Hops
APDR	Average Packet Delivery Ratio
ARLR	Average Routing Load Ratio
ATPUT	Average Throughput
ARP	Address Resolution Protocol
BB	Blackburst
CBR	Constant Bit Rate
CDMA	Code Division Multiple Access
CEDAR	Core-Extraction Distributed Ad-Hoc QoS Routing
CONN	Connectivity packet type
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CW	Contention Window
DATA	Data packet type
DCF	Distribution Coordination Function
DIFS	DCF Interframe Space
DSDV	Distance Source Distance Vector
DSSS	Direct-Sequence Spread Spectrum

DSR	Dynamic Source Routing
FDMA	Frequency Division Multiple Access
FSM	Finite State Machines
GA	Genetic Algorithm
HR-DSSS	High Rate – Direct-Sequence Spread Spectrum
INSIGNIA	In-Band Signalling For QoS in Ad-Hoc Mobile Networks
IP	Internet Protocol
LAN	Local Area Network
MACA/PR	Multiple Access Collision Avoidance with Piggyback Reservation
MAC	Medium Access Control
MANET	Mobile Ad Hoc Network
MDS	Minimum Dominating Set
MSDU	MAC Service Data Unit
NAV	Network Allocation Vector
nci	node connectivity index
NDMRD	Non-Disjoint Multiple Routes Discovery protocol
QOS	Quality Of Services
QOSRGA	QoS Routing Using Genetic Algorithm
RAL	Route Accumulation Latency
RREQ	Route Requests packet type
RREP	Route Reply packet type
RERR	Route Error packet type

RTS/CTS	Request To Send/Clear To send
RWP	Random Waypoint Mobility Model
RWS	Roulette Wheel Selection
SUS	Stochastic Universal Selection
SWAN	Stateless Wireless Ad-Hoc Networks
TDMA	Time Division Multiple Access
TMT	Theoretical Maximum Throughput
TOA	Mobile node, Time Of Arrival, a point on a circular transmission distance from another node, moving towards it.
TOD	Mobile node, Time Of Departure, a point on a circular transmission distance from another node, moving away from it.
TOUR	Tournament Selection
TRJ	Mobility Trajectory Model
VLSI	Very Large Scale Integration
WLAN	Wireless Local Area Network
WRR	Weighted Round Robin
WFQ	Weighted Fair Queueing
WRR	Weighted Round Robin

CHAPTER 1
INTRODUCTION

1.1 Overview Of Mobile Ad Hoc Networks

Wireless networks are becoming more widespread, succeeding as they do to make access "every time, everywhere" possible, through today's IP-based communication system. The main architectures for wireless networks are the wireless local area network (WLAN) and wireless ad hoc networks. The mobile nodes in WLAN directly communicate with the fixed base-station to send their traffic to nodes in the same or different WLAN. It is a single hop communication and routing is not necessary. In a mobile ad hoc network (MANET), routing is indispensable for packet forwarding. Each node acts as router, as well as source and destination node. In principle, a MANET allows an arbitrary collection of mobile nodes to create a network on demand. Numerous scenarios lacking in network infrastructure could benefit from the creation of MANETs. Such scenarios are rescue and emergency operations, natural or environmental disasters, law enforcement activities, tactical and military missions, commercial projects such as conferences, exhibitions, workshops and meetings, and in educational classrooms. It is expected to be a major focus of research for years to come.

1.2   Research Motivation

There are many challenges in the creation of a MANET, such as routing challenges, wireless medium challenges, scalability challenges and portability challenges. It is envisaged that very large groups of mobile nodes located over a large geographical area belonging to the same network grouping can benefit from this system. Chlamtac *et al.* [1] provide a comprehensive overview of this dynamic field of research, covering the technologies, characteristics, capabilities, applications and design constraints.

The routing mechanisms in a MANET are challenged by a complicated interaction of three fundamental difficulties, namely contention, congestion and node connectivity. The nature of wireless communication results in significant contention for the shared wireless channel. Bandwidth depletion results in higher congestion and reduced support for data traffic when compared to the wired connection, which is attainable with wired connections. Unique to MANET is a set of challenges created by node mobility. Node mobility in MANET makes and breaks node connectivity. These makes and breaks may occur randomly. These kinds of topology changes are the key challenge that MANET routing protocols must overcome. This means that MANETs are constrained by limited node connectivity, limited bandwidth and limited processing capabilities, due to battery capacity and mobile nature of the node. The transmission process must acquire and use all the mobile nodes' resources so as to provide an ample amount of bandwidth and delay within the limitation of node connectivity lifetime. Hence, the research goals are aimed at improving the effectiveness of multimedia transmission by the use of a genetic algorithm approach in determining the best routes within the QoS routing mechanism. Before proceeding with the implementation of GA based route selection algorithm, one has to be able to accumulate all the important network parameters instantaneously. All the available resources along the potential routes, need to be quantified before selecting them.

On-demand routing protocols in particular are widely studied because they consume more bandwidth than proactive protocols. Ad Hoc On-demand Distance Vector (AODV) and Dynamic Source Routing (DSR) are the two most

widely studied on-demand ad hoc routing protocols. Previous work has shown limitations of the two protocols. The main reason is that both of them build and rely on a unipath route for each data session. Whenever there is a link break on the active route, both of the two routing protocols have to invoke a route discovery process. A new route discovery is invoked when all of its routing paths fail or when only a single path remains available.

Future applications of MANETs are expected to be based on all-IP architecture and be capable of carrying real-time multimedia applications such as voice and video, as well as data. Multimedia applications place stringent requirements on networks for delivering real-time audio and video packets. Compared to the requirements of traditional data-only applications, these new requirements generally include a high packet delivery rate, a low delay and a small jitter. ITU-T G.114 has recommended 150 ms as the maximum desired one-way delay to achieve high-quality voice. A delay above 250 ms is felt to be unacceptable.

1.3   Research Contributions

In this thesis, a set of cooperative protocols that provide support for QoS routing for mobile ad hoc networks have been addressed. The major contributions of the work are as follows.

- A Non-Disjoint Multiple Routes Discovery (NDMRD) protocol was proposed to overcome the shortcomings of current unipath routing and disjoint multipath routing protocols in terms of connectivity lifetime. A novel aspect of NDMRD is that it achieved multiple non-disjoint routing paths. More routes with diverse qualities were discovered, hence more choices for selection. The QoS parameters extracted from each node of those routes were subsequently accumulated at the source.

- Node State Monitoring protocol is proposed. Node State Monitoring protocol monitors the arrival of all types of packets, measures and caches the relative velocity and then ensures that Node State of every node is maintained and managed. Each node calculates and accumulates various Node States information for the purpose of calculating QoS route with bandwidth and delay demand.

3

- A technique for measuring the instantaneous node bandwidth by using the NAV duration timer, which is available within the CSMA/CA IEEE 802.11 medium access control protocol.

- A connectivity metric, which estimates the length of time a node pair is connected to each other wirelessly, was proposed. It is termed as the Node Connectivity Index, *nci*. The *nci* has been defined and used it to promote the route's quality.

- A method for selecting QoS route using Genetic Algorithm was proposed. The QoS parameters were used to select the best possible routes according to the fitness functions which are based on node bandwidth, end-to-end delay, medium access delay and *nci*. Weighted sum approached was applied so that a single fitness was obtained for all four QoS parameters.

1.4 Thesis Organisation

This thesis is organised as follows. Chapter 2 reviews some of the proposed models related to QoS support in MANETs. It described those models that are related to cross layer models, mac layer models and QoS routing models. The two most popular best effort routing protocols, AODV and DSR are described. The strategies taken in this thesis were to identify the QoS parameters and operate on them simultaneously, as opposed to other methodologies such as QoS metrics ordering, sequential filtering, scheduling, admission control and control theory approaches that were previously taken. The usage of GA as an alternative to the above strategies is introduced.

Chapter 3 presents the general structure for QoS routing. The overall QoS routing framework were outlined using a set of flowcharts. The design of Non-Disjoint Multiple Routes Discovery protocols is described, detailing its features. The implementation of the routing framework using the Opnet Modeler simulation software has been elaborated upon.

Chapter 4 introduces Node Connectivity index, *nci*. It is based on node pair relative movements due to contraction and expansion mobility model. The *nci*

4

indicates the relative time of node pair connectivity by measuring the relative velocity. The relative velocity is measured by the two packet arrival method. The algorithm for the calculation of *nci* is described. The simulation examples have shown that *nci* is a viable proposal to identify the longest connectivity duration for node pair.

Chapter 5 discusses a means by which to represent the temporal and spatial characteristics of a node. The usage of Node State for proper node characterisation is proposed and it is defined in the context of QoS routing. Items which constitute Node States are listed. Bandwidth, as one of the node state, is described in detail. Furthermore, the bandwidth measurement in the shared medium using the NAV duration of the MAC layer is implemented.

Chapter 6 presents the QoS route selection algorithm. The use of Genetic Algorithm as the search algorithm is proposed. The genetic algorithm is formulated by introducing a multiple metrics fitness function using a weighted sum approach. It follows a typical GA process with slight variations, such as using variable size chromosomes, multiple crossing sites, subroute determination using two stage search during the mutation process and initial small population size.

Chapter 7 describes in detail the comparative evaluation of QOSRGA protocol with the existing protocols, BE-AODV and BE-DSR. The simulation has shown that, for most measurements, there exists an improvement in performance of QOSRGA when compared to best effort routing protocol, BE-DSR and BE-AODV.

Chapter 8 summarised the contributions and briefly examined the potential area for future work.

1.5 Summary

This chapter is an introduction to the realm of mobile ad hoc networks. A general description of a MANET is given. The chapter provides the motivation for this thesis, the contribution and the thesis organisation.

CHAPTER 2
QOS MODEL FOR MOBILE AD HOC NETWORKS

2.1   Introduction

A mobile ad hoc network is an autonomous system of mobile nodes with its own network management mechanism. Due to the increasing popularity of multimedia applications such as collaborative computing and communications, QoS support has become an unavoidable task. To support QoS, the link state information such as delay, bandwidth, cost, lost rate and error rate in the network should be available and manageable. However, getting and managing the link state information of MANETs is a difficult task due to the quality of wireless link. Therefore, the challenge is to implement QoS functionality with limited resources in a dynamic environment. QoS support in MANETs usually includes issues at all layers of the network infrastructure, broadly classified as cross layer QoS models, QoS aware routing and MAC layer QoS provisioning.

Most mechanisms geared at achieving QoS in MANETs could be traced to solutions based on mapping wireless networks to a wireline paradigm of nodes and links. This paradigm is less appropriate, because the links are not physical entities and do not accurately represent the radio frequency medium. Furthermore, link abstraction normally results in overprovisioning when needing to deliver QoS. Wired networks consist of nodes which changes less frequently,

6

such as wireless mobile nodes. Parameters in a wireless node tend to evolve according to the scenario where it resides.

In this chapter, it is argued that the critical resource is the electromagnetic spectrum in a space, where contention occurred due to the node and the neighbours. A node state is proposed where most of the important node parameters or status are saved and can be retrieved. The related work on QoS Routing for MANETs is reviewed and the approach to QoS route selection using Genetic Algorithm is outlined. Some of the routing protocols have draft specifications available on the IETF web site [2]. A node is a basic hardware and software unit in the network that has the capability to forward packets based on its local routing table. A host is another basic unit in the network that may attach to or act as a router. It can be either the source or the sink of a data flow in the network. Two nodes within the signal transmission range were neighbours to each other. A route is a sequence of nodes connecting two end hosts.

This chapter provides a survey of an existing QoS model for MANETs and the issues involved in supporting QoS across all layers in MANETs. Different approaches are classified, various techniques are discussed, and the future issues related to QoS routing in MANETs are outlined. The rest of the chapter is organised as follows. Section 2.2 describes the existing QoS model for MANETs. Section 2.3 outlines the methodology for QoS routing. Section 2.4 describes the design approach of using a Node State as the foundation of QoS routing, where Node State information is collected and cached. The chapter is summarised in Section 2.5.

2.2 Existing Qos Model For MANETs

2.2.1 Cross Layer QoS Model

2.2.1.1 *In-Band Signalling For QoS in Ad-Hoc Mobile Networks (INSIGNIA)*
INSIGNIA [3] is a signalling system which works with various routing protocols. The INSIGNIA protocol provides per-flow QoS by piggybacking soft-state reservations onto data packets in an IP option field. The reservations must

be renewed within a given interval. Each INSIGNIA IP option is self-contained, including information to setup and maintain a bandwidth reservation.

If an intermediate node on the source-destination route is not able to provide the demanded bandwidth, it will still forward the packet, but modifies the QoS header to indicate lack of resources. The destination may then provide feedback on the QoS provision of the route to the source node which, in turn, may decide to abandon the session or to adapt the flow to the available resources. This approach avoids explicit signalling and hard state reservations, so that it can deal with the dynamics of MANETs in a flexible way. The INSIGNIA QOS framework allows packet audio, video and real-time data applications to specify their maximum and minimum bandwidth needed. It plays a central role in resource allocation, restoration control, and session adaptation between communicating mobile hosts. Based on end-to-end bandwidth availability, QOS mechanisms attempt to provide assurances for these adaptive services.

The INSIGNIA QOS framework consists of the architectural components as illustrated in Figure 2.1. These components are the in-band signaling, admission control, packet forwarding, routing, packet scheduling and medium access control. The in-band signaling establishes, restores, adapts and tears down adaptive services between source-destination pairs. The admission control is responsible for bandwidth allocation to flows based on the maximum-minimum bandwidth requested. Packet forwarding classifies incoming packets and forwards them to the appropriate module, routing module tracks variations in MANET topology, making the routing table visible to the node's interface. The QoS framework expects the availability of routing protocols that can be inserted into the architecture, generating new routes. Packet scheduling responds to location-dependent channel conditions when scheduling packets utilising a WRR service disciplines. QoS-driven then accesses the MAC, the shared wireless media for adaptive and best effort services, transparent to any underlying media access control protocols.

INSIGNIA provides an effective signalling protocol which efficiently delivers adaptive real-time flows in MANETs. On the negative side, it had a scalability

problem. Multimedia applications support fine-grained throughput levels with adaptive codecs that can be configured dynamically at run-time. It is difficult to implement a two-level QoS support using INSIGNIA as it does not account for the excessive reservations. High signalling and processing overhead should be reduced. It is rather complex - signaling, systems, consume energy, reduce memory and degrade overall processing capacity.

2.2.1.2 *Stateless Wireless Ad Hoc Networks (SWAN)*

SWAN [4] classifies data packets into high-priority real-time traffic and normal, low-priority traffic, as illustrated in Figure 2.2. A *classifier* is determined if the datagram carries a high-priority real-time data traffic. Otherwise, the datagram is considered low-priority and must wait until all real-time datagrams are dispatched. The *shaper* limits the relaying of low-priority traffic, thus reducing contention. The delay of real-time traffic is kept low, to avoid starvation of bulk traffic. SWAN works with any existing routing protocol. When a source node wishes to establish a real-time session to another node, it probes the path to the destination to identify the bandwidth available for real-time traffic before it launches the new session. Topology changes may result in quality degradations to the ongoing sessions, because of rerouting of previously admitted flows. Basically, SWAN functions to maintain a stateless model without need to process complex signalling mechanisms. The problem of SWAN is how to choose a sensible amount of bandwidth for real-time traffic. Choosing a value which is too high results in a poor performance of real-time traffic. Choosing a value which is it too low results in denial of real-time flows. SWAN provides a model that deals with traffic on a per class basis, but uses only two levels of services: best effort and real time traffic.
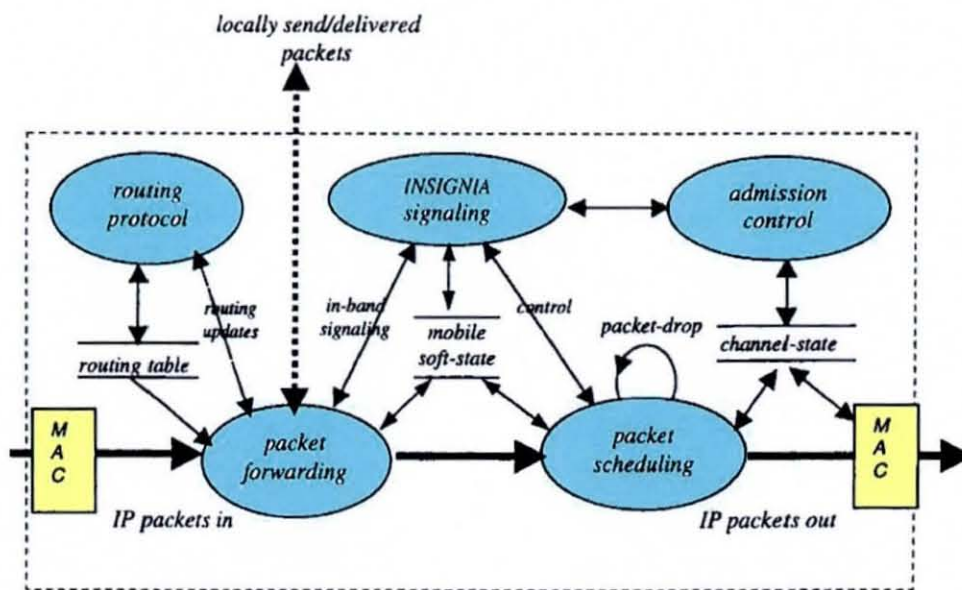
Figure 2-1 INSIGNIA Framework [3]
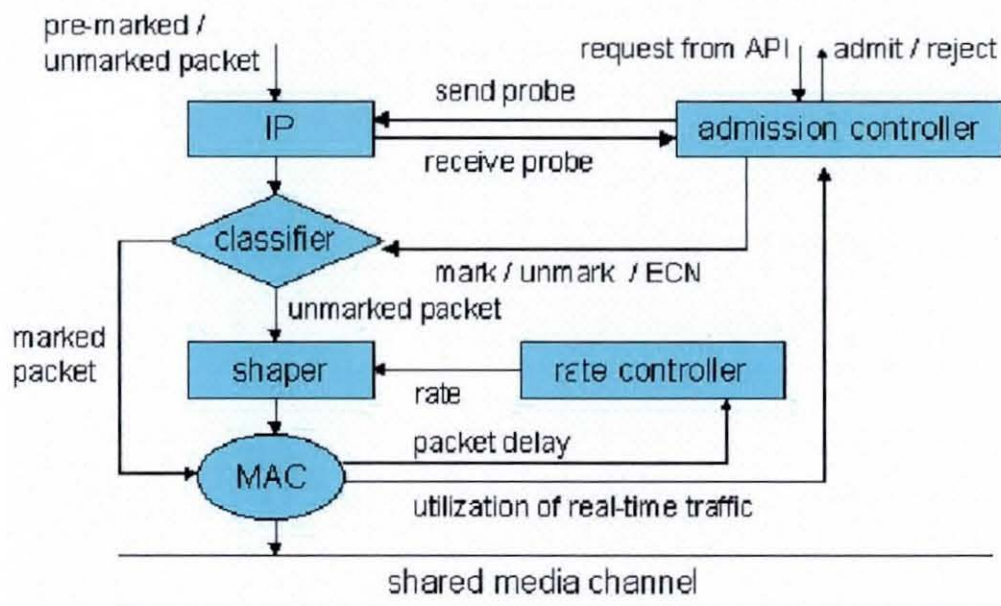
Figure 2-2 SWAN Framework [4]

10

2.2.2   MAC Layer QoS Provisioning

2.2.2.1   IEEE 802.11 DCF and its extension

Reference [5] provides detailed descriptions of all the QoS schemes at the MAC layer, including an IEEE 802.11 standard. In DCF mode, after the node has sensed the medium to be idle for a time period longer than DIFS, it begins transmitting. Otherwise, the node defers transmission and starts to back off. Each node maintains a variable contention window (CW), the low and high ends of which are represented as CW_{min} and CW_{max}, respectively. The backoff duration is decided by a backoff timer set to a random value between 0 and CW. If the medium becomes idle for a period longer than DIFS, the backoff timer is decremented. As soon as the timer expires, the node starts transmission. IEEE 802.11 DCF is a good example of a best effort control algorithm. It has no notion of service differentiation and no support for real-time traffic.

2.2.2.2   Black Burst Contention Mechanism

The black burst (BB) contention scheme avoids packet collision and mitigates packet starvation problem. It ensures that real-time traffic gets high priority. Two or more traffic flows of the same service class are scheduled in a distributed manner with a fairness guarantee. Nodes contend for the medium if it is idling for a longer period than the interframe space. Nodes with best effort traffic and those with real-time traffic use different interframe space values. A BB contention scheme is added to the CSMA/CA protocol in the following process. Real-time nodes first contend for transmission rights by jamming the media with pulses of BBs. Each contending node uses a BB of different length which is an integral number of black slots. The number of slots that form a BB are an increasing function of the contention delay experienced by the node, measured from the instant an attempt to access the channel is scheduled until the node starts transmission of its BB. Following each BB transmission, a node senses the channel for an observation interval and determines without any ambiguity whether its BB is of longest duration. Hence, only one winner is

produced after this contention and proceeds to transmit its real time packets. [6]

2.2.2.3 Multiple Access Collision Avoidance with Piggyback Reservation (MACA/PR)

MACA/PR provides guaranteed bandwidth support using reservation for real-time traffic, over a single hop. It could also work with an existing QoS routing algorithm. The first data packet in the real-time stream makes reservations along the path. A RTS/CTS dialog is used to make sure that it is transmitted successfully. Both RTS and CTS specify how long the data packet will be. Any node near the sender that hears the RTS will defer long enough so the sender can receive the returning CTS. Any node near the receiver that hears the CTS will avoid colliding with the following data packet. The RTS/CTS dialog is used only for the first packet. The subsequent packets do not require this dialog. When a sender sends a data packet, the sender schedules the next transmission time after the current data transmission and piggybacks the reservation in the current data packet. Upon receiving the data packet correctly, the intended receiver enters the reservation into its reservation table and returns an ACK. The reservation ACK serves as a protector for the given time window, and a mechanism to inform the sender if something is wrong on the link. [7]

2.2.3 QoS Aware Routing Model

2.2.3.1 Best Effort Routing Protocols in MANETs – Dynamic Source Routing (DSR)

DSR [8][9] is an on-demand routing protocol for MANETs. It is based on source routing, in which a source node indicates the sequence of intermediate routes in the header of a data packet. The operation of DSR can be divided into route discovery and route maintenance. Each node keeps a cache of the source routes that it has learned when a packet arrived. When a packet is to be sent to some destination, it checks its route cache to determine whether it already has the latest routes to the destination. If no route is found, the node initiates route discovery procedure by broadcasting a RREQ message to neighbouring nodes.

This RREQ message contains the address of the source and destination node, a unique identification number, and a route record to keep track hop sequence as it is propagated throughout network.

On receiving a route request, the intermediate node checks whether its own address is already in the route record of the route request message. Otherwise it appends its address to the route record and forwards to its neighbours. Figure 2-3 illustrates the formation of the route record as the RREQ propagates through the network. When the destination node receives the RREQ, it appends its address to the route record and returns a RREP packet to the source node. If the destination already has a route to the source, it can use that route to send the reply; otherwise, it uses the route in the RREQ message to send the reply. The first case is for situations where a network might be using unidirectional links, in which case it might not be possible to send the reply using the reversed route of the one taken by the route request. If symmetric links are not supported, the destination node may initiate its own route discovery message to the source node and piggyback the RREP on the new route request message. Figure 2-4 shows the transmission of route record back to the source node. Route maintenance uses the route error, RERR messages and acknowledgement messages, ACK. If a node detects a link failure when forwarding data packets, it creates a route error message and sends it to the source of the data packets. The route error message contains the address of the node that generates the error and the next hop that is unreachable. When the source node receives the route error message, it removes all routes from its route cache that have the address of the node in error. It may initiate a route discovery for a new route if needed. In addition to route error message, ACKs are used to verify the correct operation of links.

2.2.3.2 Best Effort Routing Protocols for MANETs – Ad Hoc On-Demand Distance Vector (AODV)

The AODV protocol [10] adopts both a modified on-demand broadcast route discovery approach used in DSR [8] and the concept of destination sequence number adopted from destination-sequenced distance-vector routing (DSDV) [11] . When a source node wants to send a packet to some destination and

does not have a valid route to that destination, it initiates a path discovery process and broadcasts a route request (RREQ) message to its neighbours. The neighbours, in turn, forward the request to their neighbours until the RREQ message reaches the destination or an intermediate node that has an up-to-date route to the destination. Figure 2-5 illustrates the propagation of the broadcast RREQs in an ad hoc network.

In AODV, each node maintains its own sequence number and a broadcast ID. Each RREQ message contains the sequence numbers of the source and destination nodes and is uniquely identified by the source node's address and a broadcast ID. AODV utilises destination sequence numbers to ensure loop-free routing and use of up-to-date route information. Intermediate nodes can reply to the RREQ message only if they have a route to the destination whose destination sequence number is greater or equal to that contained in the RREQ message. So that a reverse path can be set up, each intermediate node records the address of the neighbour from which it received the first copy of the RREQ message, and additional copies of the same RREQ message are discarded. Once the RREQ message reaches the destination (or an intermediate node with a fresh route) the destination (or the intermediate node) responds by sending a route reply (RREP) packet back to the neighbour from which it first received the RREQ message. As the RREP message is routed back along the reverse path, nodes along this path set up forward path entries in their routing tables, as shown in Figure 2.6. When a node detects a link failure or a change in neighbourhood, a route maintenance procedure is invoked. If a source node moves, it can restart the route discovery procedure to find a new route to the destination. If a node along the route moves so that it is no longer contactable, its upstream neighbour sends a link failure notification message to each of its active upstream neighbours. These nodes in turn forward the link failure notification to their upstream neighbours until the link failure notification reaches the source node.
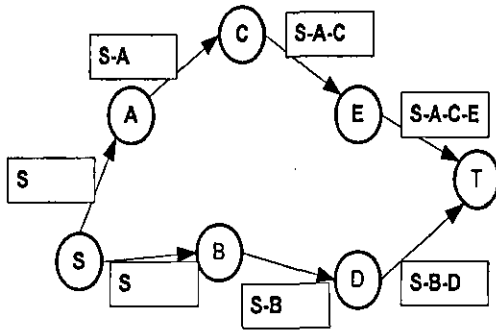
Figure 2-3 Propagating Route Request for BE-DSR
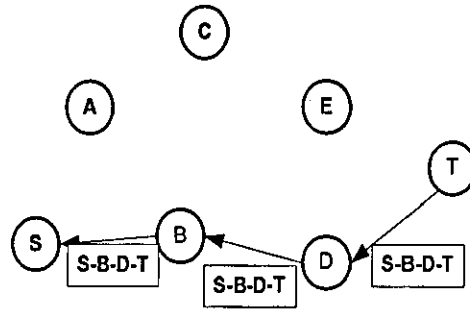
Figure 2-4 Propagating Route Reply BE-DSR
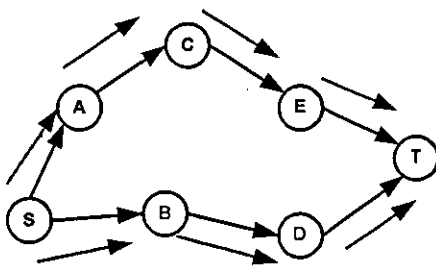
Figure 2-5 RREQ Propagation towards Destination
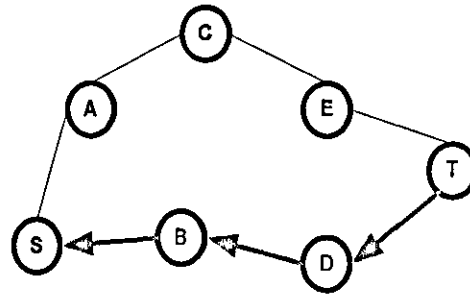
Figure 2-6 RREP Propagation to the Source

Core-Extraction Distributed Ad-Hoc QoS Routing (CEDAR)

Essentially, CEDAR consists of three components: core establishment, QoS-state propagations and routes computation. Using CEDAR, routes that satisfy the bandwidth requirement are computed [12]. In the core establishment, the network core shown in Figure 2-7 consists of a set of core nodes and a set of links which are a Minimum Dominating Set (MDS) of the network. The link connects every two core nodes that are within three hops of each other. A distributed approximation algorithm to choose core nodes is presented in the literature. It also proposes a core broadcast mechanism that propagates the core nodes information into other nodes in the network. In the QoS-state propagation, each core node keeps the state information of remote and stable high bandwidth links. Each node in the network monitors the bandwidth over the links to its neighbours. QoS routing is done by propagating the bandwidth availability information of stable high bandwidth links to core nodes. The

information about low bandwidth links is kept local. The QoS route computation consist of three main steps: destination discovery and core path establishment, search for a stable QoS route using the directive from the established path and dynamically recompute the QoS route upon link failures or topology changes. Route computation first establishes a core path from the dominator of the source to the dominator of the destination.

The literature [12] describes the test on the algorithm using bandwidth as the QoS parameter, giving the performance evaluation on message complexity for route computation, packet delivery ratio and bandwidth optimal ratio. However, it does not experiment with node movement. Another concern is that it does not run the simulations on real shared channel environment which means channel interferences and packet collision are not considered.

2.2.3.3   Ticket Based Probing (TBP)

The ticket based probing presented in [13] tries to provide QoS-constrained paths with a two stage approach. It uses proactive routing to provide nodes with imprecise state information of the network. The proactive distance vector protocol provides nodes with information on the shortest path, the maximum available bandwidth and the delay to all other nodes. To maintain low overheads it performs less frequent information exchange between the nodes.

The path discovery process utilises the state information which then initiates the probes. Probes are split at each node and propagated across different links. Each probe carries a number of tickets depending on how often the split occurs. Probes that are sent along more promising paths are assigned more tickets. In this way, route requests are done, without having to flood the entire network. Figure 2-8 illustrates an example of a route discovery with ticket-based probing. Assuming a source node **S** tries to establish a QoS-constrained connection to a destination **D**, based on its initial knowledge of the network it issues two probes P1 and P2, one and two tickets. P1 may not be further split up and is propagated directly on the most promising path towards **D**. P2 seemed more promising to **S** and thus has been equipped with 2 tickets. Consequently, it is split into two probes, P2 and P3, by an intermediate node, each carrying just 1

ticket. **D** evaluates all the three probes that arrives and issues a response to **S** along the reverse route.
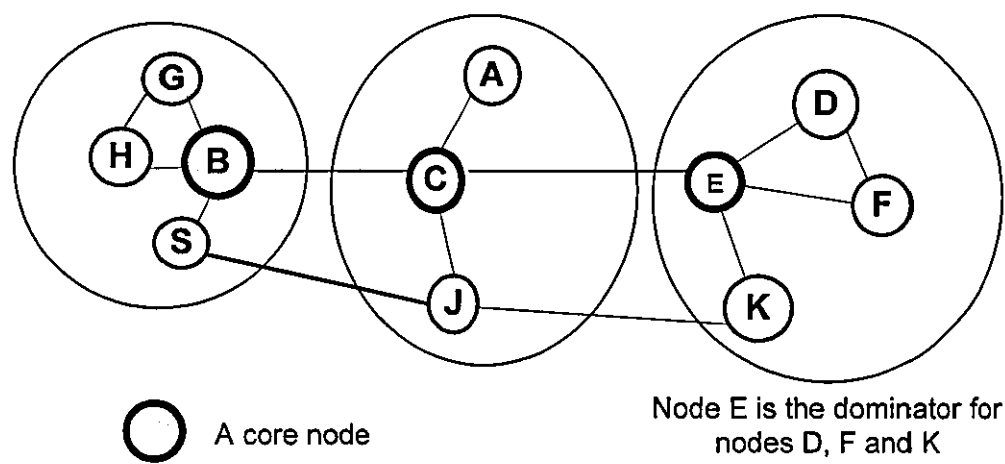
Figure 2-7 Operation of CEDAR [12]
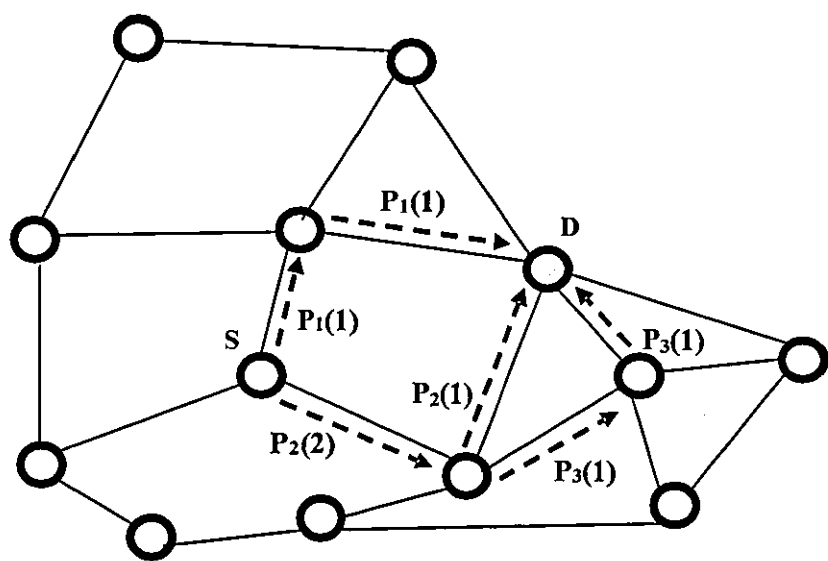
Figure 2-8 Ticket Based Path Discovery[13]

The protocol produces large overheads. The authors of the paper on TBP [13] , suggest running the proactive part of the protocol at a very low frequency, although the proactive routing protocol has to be taken into account. The more imprecise the knowledge of the network state, the more tickets are needed. Moreover, probes are sent as unicast frames, producing higher overheads than broadcast frames. The literature shows the TBP performance in a delay-constraint environment, calculating what percentage of the routes the algorithm finds that meet the delay demand. It fails to analyse other aspects of routing, such as control overhead and packet delivery ratio.

2.3 Methodology for QoS Routing

2.3.1 Challenges in Designing Qos Routing For MANET

2.3.1.1 Dynamic Topology with Infrastructureless Communication System

The topology of MANET will change dynamically due to mobile hosts changing their point of connectivity, the rate of node survivability and nodes leaving or joining the network. Saving current knowledge of the network topology and the frequent changes is an important requirement in a MANET management system. However, the frequent exchange of topology information may lead to considerable signalling overhead, congesting low bandwidth wireless links, and possibly depleting the limited battery life of the nodes. Hence the choice of mechanism used to collect topology information is critical. These complications, imposed by mobility characteristics, may severely degrade the network quality. The frequent route breakage is a natural consequence of mobility, which complicates routing. The problem multiplies when routes need to satisfy certain QoS requirements during the session lifetime. Hence, the design of QoS routing protocols in MANETs is challenged by frequent topological changes. In MANETs, pre-existing infrastructure is not available as that of the design of standard mobile networks. The nodes in an infrastructured network perform a specific network task. In contrast, nodes in MANETs are expected to route packets for other nodes, while they themselves may be a source or destination of application flows. Further, in MANET all nodes may move around. Therefore,

18

QoS routing protocol must consider the self-creating and self-organising features of MANETs.

2.3.1.2 System with limited networking resources

The wireless frequency spectrum is resource limited, which must be utilised efficiently. Additionally, the wireless medium is a shared medium, where signal attenuation, interference and multipath propagation effects, such as fading and the unguided nature of the transmitted wave, all contribute to the wasting of bandwidth resources. Wireless links are typically more bandwidth constrained than wireline networks. Contentions among the transmitting nodes contribute significantly to this effect. An overhead is often required to support reliable data transmission.

Hardware requirements to keep nodes compact, light, portable and wearable impose limitations on battery life, storage and processing capability. Solutions that reduce power consumption will often be favoured. Mobile nodes need to utilise their battery, in a manner that prolongs its lifetime. If used excessively, mobile nodes will disconnect quickly, and this affects the network functionality. QoS routing consists of three basic tasks: routes discovery, route selection and data forwarding. The first task is to discover the route, collect the node, state information and keep it up-to-date and distribute the information. The second task is the feasible route calculation using certain heuristics, and is based on the collected information. The third task is assigning data to QoS class and forwarded according to QoS routing calculations. The performance of any QoS routing algorithm directly depends on how well the first task is accomplished. In the local Node State, each node is assumed to maintain its up-to-date local information, such as queuing and propagation delay, the residual bandwidth and the availability of other resources. Every node is able to maintain the global state periodically by the use of an information exchanges mechanism.

The choice of the MAC scheme in MANETs is difficult, due to the time-varying network topology and the lack of centralised control. TDMA is complex, since there is no centralised control. FDMA is inefficient in dense networks, and CDMA is difficult to implement due to node mobility and subsequently needs to

19

monitor the frequency-hopping patterns and the spreading codes for nodes in a time-varying neighbourhood. At the MAC layer, there exists a link layer reliability problem which is related to the high bit error rate, in addition to the possible packet collision problems. QoS at the MAC layer can serve as an infrastructure for facilitating the QoS routing at the network layer. CSMA/CA is the MAC layer of choice in this thesis.

2.3.1.3 Heterogenity

MANETs are typically heterogeneous networks with diverse communication technologies employed. Their diversity comes in the form of different types of nodes, ranging from sensors, palmtops and laptops within an organisation or multiorganisation consortium. In a military application, different military units ranging from soldiers to tanks can come together, hence forming a MANET system. Nodes differ in their energy capacities and computational abilities. Hence, mobile nodes will have different packet generation rates, routing responsibilities, network activities and energy draining rates. Coping with node heterogeneity is a key factor for the successful operation of MANETs.

The quality of link is particularly significant in MANETs. The essential effect on MANETs is that the link quality can become randomly variable. Although some parts of this effect can be predicted, since variations in link quality impact packet delivery, the main QoS routing parameters, such as bandwidth, latency, reliability and jitter, are all affected. This effect can happen either during or between connections. The effect of link quality constitutes a more fundamental understanding of MANETs. MANETs are composed of a system of wirelessly interconnected nodes defined by the radio propagation model, which is a physical phenomena in nature. The whole issue of QoS routing need to be redefined such that it is the quality of networks that determine the performance. Quality of network, then, largely depends on the quality of the nodes. Therefore, node state model for QoS routing is essential. In Chapter 5, a node state model is proposed to support QoS routing. The QoS routing algorithms specifically aim at selecting reliable paths for traffic with QoS requirement. However, there is also the need to maintain the performance of

20

best-effort traffic. Thus, QoS routing mechanisms must take into consideration best-effort traffic.

2.3.2 The QoS Route Selection Strategies

The route selection algorithm has a degree of computational complexity depending on the rule of metric composition. Since applications generate traffic with very diverse requirements in terms of QoS, the route selection algorithm must select routes that satisfy a set of restrictions. The value of a metric along a route, based on its value in each hop, depends on the nature of the metric. There are additive, multiplicative and concave metrics. The rule for additive metrics composition is that the value of this metric over a path is the sum of the values of each hop. Delay and number of hops are examples of additive metrics. With a multiplicative metric, the value of a metric over a path is the product of its values in each hop, as it is the case of packet losses. The value of a concave metric over a path corresponds to the minimum value observed in all hops of that path. Bandwidth is a common example of a concave metric. When using two additive or multiplicative metric, or one additive and one multiplicative metrics, MANET QoS routing is a NP-complete problem [14] . The challenge is that the QoS route selection algorithm must be addressed in order to conceive QoS routing strategies that are efficient and scalable. The major heuristics for the solution to this problem are discussed in the following sub-section.

2.3.2.1 The Ordering of QoS Metrics

Metric ordering requires the identification of the metric that has higher priority and the computation of the best paths according to this metric. Next, the second metric is used in case of a tie, to decide which the best paths are. This is the case of shortest-widest path and widest-shortest path algorithms. Shortest-widest path algorithms first find paths with maximum available bandwidth. Next, if there are paths with the same amount of available bandwidth, the path which has the shortest number of hops is selected. The main objective of this type of algorithm is to do load balancing, showing the best performance when the load in the network is light. However, this approach

damages best-effort traffic performance because it contributes to resource consumption. The reason for this is that paths with many hops, normally exhibit greater available bandwidth [15] . Another type of shortest-widest path algorithm uses, as the second metric, the hop count and propagation delay. The corresponding path computation algorithms, based on distance-vector and link-state, are presented in [14] . Widest-shortest path algorithms select from the shortest paths with equal number of hops, the path that has higher bandwidth availability. The Dijkstra algorithm can also be used to compute widest-shortest paths [15] . For each number of iterations of the path computation algorithm, it selects the node with the least number of hops. If more than one node has the same number of hops, then the node with maximum available bandwidth is selected.

2.3.2.2 The Sequential Filtering

In the sequential filtering heuristics [14] , links that cannot handle the required bandwidth are excluded from the network. The route selection algorithm, such as shortest path, is computed. The threshold that determines the exclusion of a link depends on the instant of application of the route selection algorithm. When paths are computed on-demand, the desired value of bandwidth and delay can be expressed on the request. For path pre-computation, it is necessary to compute and store several pre-computed routes that satisfy the defined range of bandwidth values. Sequential filtering can also be used to find paths subject to more than two constraints.

2.3.2.3 Scheduling Discipline of Metrics

The problem of the complexity of path selection algorithms can be surpassed using the relationships among QoS parameters determined by the nature of scheduling disciplines. Particularly, if it used a WFQ scheduling mechanism, it is possible to use the relations between bandwidth, delay and jitter, to find a path, in polynomial time, subject to constraints of bandwidth, delay and jitter [15] . In INSIGNIA[3] model, a scheduling module utilises a WRR discipline in response to location dependent channel conditioning.

2.3.2.4 The Admission Control Techniques

In some QoS architectures, the admission of new flows is subject to a mechanism of admission control. It interacts with routing mechanism whereby information about the network topology can be of use to admission control decision. Admission control and QoS routing are tightly connected with resource reservation. INSIGNIA[3] and SWAN[4] QoS model utilise the admission control mechanism.

2.3.2.5 The Control Theory Approach

The control theory approach offers a successful track record in physical process control. It gives performance guarantees in the face of uncertainty, non-linearities and time variations in the system. It does not require accurate system models and utilises a feedback mechanism. The performance of software services is governed by queuing dynamics which may be expressed by differential equations, akin to those of physical systems. Bao Li *et al.* [16] presented a model for a QoS mechanism employing feedback control theory. The ideal objectives of the model consist of two aspects. Firstly, it could accommodate variable QoS requirements, in a timely fashion. Secondly, it could accommodate concurrency of resource access among multiple applications sharing the same pool of available resources.

2.3.3 Using the Genetic Algorithm for QoS Route Selection

As seen in the previous section, QoS routing is a key MANET function for the transmission and distribution of multimedia services. It has two objectives: (1) finding routes that satisfy QoS constraints and, (2) making efficient use of limited resources. The complexity involved in the networks may require the considerations of multiple objectives at the same time, for the routing decision process. In this thesis Genetic Algorithm based QoS routing for MANET as a multiple objective optimisation is proposed. Moreover, the emergence of the hardware implementation of a GA could transform it into a high speed hardware implementation working in real time.

23

2.3.4 Application of GA to communications network

The GA has been successfully applied to numerous combinatorial search problems. Coley [17] outlines wide-ranging practical areas in the field of engineering to which a GA had been effectively applied. These are: image processing, VLSI routing, water networks, control and communication networks. M. Gen *et al.* [18] produced a detailed study of various GA-based industrial engineering applications. A GA was applied to problems related to scheduling, spanning tree, transportation, reliability, optimisation, network design and network routing. In communication networks, a GA was utilised to optimise networks operations. R. Elbaum *et al.* [19] use a GA in designing a LAN with objectives to minimise the network delay when addressing the issue of clustering and routing. S. Mao *et al.* uses a GA to formulate effectively the routing problem as applied to multiple description video in wireless ad hoc networks [20] . Hwang *et al.* [21] proposed a multicast routing algorithm based on GA in a real computer network. It produced an extension to the algorithm addressing the issue of QoS constraints. Researchers have applied a GA to the shortest path routing problem [22], the multicasting routing problem [23] , the dynamic channel allocation problem [24] and also the dynamic routing problem [25]. Munetomo [26] proposed the GA algorithm which can be used for a wired or wireless network. It uses variable-length chromosomes [27] to encode a feasible solution. In contrast, Inagaki [28] proposed a GA that employs fixed length chromosomes. The chromosomes consist of sequences of integers, and each gene represents a node identity.

The successful application of GA is due to the fact that a GA is computationally simple and provides a powerful parallel search ability. It is immuned to a problem's lack of locality, because their search takes place simultaneously throughout the solution space. Functionally, GA emulates the criteria of natural selection in a search procedure. Organisms in nature have certain characteristics that affect their ability to survive and reproduce. The characteristics are represented by a chromosome of the organism. In a reproduction process, the offspring's chromosomes will consist of a combination of the chromosomal information from each parent. Natural selection then

24

operates, such that fitter individuals will have the opportunity to mate most of the time. In turn this leads to the expectation that the offspring stand a good chance of being similarly fit.

Mutations can occur occasionally, enhancing the individual's characteristics and improving its progress to the next generation. Conversely, the chromosomal changes may have no effect at all. Without mutation, the population tends to converge to a homogeneous state wherein individuals vary only slightly from each other.

2.3.5 Elements of Genetic Algorithm

GA encodes the decision variables of the underlying problem into solution strings, called *chromosomes*, representing a candidate solution. Characters of the string are called *genes*. The position and the value in the string of a gene are called *locus* and *allele*, respectively. A *fitness function* is used to differentiate between the fit or unfit solutions. The fitness function may be represented in a mathematical formulation, which generates a differential signal in accordance with which GA guides the evolution through a series of solutions towards the final solution to the problem. GAs are powerful search mechanisms, which can traverse the solution space in search of optimal solutions. The *initial population* is randomly created or has prior knowledge about the problem. The individuals are evaluated to measure the quality of candidate solutions with a *fitness function*. In order to generate or evolve the offspring, genetic operators are applied to the current population. The genetic operators are *selection, crossover* and *mutation*.

2.3.5.1 Chromosomes

The most important step in applying a GA to a problem is to choose a way to represent a solution to the problem as a finite-length string or chromosome. In a concentrator network design problem, the chromosome might be a string of binary digits representing different aspects of a design. Figure 2-9 shows an example of a chromosome.
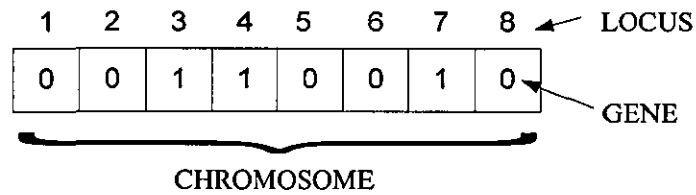
Figure 2-9 Chromosome Representation

The values on the chromosome may be arranged and interpreted as needed. They may represent Boolean values, integers, or even discrete real numbers. If the need arises to encode a real-value parameter, the encoding must make discrete the number for representation as a finite string.

2.3.5.2 Fitness Function

A function is needed that will interpret the chromosome and produce an evaluation of the chromosome's fitness. This function must be defined over the set of possible chromosomes and is assumed to return corresponding values representing the fitness of each chromosome. The definition of this function is crucial because it must accurately measure the desirability of the features described by the chromosome. In addition, the function must make this evaluation in a very efficient manner due to the large number of times the function will be utilised during the execution of the genetic algorithm.

2.3.5.3 Genetic Operators

These operators perform the selection, crossover, and mutation of chromosomes. These operators actually manipulate the individual chromosomes and must, therefore, be written with the underlying encoding of the chromosome in mind. For example, when defining an operator, such as the crossover which will actually perform the mating of two chromosomes, care must be taken to balance the mixing of gene values with producing feasible offspring and/or cost of repairing non-feasible offspring. Crossover doesn't always occur, rather it does so with some probability, P_c. A simple crossover operator is shown in Figure 2-10. In this example, a random point is chosen

between two genes, of the two parent chromosomes. Each of the chromosomes is cut at that point and the two ends are exchanged. This typically results in two different chromosomes with different characteristics. After crossover is completed, each position in the new chromosomes will be mutated, typically with a very small probability, P_m , forming another new chromosome. A small probability will result in little diversification but the important solution is not lost. The mutation is employed to ensure with some probability that the selection and crossover maintain useful genetic material. The new chromosomes resulting from the selection process, crossover process and mutation process are the offsprings of this mating, then evaluated and finally added to the population of the next generation.
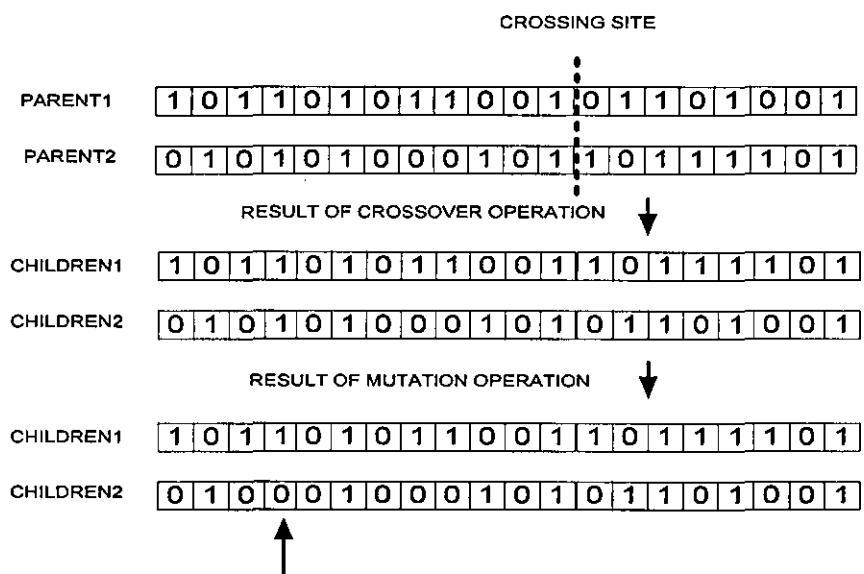
Figure 2-10 Crossover Operation

2.3.5.4 Initial Population

In a pure GA, the initial population is chosen randomly in order to select the chromosomes from all over the search space. Whatever genetic material is in the initial population, it will be the only material available to the genetic algorithm during its search. One might employ a heuristic to choose the initial population. However, this can lead to problems since a GA is an opportunistic

algorithm. The presence of just a few chromosomes which are fitter than all the others may cause a GA to converge prematurely to a local optimum.

2.3.6 Outline of GA procedure

Once the above components are specified, the basic genetic algorithm process proceeds as follows:

Step 1. *Initialisation.* Generate initial population of chromosomes at random or with prior knowledge into a matrix R.

Step 2. *Fitness Evaluation.* Evaluate the fitness of all chromosomes in matrix R. The fitness evaluation can be achieved by employing the objectives function which was designed.

Step 3. *Selection.* Select a set of promising candidates from matrix R. Select pairs of chromosomes using a random selection weighted by their fitness.

Step 4. *Crossover.* Perform crossover on the chromosomes pair in order to exchange information, from matrix R to generate a set of offspring O .

Step 5. *Mutation.* Apply mutation to the offspring chromosomes set O to obtain its perturbed chromosomes set O', with reasonably low probability.

Step 6. *Replacement.* Replace the current population R with the set of offspring O'.

Step 7. *Termination.* Check the termination criteria, if not met, then proceed to Step 2. Otherwise, terminate the process.

2.4 Summary

In this Chapter, a study of a number of existing QoS models for MANETs was presented. The protocol was classified according to the Cross-Layer QoS model, MAC Layer QoS model and QoS Aware Routing Protocols. In the Cross-Layer model, both INSIGNIA and SWAN represent a generalised model to fit into a wide class of applications. Both implementations are rather more complex. In MAC Layer QoS provisioning, IEEE802.11, Black-burst and MACA/PR was introduced as the most widely used MAC protocols. The network layer QoS model consists of CEDAR and TBP. The BE routing protocols DSR and AODV

was also reviewed. Both are the most widely used routing protocols. The fact that, there exist various heuristics for QoS route selection, such as QoS metrics ordering, sequential filtering, scheduling, admission control and control theory, was highlighted. GA was introduced as another method for calculating QoS route. The fundamental aspect of GA was also discussed. The next chapter describes the design and development of a QoS routing framework.

CHAPTER 3
QOS ROUTING FRAMEWORK FOR MANET

3.1 Introduction

Discrete event simulations of wireless communication networks are often computationally intensive and involve a considerable amount of time and effort to develop the coding. OPNET Modeler has a variety of tools, features, program modules and design techniques to help model developers to decrease substantially the time required to code the algorithm and run the simulations. Using the facilities and the available modeling techniques, a framework has been created for the simulation of the QoS routing protocol for MANET. The protocol is termed as QoS Routing for MANETs Using Genetic Algorithm and its acronym as QOSRGA. Basically, QOSRGA consists of three cooperative protocols to support QoS routing. These three cooperative protocols are: (1) Non-Disjoint Multiple Routes Discovery (NDMRD) protocol which is described in this Chapter; (2) Node State Monitoring (NSM) protocol which updates, accumulates and disseminates the QoS routing parameters. It consists of (a) monitoring and measurements of node connectivity index (*nci*) described in Chapter 4 and (b) node bandwidth monitoring and measurements which are described in Chapter 5 and (3) QoS Route Selection Algorithm using GA described in Chapter 6. Our

approach in designing QoS routing protocol is to initially establish multiple routes between source and target. Each route has a certain amount of resources and constraints due to node mobility. Every resource and constraint was recorded as Node State and was then used later to evaluate each route using Genetic Algorithm. The protocol is expected to improve the performance of MANETs supporting real-time applications, as compared to non-QoS routing protocols such as BE-AODV and BE-DSR.

The remainder of this chapter is organised as follows. Section 3.2 describes the overall framework for MANET routing protocol development using OPNET [29]. Section 3.3 presents the description of QOSRGA protocol and illustrates with the corresponding flowchart. Section 3.4 presents the Non-Disjoint Multiple Routes Discovery protocol. Section 3.5 describes the performance evaluations of NDMRD protocol. Finally, Section 3.6 summarises the chapter.

3.2 Framework for MANET Routing Protocol Development

3.2.1 Mobile Ad Hoc Networks Modelling

OPNET Modeler is a discrete event simulator which is able to simulate various kinds of wireline and wireless network in a number of configurations. It is an 802.11 compliant MAC layer implementation which is necessary for this work. The design of networking algorithm by reusing many existing components and modules. Most of the protocol design is implemented through a hierarchical graphic user interface. The design process goes through the following phases. Firstly, it need to choose and configure the node models useful in the simulations, such as a wireless node, a workstation, a router or an ftp server. Next, the different entities are to build and organise the network by connecting them. The last step consists of selecting the statistics that need to be collected during the simulations.

A new process model which performs all the QoS routing function was created. The function was registered as a child to MANET manager process model, *manet_mgr*. The MANET manager process model is included in Opnet Modeler and it is a child process of the standard IP node model. A process model is described in the form of Finite State Machine. The challenge is to build

this FSM for each level of the protocol stack from scratch using pseudo-coded algorithm. It is possible to reuse a lot of existing components such as MAC layer, transceivers, links and modules to accelerate the development process.

The mobility of nodes is an important metric when designing and evaluating MANET. The goal is to use a mobility model which can be varied in a controlled way and observe the effect on the performance of our QoS routing. The mobility models that are available in the Opnet Modeler are the random waypoint model (RWP) and trajectory (TRJ) model. The RWP mobility model was used in most of the literature. In the TRJ model, the path was set unique for each node, such that it moves in a straight line, at an angle or with any number of turns.

OPNET Modeler provides a comprehensive development environment for the specification, simulation and performance analysis of communication networks. A large range of communication systems from a single LAN to global satellite networks can be supported. Discrete event simulations are used in the analysis of system performance and their behavior. The key features of OPNET Modeler are modeling and simulation cycle, hierarchical modeling and the provision of a specialised communication networks simulation environment.

3.2.1.1 Hierarchical Modelling
OPNET provides four tools called editors to develop a representation of a system being modeled. These editors, the Network, Node, Process and Parameter Editors, are organised in a hierarchical fashion which supports the concept of model level reuse. Models developed at one layer can be used by another model at a higher layer. Figure 3-1 illustrates this hierarchical organisation. The following sections introduce each of the modeling domains.

3.2.1.2 Network Model
Network Editor is used to specify the physical topology of communication networks. It defines the position and interconnection of communicating entities which consists of nodes and links. The specific capabilities of each node are realised in the underlying model with a set of attributes attached to each model.

32
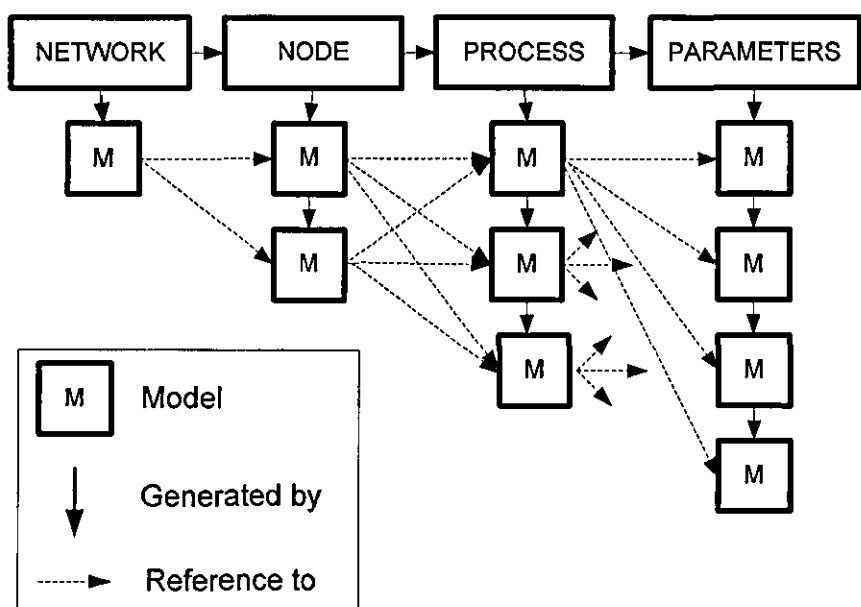
Figure 3-1 Hierarchical Organisation of Editors

3.2.1.3 Node Model

Node models are expressed as interconnected modules of two distinct categories. The first module set has predefined characteristics and a set of built-in parameters such as packet generators and radio receivers. The second sets are programmable modules, referred to as processors and queues. Each node is described by a block structured data flow diagram in which its functionality is defined by a Process Model. Packets are transferred between modules using packet streams and numeric signals and are conveyed by statistic wires. A node is either fixed or mobile.

3.2.1.4 Process Model

Process models are used to describe the logic flow and behavior of processor and queue modules. Interprocess communication is supported by interrupts. Process models are expressed in Proto-C language, which consists of state transition diagrams, a library of kernel procedures and the standard C programming language. The Process Editor uses a powerful state-transition diagram approach to support specification of any type of protocols, resources, applications, algorithms or queuing policy. States and transitions graphically define the progression of a process in response to events. Within each state,

general logic can be specified using a library of predefined functions and standard C language. A parent process may create new processes referred to as child processes, to perform subtask operations.

3.2.1.5 Simulation Design and Data Analysis

In Opnet Modeler, the network models emulate the multi-layered IP network protocol. Each node performs a particular service within its layer. After completion, the packets are pushed out to the next IP layer, either Ipv4 or Ipv6. Hence, it is required to follow strictly the process of developing the routing protocol. In order to manage these protocols effectively, a *manet_mgr( )* module is included to interface between the IP node model and the MANET routing protocol. The protocol designers based their work on this model and could abide strictly to the programming styles. In the next section, the models and the modification for the purpose of designing new QoS routing is described.

3.2.1.6 Node Model and Process Model for MANET

Node models for MANET consist of Manet Station, Manet WLAN Station and WLAN Servers. These node models can be used to generate raw packets over IP over WLAN. They can function as a traffic source or destination and can be configured to run the routing protocol such as our QoS Routing protocol (QOSRGA) and other protocols such as BE-DSR and BE-AODV. Figure 3-2 shows the node model of MANET station. The node model is used to run the routing protocol, which is based on WLAN models, IEEE802.11. Wireless LAN Workstations and Servers can run all applications such as FTP, HTTP, emails, voice and video over TCP/UDP over IP over WLAN. These nodes can be configured to run basic routing protocol and also our QoS Routing Protocol. The node model is made up of various process models, which had to be modified appropriately to suit the design of our QoS Routing protocol and be able to communicate seamlessly with the IP protocol stack. The modifications were done at the MAC layer, IP and Traffic Source process model. The most vital process model is the CSMA/CA, which represents the 802.11 MAC layer protocol, illustrated in Figure 3-3. Basically, it is a single channel contention mechanism. It is necessary to extract the signal strength in terms of power

34

received triggered by an arrival of packet from a neighbouring node. The power received detected within the physical layer is captured here and transferred to the network layer and then to the QOSRGA module, where further processing is performed.
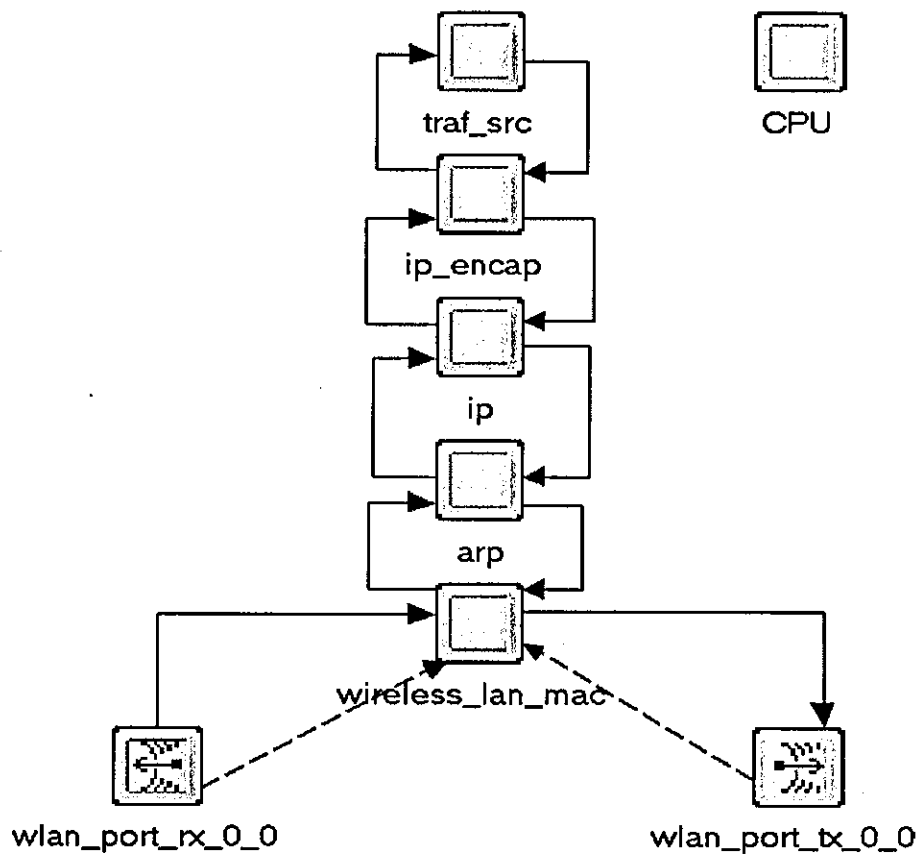
Figure 3-2 Node Model For Manet Station[29]
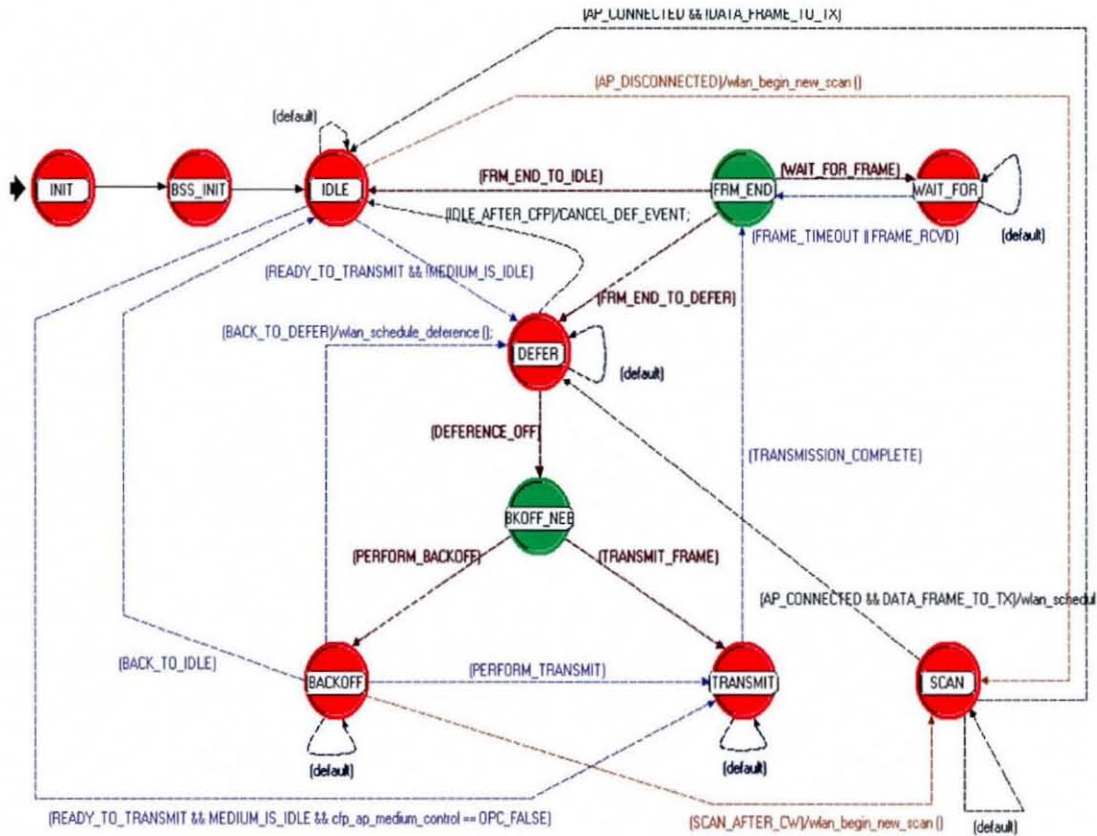
Figure 3-3 WLAN MAC Process Model [29]

Within the FSM, interrupts occur at all the states mention above. These interrupts are serviced by the interrupt process module: *wlan_interrupts_process()* module. The function of the *wlan_interrupts_process()* is to handle the appropriate processing need for each type of remote interrupts. The types of interrupts are: (1) stream interrupts, from lower layer and from higher layer, and (2) statistics interrupts, from transmitter and receiver module at physical layer.

In our context, the most useful interrupts are from the lower layer. It is here that the signal strength received from the transmitting node could be captured. If the event is the arrival of the packet from the physical layer, then there is a need to process the stream interrupt. The module *wlan_physical_layer_data_arrival()* will process the stream interrupt. The function of the module is to process the frame received from the physical layer,

36

decapsulate the frame and set appropriate flags if the stations need to generate a response to the received frame. The power received is measured in this module and sent to the network layer.

The IP dispatch process model is shown in Figure 3.4. It initialised all aspects of transition between ARP protocol layer and the IP encapsulation process. The process registers all the attributes into the process registry which consist of subnet information: ip protocol, address information, multicast address, radio interface and module information. One of the functions of the *ip_dispatch( )* model is to register all the protocols that are going to be operated in this node. In this context, the network layer protocol in which there is interest is the MANET routing protocol, which is embedded within the IP dispatch process model. The QOSRGA routing protocol is designed as a child process of *manet_mgr( )* process model.

The IP dispatch process model checks whether the MANET routing protocol is enabled by inspecting the attributes of the node. Then one invokes the MANET manager to spawn the QoS routing protocol. The action results in the packet being sent to the MANET protocol child process. After the *manet_mgr()* has dealt with the packet, it sends back to the *ip_dispatch()* for it to be forwarded to the upper or lower layer as appropriate. Accordingly, the *ip_dispatch()* process model act as the IP interface to our QoS routing Protocol and other MANET routing protocol. This gives an easier way of accommodating the protocol design without having to be involved in the coding and the intricacies of the IP layer and the MAC layer.
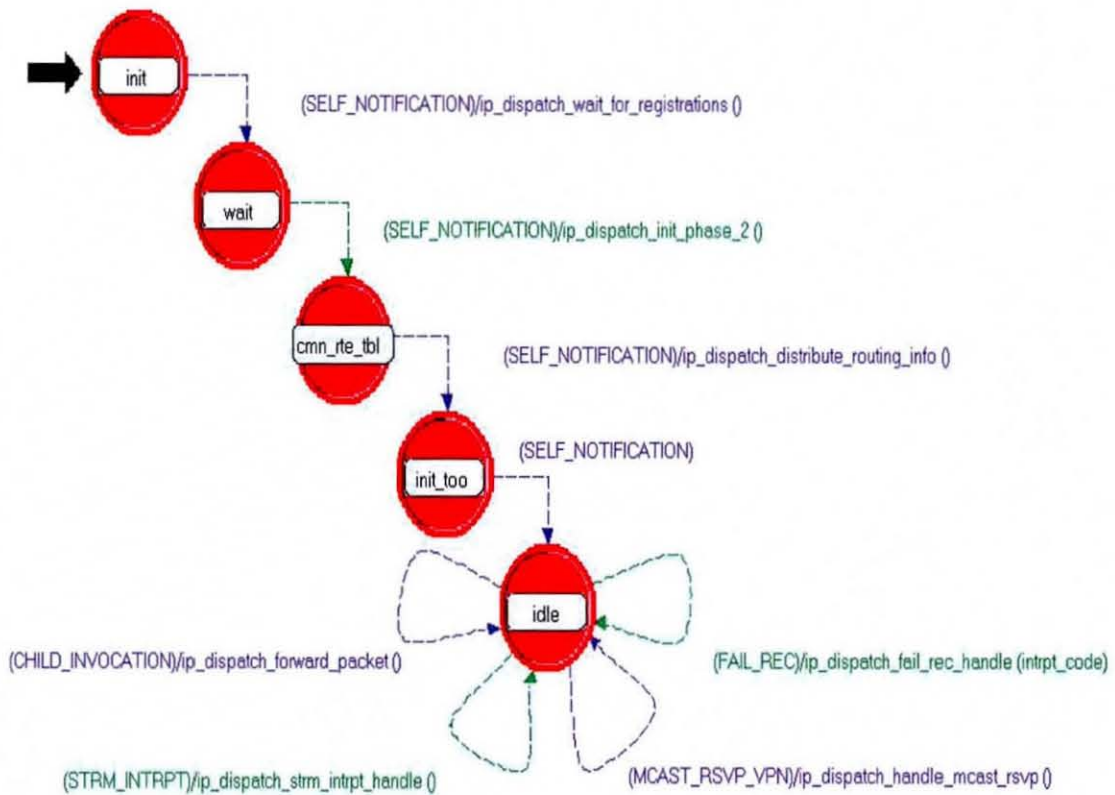
Figure 3-4 IP Dispatch Process Model [29]

The *manet_mgr()* process model, shown in Figure 3.5, is a child process of IP dispatch. It consists of an init state and wait state. It begins with the initialisation of the state variables. Then it determines the type of MANET routing protocol running on this node. All interfaces that have MANET enabled should have the same MANET routing protocol on them. The routine, *manet_mgr_routing_protocol_determine()*, create and invoke the appropriate MANET routing protocol that has been configured. The routine, *manet_mgr_routing_process_create( )* , spawns the appropriate MANET routing protocol as a child process of *manet_mgr()*. The process model then exits the wait state on receiving the invocation from another process. It checks whether the process is invoked from the child process or from the parent process.

38
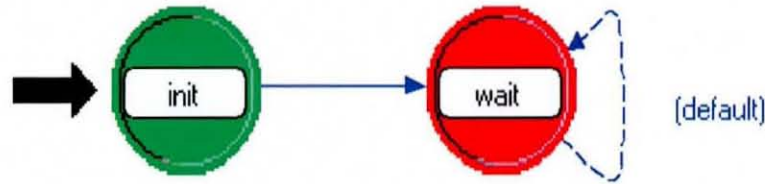
Figure 3-5 Manet Manager Process Model

If this is an invocation of parent process from the child process, it passes the invocation to the IP process. On the other hand, if this process was invoked by the *ip_dispatch*, it then passes the invocation to the child process, that is the MANET routing process model.

3.2.2 Implementation

The MANET routing protocols were embedded in the MANET stations, routers, workstations and server nodes. Taking advantage of the Opnet open source programming modules and the existing model setup of Opnet, the key stages are followed rigorously, identifying the modules and process models. It must be ensured that the process model can communicate with the IP layer above the Network Layer and also with the MAC Layer. In maintaining consistency with the IP layer and the *manet_mgr*, the following steps outline the modifications that need to be done within the node and the process model.

Step 1:

The header files, ip_higher_layer_proto_reg_sup.h and ip_rte_v4.h must first be edited to include the following enumeration.

ip_higher_layer_proto_reg_sup.h → IpC_Protocol_Qosrga = 400

ip_rte_v4.h → IpC_Rte_Qosrga

Step 2:

The attributes of the Qosrga routing protocol is added under the heading of the Ad Hoc Routing Parameters in the *manet_mgr()* process model. The value of the attributes can then be edited in the network model. This is achieved by opening the *manet_mgr()* process model and proceeding to: *Interfaces -> Model Attributes -> Edit Properties -> Edit Compound Attributes.* All the properties of the QOSRGA can then be added and edited. Before running the simulation, all the parameters that will affect the network model can be edited and the effect can be observed. All the parameters are similar to other BE routing protocols, except the GA parameters which are the mutation rate, crossover rate and population size, maximum number of generation and selection method. These attributes would be parsed by the QOSRGA routing protocol process model to be the input to the genetic algorithm.

Step 3:

The protocol name "Qosrga" need to be included in the following functions within the *manet_mgr()* process model:

(1) *manet_manager_routing_protocol_determine( )*,

(2) *manet_manager_routing_process_create( )*.

Subsequently, the new routing protocol should be declared as the child process of the *manet_mgr()*.   The process model for the Qosrga routing is shown in Figure 3-6.

Step 4:

The protocol name "Qosrga" should be added to the routing protocol list and then set manet_enable attribute to TRUE in the following functions within the *ip_dispatch* process model:

(1) *ip_rte_proto_string_parse( )*,

(2) *ip_interface_routing_protocols_obtain( )*,

(3) *ip_dispatch_init_phase_2 ( )* .

It will invoke the program to spawn the relevant type of routing protocol.

Step 5:

The most important aspect when developing the MANET routing protocol is the ability of the protocol module to communicate with the IP layer. The general formulation is that the protocol is developed as a child process of the IP layer process model, which is the *manet_mgr( )* which interfaces the protocol to the IP protocol stack. All the available functions within the manet manager are used, to communicate with the IP layer, whether it is an upper layer or the lower layer. The communication has to be made transparent to the IP layer and that all of the QOSRGA packet is well embedded in the IP packet structure. All MANET protocols use IP's addressing and forwarding capabilities. This ensures that the protocol works seamlessly with the IPv4 protocol standard. All the functionalities of Ipv4 were carried out in the external file *ip_rte_support.ex.c.* Within this file, the following functions are to be modified:

(1) *ip_rte_packet_arrival()* ;

(2) *ip_rte_datagram_dest_get()* ;

(3) *ip_rte_packet_send()* .

The interfacing from the IP protocol stack to the QoS routing module is done in the process model, ***ip_dispatch()*** which is really the process model for reading and invoking the child process. The following modules need to be modified:

(1) *ip_dispatch_cleanup_and_create_child_process()* to direct to the *manet_mgr* process model if the node is MANET enabled;

(2) *ip_interface_routing_protocols_obtain()* which looks for routing protocols that were set up at this node. It returns the list of routing protocols set for the node.

3.2.3   QoS Routing Protocol Design

3.2.3.1 Process Model for QOSRGA

The design of the QoS routing process model or QOSRGA is shown in Figure 3-6. It consists of an INIT STATE and a WAIT STATE. The event of

PACKET_ARRIVAL will cause the execution of QOSRGA modules until the end, when it returns to the wait state. The event CONN_EXPIRED will cause the connectivity packet being broadcast.
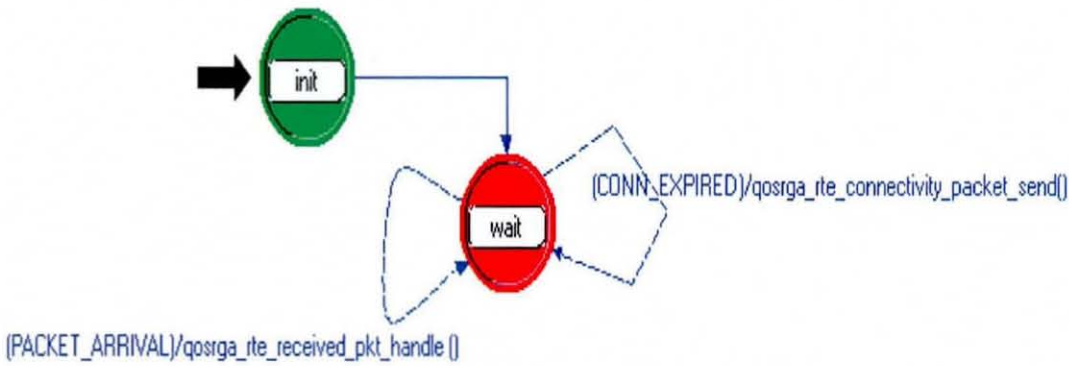
Figure 3-6 QOSRGA Process Model

The design of QOSRGA Model architecture consists of a process model, *qosrga_rte* and *external_files*. In this section all the modules needed for the implementation of QOSRGA protocol are listed as the module names and their function stated. The *qosrga_rte* process model consists of the following sections: (1) Initialisations (2) Packet Arrival and processing (3) Packet Creation and Sending (4) Tabulating Of Routes (5) Interfacing With IP (6) External Support Files (7) Node State Monitoring Module (8) Genetic Algorithm Module.

3.2.3.2 The QOSRGA Message Format

All the control packets and data packet of QOSRGA are embedded within the IP datagram. The IP datagram packet header is shown in Figure 3-7. The QOSRGA Packet Header is shown in Figure 3-8. It consists of <*next header*>, <*reserved*>, <*payload length*>, <*options*> and <*data*>. The type of control messages for the protocol will be residing within the <*options*> in the QOSRGA packet format.

IP Header	QOSRGA Pkt Header	IP Payload

Figure 3-7 Datagram Header
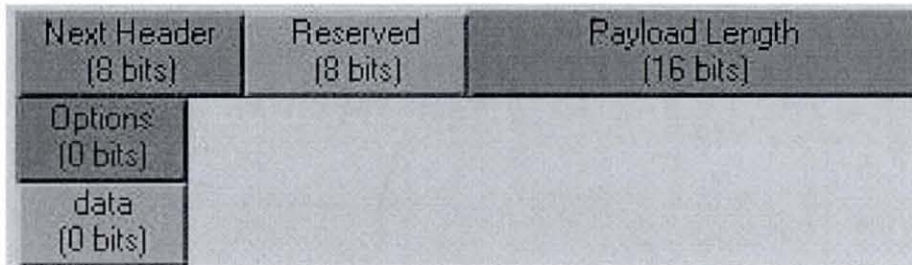
Figure 3-8 Fixed Portion of QOSRGA Packet

There are four types of control messages (RREQ, RREP, RERR and CONN) and a type of data message in QOSRGA.

- **RREQ** (*Type, Length, Identification, targetAddr, flowID, bwDemand, route_lptr*): RREQ represent a route request packet, broadcasted from source, expected to arrive at target, *targetAddr*. The field < *Type*> is the type of message. The <*Length*> is the length in bytes of the option fields excluding Type and Length; < *Identification* > is a unique value ID generated by the initiator of the RREQ; < *targetAddr* > is the address of the node that is the destination of the RREQ; < *route_lptr* > is the list of nodes traversed by the RREQ packet.

- **RREP** (*Type, Length, reply_id, last_hop_external, route_lptr, node_state_bwa_lptr, node_state_dly_lptr, node_state_mdl_lptr, node_state_nci_lptr*): This is a route reply message from destination to source along the reverse route that was discovered by RREQ packet, pointed by the list pointer, *route_lptr*. The <*Type*> is type of the message; <*Length*> is the length of options; <*Reply_id*> is the ID of the RREP,

43

which is equal to the ID of RREQ which cause this RREP; < *route_lptr* > is the list of nodes traversed by this RREP, reverse to the one traversed by RREQ; <*last_hop_external*> is the arbitrary path outside MANET environment; <*node_state_bwa_lptr*> is the list of bandwidth available from the nodes traversed; <*node_state_dly_lptr*> is the list of end to end delay from the nodes traversed; <*node_state_mdl_lptr*> is the list of medium access delay from the nodes traversed; <*node_state_nci_lptr*> is the list of **nci** from the nodes traversed.

- **RERR** (*Type, Length, error_source_address, error_dest_address, unreachable_node_address*): RERR represent the error message packet from the node that detects the node unreachability. The <*error_source_address*> is the node originating the error message; <error_dest_address> is the address where the error must be delivered; <*unreachable_node_address*> is the address that the packet should be sent.

- **CONN** (*Type, Length, Identification, targetAddr*): This is a connectivity packet broadcast every 1 second, if there are no packet transmission activities 1 second after the last activity. Its operation is similar to RREQ packet but the TTL is always set to 1. The <*Type*> is type of the message; <*Length*> is the length in bytes of the option fields excluding Type and Length; <*Identification*> is a unique value ID generated by the initiator of the CONN; <*targetAddr*> is the address of the node that is the destination of the CONN.

- **DATA** (*Type, Length, first_hop_external, last_hop_external, segments_left, route_lptr*): The DATA represent the data packet that is delivered from or to the upper layer. The <*Type*> is the type of packet; <*segments_left*> is the number of segments remaining; <*route_lptr*> is the sequence of address of nodes that needs to be traversed.

3.2.3.3 Data Structures

Each node of the ad hoc network keeps and maintains a **Forwarding Request Table, Originating Request Table, Neighbours Table, Route**

44

Table and **Node State Table**. Forwarding Request Table is used to record the information about forward request of the RREQ packets. Originating Request Table is used to record the information about the RREQ that have been originated by this node. Neighbours Table includes an updated list of its neighbours, which is done periodically. The Route Table maintains an updated list of all the possible routes to the required destinations. The Node State Table is where all the information regarding the Node State is stored.

3.3 QOSRGA Protocol Description

This section presents an overall description of the QOSRGA protocol.

3.3.1 Overall Flowchart

The overall flowchart consists of the following operation: (1) initialisation of all the variables, tables and statistics; (2) initialisation of the periodic transmission of connectivity packet (CONN), periodic node bandwidth and medium access delay monitoring; (3) node state monitoring; (4) processing of lower layer packet arrival; and (5) processing upper layer packet arrival. The overall flowchart of the protocol is shown in Figure 3-9.

3.3.1.1 Node Connectivity Packet Transmission module

At the startup operation, all nodes are stationary and no connectivity with its neighbour occurs. The Node Connectivity packet functions as a wireless connector, which ensures there exists a means to know there are neighbours around the node. CONN packet transmission is only active when there is no normal transmission session. The protocol creates the QOSRGA packet type RREQ but with identification set to '0'. It is then encapsulated into IP datagram with TTL set to '1', signifying a single hop transmission, before being broadcasted. Hence any node receiving the CONN packet must update their Node State. The node stops transmitting the CONN packet when it senses some real QOSRGA packet.

3.3.1.2 Periodic Node Bandwidth and Medium Access Delay Calculation module

When dealing with QoS routing, all the QoS parameters including the instantaneous node bandwidth and medium access delay need to be measured.

45

The bandwidth measurement detailed is elaborated in Section 5.4. From Figure 3-9, the node bandwidth availability is monitored and recorded in each sampling interval, set at 20 ms which coincide with the Beacon transmission rate. The actual operations of capturing the bandwidth and delay measurement were done at the MAC layer.

3.3.2 Process the lower layer packet arrival

The lower layer packet arrival consists of the arrival of packets from neighbouring nodes. All packets that have arrived at the lower layer will be monitored. All the relevant information due to the arrival of the packets are extracted and cached. The type of packets that arrived are the RREQ, RREP, RERR, CONN and DATA packet. The protocol is tested for each of them. Each of them would be processed appropriately, as shown in Figure 3-9.

3.3.3 Process the upper layer packet

When a new application packet arrives at the network layer, the protocol determines whether any routes are available to the packet's destination in the node's routing table. If no route exists, then the route discovery is performed by initiating a route request process. If there are multiple routes in the routing table, one performs the GA route selection calculation if the last GA route calculation is more than Route Accumulation Latency. Route Accumulation latency is the time delay between the last route request sent and last route reply packet received. By setting the Route Accumulation Latency to a larger value, more routes can be accumulated, but incur an additional delay. So, the value must be well chosen. The flowchart for the processing of the upper layer packet is shown in Figure 3-10.

46
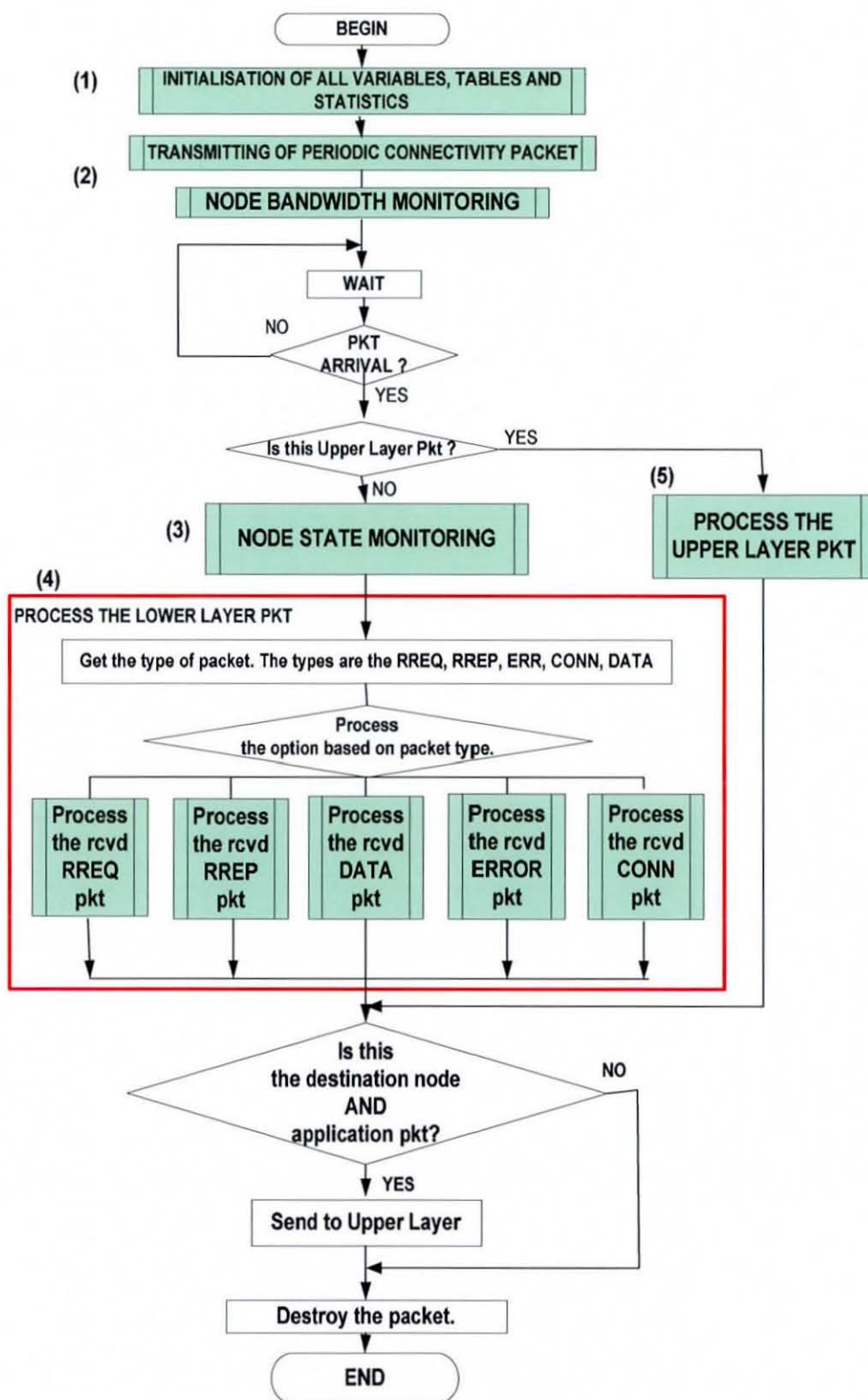
Figure 3-9 Overall Flowchart of QOSRGA
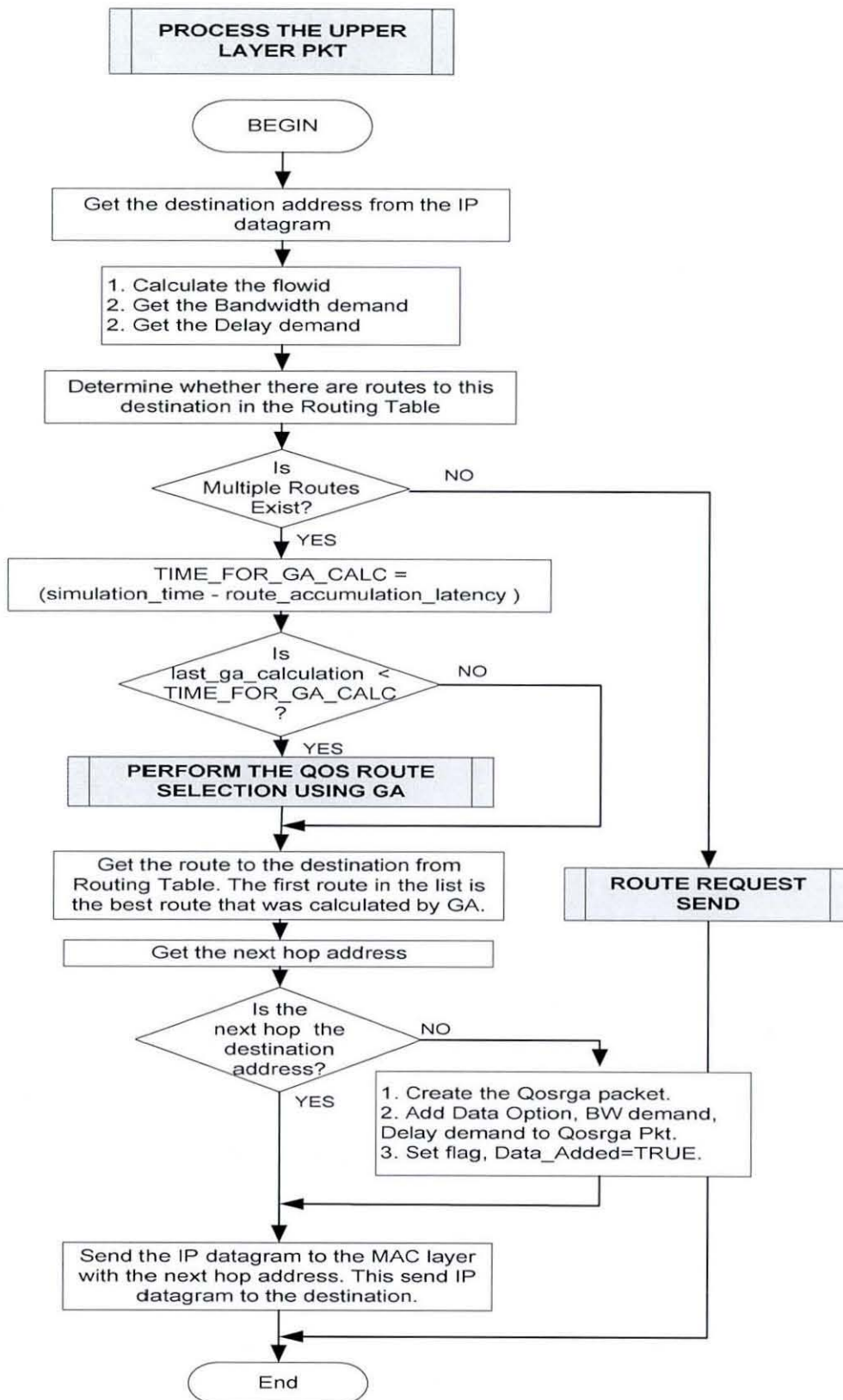
Figure 3-10 Process the upper layer packet

3.4 Non-Disjoint Multiple Routes Discovery Protocol

3.4.1 Non-Disjoint Multiple Routes vs Disjoint Multiple Routes

Disjoint routes are those routes where there are no common nodes in their entire routes except the source and destination nodes. Most of the previous works on MANET multiple paths has restricted the number of potential routes to a small number, typically two. AOMDV[30] allows up to k-link-disjoint RREPs, where one is the quickest path and the others are chosen from the next link-disjoint RREQs. SMR[30] builds two paths from the quickest RREQ and then collects the RREQs for a period and chooses a second maximally disjoint path from the first. The QoS routing protocol for MANET should exploit fully the rich connectivities of the multiple routes network by considering non-disjoint routes, to improve the reliability of packet delivery.

Definition 3.1: A set of **node non-disjoint routes** is defined as routes which exist such that an intermediate node is a member of at least two different routes simultaneously.

Consider a set of **m** valid routes from source to target as $P_{ST} = \{ P_0, P_1, P_2, ...P_{m-1} \}$. Each route consists of a collection of nodes, $P_i = [n_0, n_1, ..., n_i, ..., n_{k-1}]$. A node non-disjoint set of routes is said to exist if n_i, which is not the source or target, is a member of at least two different routes simultaneously. Each route then must have at least one node that is common to any other routes in P_{ST}.
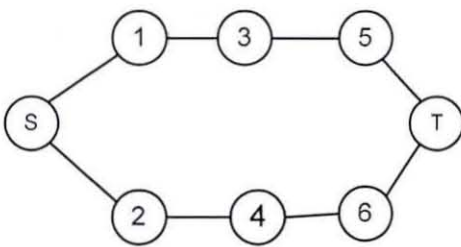
Figure 3-11 Disjoint Network
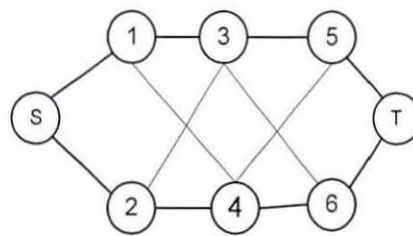
Figure 3-12 Non-Disjoint Network

49

Consider the networks in Figure 3-11 and Figure 3-12. Figure 3-11 shows a disjoint **S-T** connection and Figure 3-12 of the network shows rich node non-disjoint connectivity. The disjoint network has two minimum paths, {S,1,3,5,T} and {S,2,4,6,T}. There are no common nodes except the source and target. In Figure 3-12, the non-disjoint network has 8 minimum paths, { S,1,3,5,T} , {S,1,4,6,T}, {S,1,3,6,T}, { S,1,4,5,T}, {S,2,4,6,T}, { S,2,3,5,T }, {S,2,3,6,T } , { S,2,4,5,T} . One of the characteristics of a non-disjoint network is that there exist a number of common nodes, excluding the source and target nodes. In this thesis, it is specified that there must be at least one common node excluding source and target. The non-disjoint multiple routes is still valid if there exist common links. If each node has different node qualities in terms of Node State, then the combination of all nodes in the route could produce a measure of route quality. In the non-disjoint network the probability of selecting the most reliable routes, is increased. Most of the routing protocols dealing with multiple paths [31] only utilised the disjoint networks. In Chapter 2, it was stated that Node State was the foundation of QOSRGA routing protocol. By combining the Node State information for each node in the routes and the routes discovered within the node non-disjoint network, there will be a rich choice of possible source-target routes. It is in this respect that the Genetic Algorithm would be used to find the most suitable routes with several simultaneous constraints as opposed to sequential filtering, metrics ordering or rescheduling principles. This would be consistent with the work on the design of QoS routing. Hence a Non-Disjoint Multiple Routes Discovery (NDMRD) protocol is proposed as part of the QoS routing (QOSRGA) protocol.

3.4.2 Multiple Routes Discovery and Route Accumulation

When a source node wants to communicate with a destination node, it checks its route table to confirm whether it has a valid route to the destination. If so, it sends the queued packets to the appropriate next hop towards the destination. However, if the node does not have a valid route to the destination, it must initiate a route discovery process. To begin such a process, the source node creates a **RREQ** packet.

The first aim of the NDMRD protocol is to generate a set of source-destination routes which are node non-disjoint. The set of routes are to be accumulated at the source node. To achieve this aim, each node must be set to receive duplicates of **RREQ** packet. If each node is set to only allow one duplicate of RREP to be forwarded, then it would have a set of routes that are disjoint. To achieve non-disjoint routes, many duplicates could be allowed to be received and forwarded by the node. For example, consider a scenario in Figure 3-13 consisting of five nodes. **S** is the source and **T** is the target node where {1, 2, 3} are neighbours to S, {S, 2, T} are neighbours to node 1 and also to node 3. Node **S** is out of range to **T** and 3 is out of range to 1.
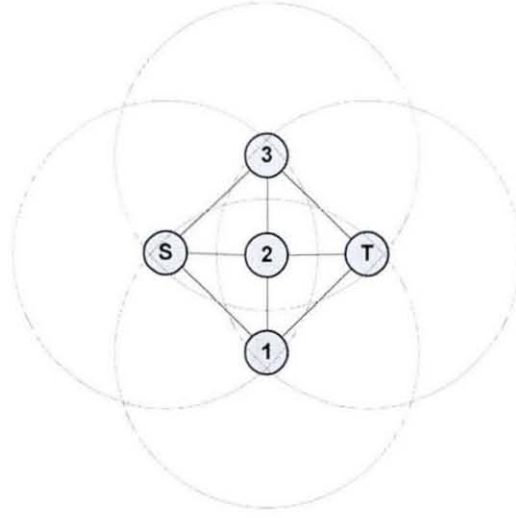
Figure 3-13 A Five Nodes Scenario

Definition 3.2 : The **RREQ duplicate** is defined as the number of times the **RREQ** packets from a particular source with the same identification arrived at an intermediate node and were forwarded.

To initiate the route discovery, node **S** starts transmission of packet **RREQ**. It is assumed, that the maximum duplicates for **RREQ** is set to 10. When **RREQ** is generated at **S**, the packet is then broadcasted. Node 1, node 2 and node 3 will receive the packet and forward it to node **T**. When the **RREQ** packet has arrived at node **T**, **RREP** packet is then generated and sends in reversed direction to the route list of the **RREQ** packet. After a time, τ the source node, **S** produces a set of non-disjoint routes as { [S-3-T] , [S-2-T] , [S-1-T] , [S-3-2-

51

T] , [S-3-2-1-T] , [S-2-3-T] , [S-2-1-T] , [S-1-2-T] , [S-1-2-3-T] }. It is observed that the **RREQ** packet goes through node 2 seven times. If we limit the number of duplicates to 5 for example, the number of non-disjoint routes returned will be 6. Figure 3-14(a) illustrates the process of route accumulation and Figure 3-14(b) shows the outcome as a set of routes in the node's Routing Table. The number of routes will also be reduced if the time τ is set less than before.

Definition 3.3 : The **Route Accumulation Latency** is defined as the amount of time allowed for a source node to accept a number of **RREP** packets destined to this node with the same **id** as that **RREQ** which originated from it. The symbol, τ represents the route accumulation latency. The value of τ chosen must ensure a good number of routes are accumulated. The number of **RREP** packets received represents the number of **node non-disjoint routes** discovered.

Hence there are two parameters that governed the number of non-disjoint routes to be accumulated: (1) the maximum number of RREQ duplicates allowed and (2) the route accumulation latency. The question is how to determine the value for the number of RREQ packet duplicates and the route accumulation latency ?

The second aim of the NDMRD protocol is to facilitate the functions of the Node State Monitoring protocol in the updating, disseminating and accumulating the QoS routing parameters. The Node State Monitoring protocol will be described in detailed in **Chapter 5**. Another example of generating multiple routes is shown in Table 3-1. It consists of a list of routes accumulated for a 10 nodes scenario as in Figure 3-15. For each route list, a different node combination is produced. It is obtained by capturing the route table at a given time. A simulation is done for 40s, offering a CBR load of 820 kbps with a single source and single target. This set of routes could be use as an initial population for route selection using Genetic Algorithm.

RREQ	S	S-1	S-1-2	S-1-2-3	

(circles: S — 1 — 2 — 3 — T with orange RREQ arrows forward and blue RREP arrows backward)

RREP	S-1-2-3-T	S-1-2-3-T	S-1-2-3-T	S-1-2-3-T

ROUTING TABLE ENTRY (after RREQ arrived)		S-1	S-1-2	S-1-2-3	S-1-2-3-T

ROUTING TABLE ENTRY (after RREP arrived)	S-1-2-3-T	S-1 / 1-2-3-T	S-1-2 / 2-3-T	S-1-2-3 / 3-T	S-1-2-3-T / T
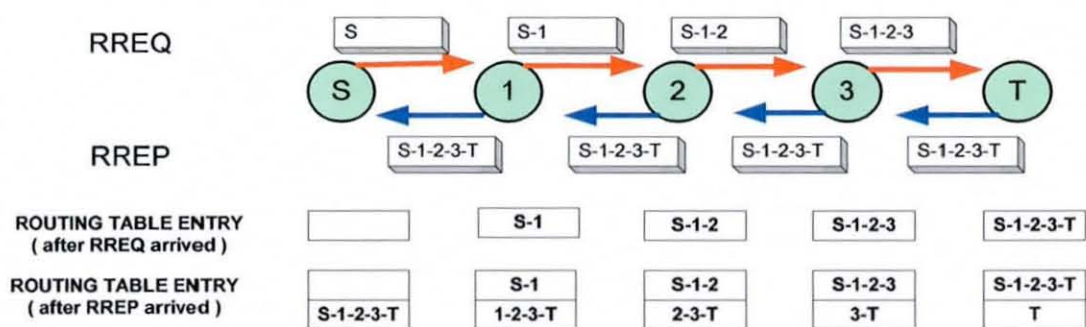
(a) Process Of Route Accumulation

S-1-2-3-T
S-1-2-T
S-1-T
S-2-T
S-3-T
S-2-1-T
S-2-3-T
S-3-2-T
S-3-2-1-T
S-1-2-3-T

(b) After Stepping Through For the Whole Route Discovery Process, The Routing Table Of Source Node Is Shown Here.

Figure 3-14 Route Accumulation at the Source Node

S=1
T=10
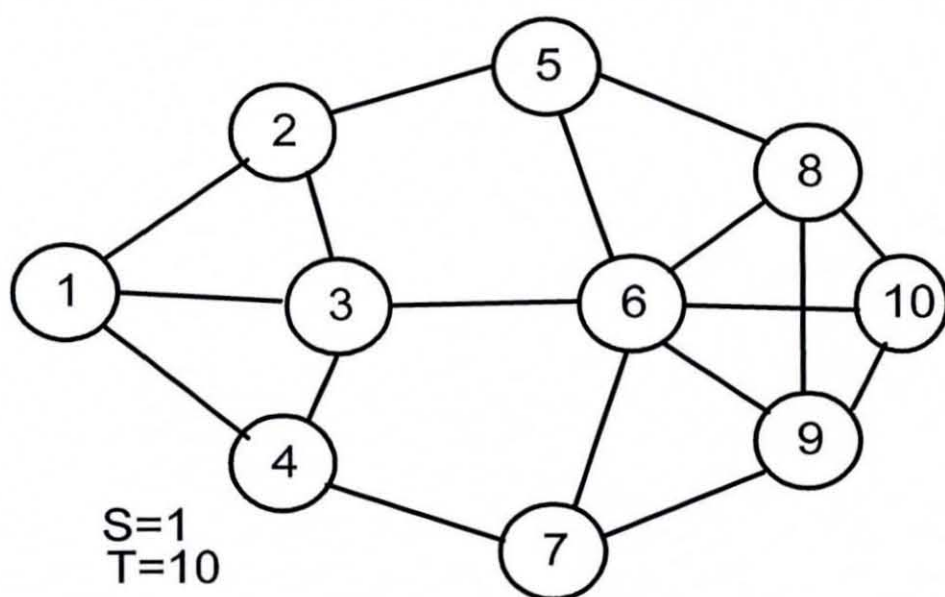
Figure 3-15 A Ten Nodes Static MANET

Table 3-1 Example list of routes for a single source single target

Time	List Of Routes Discovered
13.405	192.0.0.1->192.0.0.4->192.0.0.3->192.0.0.6->192.0.0.10
13.405	192.0.0.1->192.0.0.2->192.0.0.5->192.0.0.6->192.0.0.10
13.405	192.0.0.1->192.0.0.2->192.0.0.5->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.9->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.6->192.0.0.10
13.405	192.0.0.1->192.0.0.3->192.0.0.6->192.0.0.9->192.0.0.10
13.405	192.0.0.1->192.0.0.2->192.0.0.5->192.0.0.8->192.0.0.6->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.9->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.2->192.0.0.5->192.0.0.8->192.0.0.9->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.9->192.0.0.6->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.6->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.3->192.0.0.4->192.0.0.7->192.0.0.6->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.3->192.0.0.6->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.6->192.0.0.9->192.0.0.10
13.405	192.0.0.1->192.0.0.3->192.0.0.6->192.0.0.9->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.2->192.0.0.5->192.0.0.8->192.0.0.9->192.0.0.6->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.9->192.0.0.8->192.0.0.6->192.0.0.10
13.405	192.0.0.1->192.0.0.3->192.0.0.4->192.0.0.7->192.0.0.9->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.6->192.0.0.9->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.7->192.0.0.9->192.0.0.6->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.3->192.0.0.6->192.0.0.8->192.0.0.9->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.3->192.0.0.6->192.0.0.5->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.4->192.0.0.3->192.0.0.6->192.0.0.9->192.0.0.8->192.0.0.10
13.405	192.0.0.1->192.0.0.2->192.0.0.5->192.0.0.8->192.0.0.9->192.0.0.7->192.0.0.6->192.0.0.10

In this route discovery procedure, the source node transmits a **RREQ** packet, identifying the target for which the route is needed. An intermediate node receiving the **RREQ** packet retransmits the packet if it has not yet achieved maximum allowed **RREQ** duplicates, then forwarded a copy of it. When the target node has received the RREQ packet, it returns a Route Reply (RREP) to the source. The RREP packet then traverses the route taken by the RREQ, in the opposite direction of the RREQ and is propagated towards the source.

Certain optimisations have been defined for this basic NDMRD discovery scheme to reduce the frequency of performing route discovery and to limit the flooding of the network by forwarding the RREQ mechanism. These are: (1) sending the periodic CONN packet at the interval rate of one per second. This will ensure nodes within the transmission range were connected as neighbour nodes and to activate the node state cache, while nodes that have moved out of range will have their node state cache deactivated, avoiding stale information. When the source node starts transmitting RREQ packets, the sending of periodic CONN packets is stopped until the transmission session is completed or the route is broken. (2) The sending of periodic CONN packets with the setting of TTL to 1, to limit the flooding of CONNs over the network.

For each individual route discovery attempt, each node forwards the RREQ base on the following conditions: (1) should this be the first occasion that RREQ has arrived at the node, then it forwards the packet to the next one-hop node accordingly; (2) if it is not the first time, then this is a RREQ duplicate. The RREQ duplicate counter is then incremented and, if the counter is more than the maximum number of RREQ duplicates allowed, the RREQ packet is destroyed; and (3) if no more hop, the RREQ packet is destroyed.

Each route discovered is constrained by the *node bandwidth, end-to-end delay, medium access delay* and *route lifetime*. The route lifetime is indicated by the *node connectivity index (**nci**)* in the form suitable for QoS routing determination. The determination of Node Connectivity Index will be described in **Chapter Four** whereas the delay and bandwidth available is described in **Chapter Five**.

3.4.3 NDMRD Protocol Implementation

The Opnet implementation of NDMRD protocol consists of Send Route Request module, Send Route Reply module, Received Route Request module and Received Route Reply module. The implementation of NDMRD is best explained by the use of a flowchart. Each individual module will be described as follows.

3.4.3.1 Send Route Request module

The RREQ packet is activated when event packet arrival has occurred. When a packet arrives from upper layer, it is an application packet. The protocol then checks the Routing Table and determines whether any route exists to the destination. If no route exists, then the RREQ packet is initiated. It creates the QOSRGA packet, issues an id, encapsulates the packet into IP datagram, sets TTL and lastly sets the Originate Request Table. The IP encapsulated RREQ packet is then broadcasted. The flowchart of the Send Route Request module is shown in Figure 3-16.

3.4.3.2 Received Route Request module

The Received Route Request module is activated when type RREQ packet arrives at the node. If the node is an intermediate node, the packet is checked for RREQ duplicates, using the list in the Forward Request Table. Then, the packet is registered and allowed for up to a maximum number of the RREQ duplicates. This will result in the creation of non-disjoint routes. If TTL is 1, destroy the packet, otherwise set the entry to Forward Request Table. Then rebroadcast the RREQ, with this node address add to the address list of RREQ packet. If the node is a source node, then destroy the packet. This means that the node is receiving its own RREQ packet. If the node is the destination of the packet, then initiate a Send Route Reply operation. The overall operation of the Received Route Request operation is shown as a flowchart in Figure 3-17.

3.4.3.3 Send Route Reply module

Send Route Reply module will be activated on the arrival of RREQ packet if the node is the destination node. RREP packet is then created with the source address from the IP datagram of the RREQ packet. The protocol then extracts the Node State information from the Node State Table of this node and inserts it into the RREP packet. The value of node bandwidth, medium access delay, *nci* and end-to-end delay is piggybacked onto the RREP packet. The IP datagram is then created and subsequently encapsulates the RREP packet. Following this, RREQ is copied into the Forward Route Request Table. The protocol allowed the destination node to permit the RREQ duplicates. This

allows the non-disjoint routes to be returned. Since the node is the destination node, RREP is then unicast to the next-hop neighbour following the reversed routes obtained from the RREQ packets. The flowchart for the Send RREP packet operation is in Figure 3-18.

3.4.3.4 Received Route Reply Module

If the node is the intermediate node, then the QoS parameters are obtained from the node's Node State Table and inserted into the RREP packet. The QoS parameters from the previous hop are extracted from the RREP packet into this node's QoS parameter's list. The route traversed by RREP packet is then copied into the node's Routing Table. After setting the RREP packet into the IP datagram, it is then forwarded to the next hop.

If the node is the source node, the QoS parameters list are extracted from the RREP packet and inserted into the node's QoS parameter's list. The QoS parameters matrix are generated, given as Eqn. (3-1), Eqn. (3-2), Eqn. (3-3) and Eqn.(3-4) ,

$$D = \begin{bmatrix} D_{0,0} & \cdots & D_{o,k-1} \\ \vdots & \ddots & \vdots \\ D_{k-1,0} & \cdots & D_{k-1,k-1} \end{bmatrix} \qquad \text{(Eqn. 3-1)}$$

$$C = \begin{bmatrix} nci_{0,0} & \cdots & nci_{o,k-1} \\ \vdots & \ddots & \vdots \\ nci_{k-1,0} & \cdots & nci_{k-1,k-1} \end{bmatrix} \qquad \text{(Eqn. 3-2)}$$

$$BW = \begin{bmatrix} B_0 & B_1 & B_2 & \cdots & B_{k-1} \end{bmatrix} \qquad \text{(Eqn. 3-3)}$$

$$d = \begin{bmatrix} d_0 & d_1 & d_2 & \cdots & d_{k-1} \end{bmatrix} \qquad \text{(Eqn. 3-4)}$$

where $D_{i,j}$ is the end-to-end delay, $nci_{i,j}$ is the node connectivity index, B_i is the node bandwidth and d_i is the medium access delay. The matrices representing the end-to-end delay are node connectivity index, node bandwidth and medium access delay respectively. Finally the RREP packet is destroyed. Figure 3-19 shows the flowchart for the Received Route Reply module.

3.4.3.5 Received Data Module

If there are still more hops in the data packet, the current node is not yet the destination of the data packet. After running some data processing maintenances, unicast the data packet to the next hop according to the data packet route list. On the other hand, if it is the destination node, then the packet will be sent to the upper layer.

3.4.3.6 Received Error Module

If the packet is of type RERR, then all routes in the node Routing Table that have a link from the error source address to the unreachable node address were removed. It avoids using stale information regarding the current Node State.

3.4.3.7 Received Connectivity Packet Module

The partially periodic transmission of CONN packet is explained in Section 3.3.1.1. The format of the packet is of the form similar to RREQ packet but with the identification set to zero. On receiving the packet type RREQ, the protocol checks the identification field. If it is '0 ', then the packet is CONN packet, otherwise it is a RREQ packet. If it is a CONN packet, it is then destroyed. The purpose of a CONN packet is to ensure the monitoring protocol can always measure the topology changes during the non-transmitting phases and maintain the Node State.

```
         ┌─────────────────────────┐
         │   ROUTE REQUEST         │
         │        SEND             │
         └─────────────────────────┘

              ╭──────────────╮
              │    Begin     │
              ╰──────────────╯
                     │
                     ▼
         ┌─────────────────────────┐
         │  Create the Route Request│
         │ Option for the Qosrga packet│
         └─────────────────────────┘
                     │
                     ▼
         ┌─────────────────────────┐
         │  Create the Qosrga packet │
         └─────────────────────────┘
                     │
                     ▼
         ┌─────────────────────────┐
         │  Create IP datagram, ver 4.│
         └─────────────────────────┘
                     │
                     ▼
         ┌─────────────────────────┐
         │ Increment Route Request Id, by 1.│
         └─────────────────────────┘
                     │
                     ▼
         ┌─────────────────────────┐
         │  Set the TTL field to 1 if │
         │ propagating request is set.│
         └─────────────────────────┘
                     │
                     ▼
         ┌─────────────────────────┐
         │ Set the originating Route Request Table│
         └─────────────────────────┘
                     │
                     ▼
         ┌─────────────────────────┐
         │ Send the IP packet to the CPU which│
         │      will broadcast it.   │
         └─────────────────────────┘
                     │
                     ▼
         ┌─────────────────────────┐
         │ Set the last broadcast time to│
         │       current time.       │
         └─────────────────────────┘
                     │
                     ▼
              ╭──────────────╮
              │    End.      │
              ╰──────────────╯
```

Figure 3-16 Route Request Send Operation

Process the rcvd RREQ pkt

Begin

Access the information from incoming IP packet

Insert this node into the list of hops traversed by the RREQ.

Update the Nodes Routing Table, which store the routes that had been traversed by this RREQ.

Access Info from the IP datagram.

Get the route request option

Is target addr in RREQ pkt match this node addr — YES

NO

Is source addr in RREQ pkt match this node addr — YES → SEND ROUTE REPLY PKT

NO

Get the number of hops from route list of RREQ packet

Test each node IP addr if it exist in the RREQ pkt ? — YES

NO

Check the entry of the originator node addr in the request Forwarding Table. Allow a number of duplicated entry, to produce non-disjoint multiple path.

Is the number of the duplicate entries are more than allowed ? — YES

NO

If TTL set to 1? — YES

NO

Destroy the packet

Add the entry for the Route Request Table

No route exist to the destination specified from this node. Rebroadcast this IP packet, with the option set to RREQ. At the same time, time stamp the packet.

End.

Figure 3-17 Received Route Request

SEND ROUTE REPLY PKT

Begin

Access the IP datagram. Get the RREQ option.

Access the RREQ option. Get the node address.

Remove this node address from the route list

Get the following Node State from this node and allocate the memory: bandwidth, medium_access_delay, nci, end_to_end delay

Create the RREP Option with the following option added representing qos parameters: bandwidth, medium_access_delay, nci, end_to_end delay, id.

Create the Qosrga pkt with the RREP option added.

Create the IP datagram and encapsulate the Qosrga packet with the destination, set as source address (originator of RREQ packet).

Insert the RREQ information into Forward Table. In this routine, the duplicated ID is allowed, in order that the multiple non-disjoint routes are returned.

Send the RREP packet.

END

Figure 3-18 Send Route Reply Option

60

```
┌─────────────────────────────┐
│  Process the rcvd RREP pkt   │
└─────────────────────────────┘

              ( Begin )
                 │
          ╱──────────────╲         YES
         ⟨  Is this        ⟩────────────────┐
          ⟨ node the destination of ⟩       │
          ╲ RREP pkt ?    ╱                  │
           ╲────────────╱                    ▼
                 │ NO              ┌─────────────────────────────┐
                 ▼                 │ Extract the QOS parameters from │
   ┌──────────────────────────┐   │ the RREP packet and save into   │
   │ Get the QOS parameters from │ │ the QOS parameters list.        │
   │ this Node State Cache. Insert │ └─────────────────────────────┘
   │ the QOS parameters into the   │               │
   │ RREP options packet           │               ▼
   └──────────────────────────┘   ┌─────────────────────────────┐
                 │                 │  Generate the QOS            │
                 ▼                 │  Parameters Matrix           │
   ┌──────────────────────────┐   └─────────────────────────────┘
   │ Extract the QOS parameters │                 │
   │ from the RREP packet and   │                 │
   │ save into the QOS          │                 │
   │ parameters list.           │                 │
   └──────────────────────────┘                  │
                 │◄──────────────────────────────┘
                 ▼
   ┌──────────────────────────┐
   │ Set the Qosrga packet into │
   │ the IP datagram.           │
   └──────────────────────────┘
                 │
                 ▼
   ┌──────────────────────────┐
   │ Insert the route traversed │
   │ by the RREP packet into    │
   │ the Routing Table          │
   └──────────────────────────┘
                 │
                 ▼
   ┌──────────────────────────┐
   │ Acess the packet           │
   │ information from the IP    │
   │ packet                     │
   └──────────────────────────┘
                 │
                 ▼
          ╱──────────────╲        YES
         ⟨ Is this node   ⟩──────────────────┐
          ⟨ is the destination of ⟩          │
          ⟨ the Route Reply pkt ? ⟩          │
           ╲────────────╱                    │
                 │ NO                         │
                 ▼                            │
   ┌──────────────────────────┐              │
   │ The node is not the        │             │
   │ destination of the Route   │             │
   │ Reply pkt. Determine the   │             │
   │ next hop to which this      │             │
   │ packet is to be sent.       │             │
   └──────────────────────────┘              │
                 │                            │
                 ▼                            │
          ╱──────────────╲        NO          │
         ⟨ Is the address ⟩──────────────────┤
          ⟨ belongs to this node ? ⟩         │
           ╲────────────╱                    │
                 │ YES                        │
                 ▼                            │
          ╱──────────────╲        NO          │
         ⟨ Is the next hop ⟩─────────────────┤
          ⟨ the destination node? ⟩          │
           ╲────────────╱                    │
                 │ YES                        │
                 ▼                            │
          ╱──────────────╲        NO          │
         ⟨ Is the address ⟩──────────────────┤
          ⟨ belongs to this hop at all? ⟩    │
           ╲────────────╱                    │
                 │ YES                        ▼
                 ▼                  ┌──────────────────┐
   ┌──────────────────────────┐    │ Destroy the packet.│
   │ Forward the packet to the  │   └──────────────────┘
   │ next hop address.          │            │
   └──────────────────────────┘            │
                 │◄────────────────────────┘
                 ▼
              ( End )
```

Figure 3-19 Received Route Reply

3.5 Performance Evaluations of NDMRD Protocol

3.5.1 The Number of Routes Discovered

The strength of the NDMRD protocol is the number of routes that it discovers as a result of RREQ packets sent and RREP packets received. The total number of routes that were discovered depends largely on two related variables, the Route Accumulation Latency (RAL) and the number of RREQ duplicates. The number of routes available and the Node State information that are piggy-backed to RREP packet will enhance the ability to discover the routes based on the given metrics. The multiple routes discovery is done within the Route Accumulation Latency specified. The total number of routes is obtained by counting the total number of RREPs from the destination. A simulation experiment is done to ascertain the value of RREQ duplicates and Route Accumulation Latency that need to be specified for our QoS routing protocol. The setup consists of 20 nodes with the Random Waypoint Mobility model, representing the nodes that move randomly inside a field configuration of 1000m x 1000m. The objectives of the simulation experiments are three fold: (1) to study the effect of Route Accumulation Latency on the number of routes discovered; (2) to study the effect of RREQ duplicates on the number of routes discovered; and (3) to study the relationship between the numbers of routes discovered and the overall performance of the NDMRD protocol for one and five CBR sources when considering various numbers of routes. The overall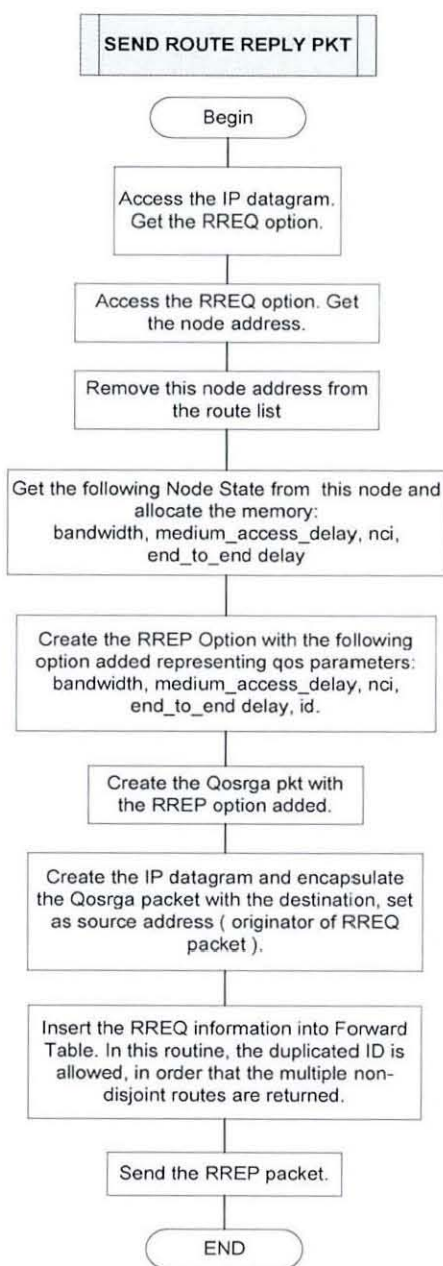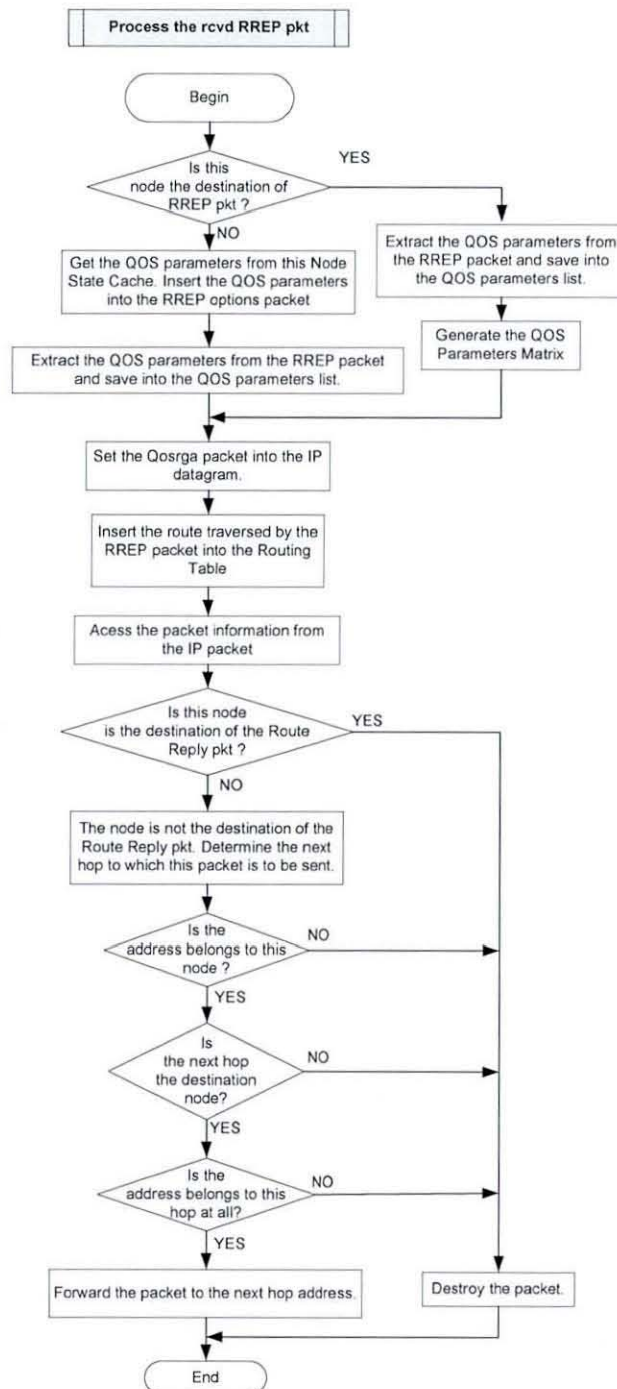 performance of the NDMRD protocol will indicate the optimum value of Route Accumulation Latency and the number of RREQ duplicates that need to be set.

3.5.2 The Effect of RREQ Duplicates

Figure 3-20 shows the number of routes accumulated as a function of the duplicates. When number of duplicates is set to zero, an average of 2 routes is obtained. Maximum number of routes is accumulated when duplicates is set to 10 until 15. After 15 duplicates the number of routes starts to drop to 15 routes and when 20 duplicates are used, the routes discovered are reduced to 16. Initially, no congestion occurs, so many routes are accumulated. As the RREQ duplicates are increased, the broadcast and the unicast packet that are

generated greatly increased the traffic congestion. Hence the number of initial routes discovered is less.

3.5.3 The Effect of Route Accumulation Latency (RAL)

Figure 3-21 shows the effect of changing the Route Accumulation Latency on the number of initial routes. The rate of change of the number of routes from RAL=0.05 until RAL=0.1 is very steep. The number of initial routes accumulated is increasing proportionately. Beyond that, the rate of change of the number of routes decreased tremendously. When RAL is set more than 0.1 seconds, the duplication of RREQ packet becomes more of a nuisance than a necessity. It generated more congestion and caused RREP packet to drop more often. For RAL beyond 0.1, the rate of change of number of routes is very small. It may be concluded that the value of RAL to be set in all future experiments should be between 0.08 to 0.1 seconds.

3.5.4 The Overall Performance of NDMRD Protocol

In this section the performance of NDMRD protocol with Average Packet Delivery Ratio (APDR) as metrics is presented. APDR is defined as ratio between the average total traffic received and average total traffic sent. The source data rate used is 40 kbps. Two sets of experiments were done, one with a single CBR source and the second with five CBR sources. Referring to Figure 3-22 and Figure 3-23, APDR is drawn against the simulation time for single CBR source and five CBR sources. Generally, APDR for single CBR source is better than APDR for five CBR sources. The difference in performance is due to congestion of a five-sources scenario as being more prominent than a single source scenario. The performance of NDMRD protocol also depends on the number of initial routes accumulated. The results are redrawn to show the relationship between APDR and the number of routes as shown in Figure 3-24. It may be observed that the smaller number of routes produce slightly better APDR compared to the higher number of routes. It can be attributed to the fact that the larger number of routes takes more time to generate a QoS route compared to the lesser one. However, the percentage difference is approximately 10 %.
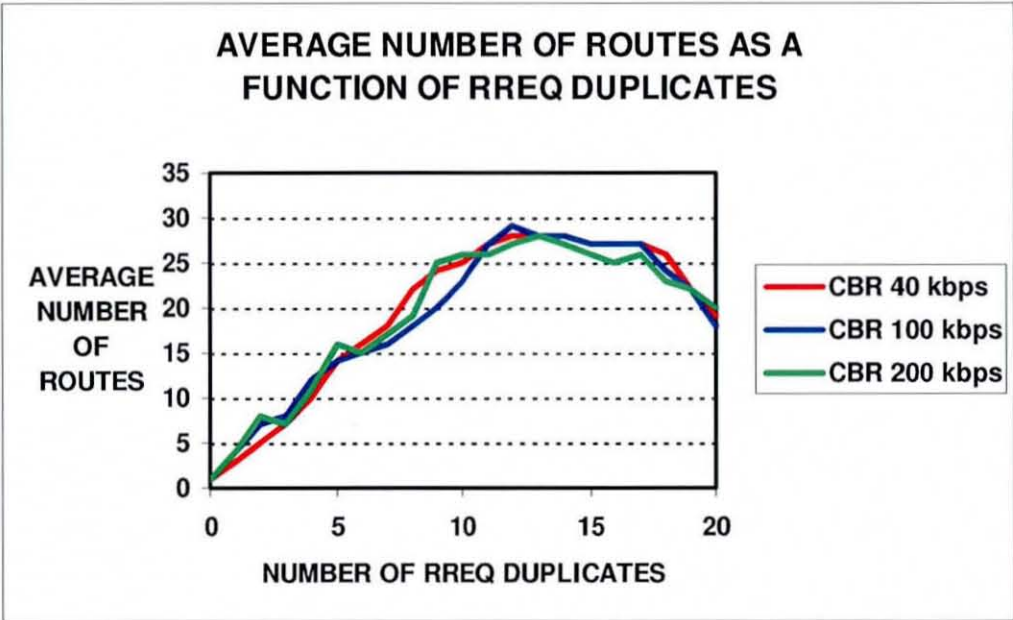
Figure 3-20 Effect of RREQ Duplicates on the Number of Routes
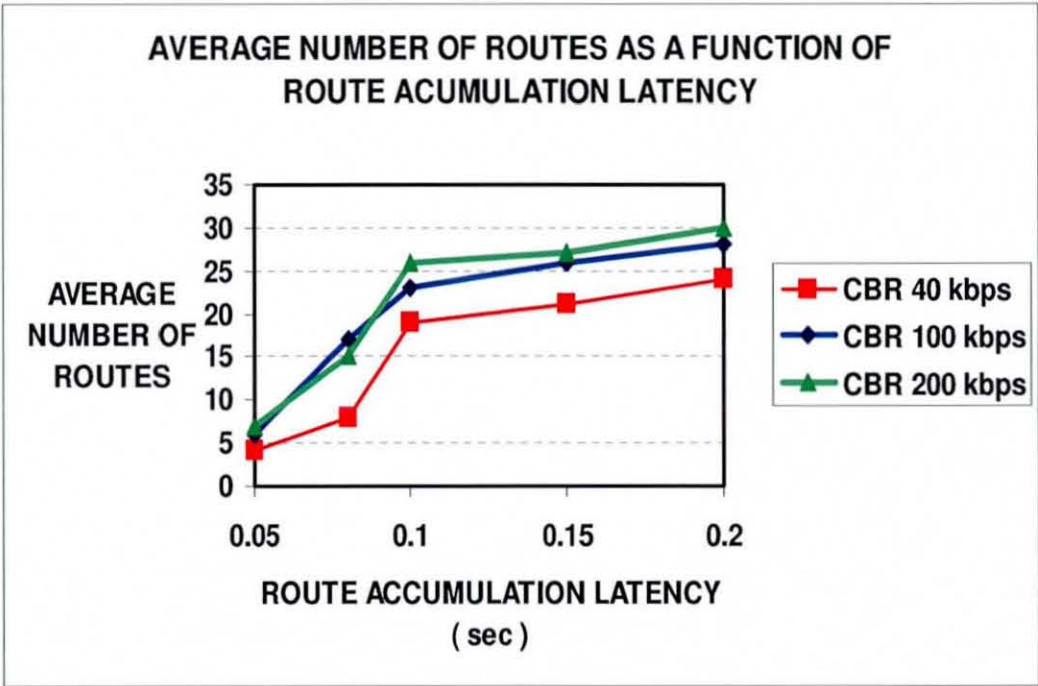
Figure 3-21 The Effect of Route Accumulation Latency
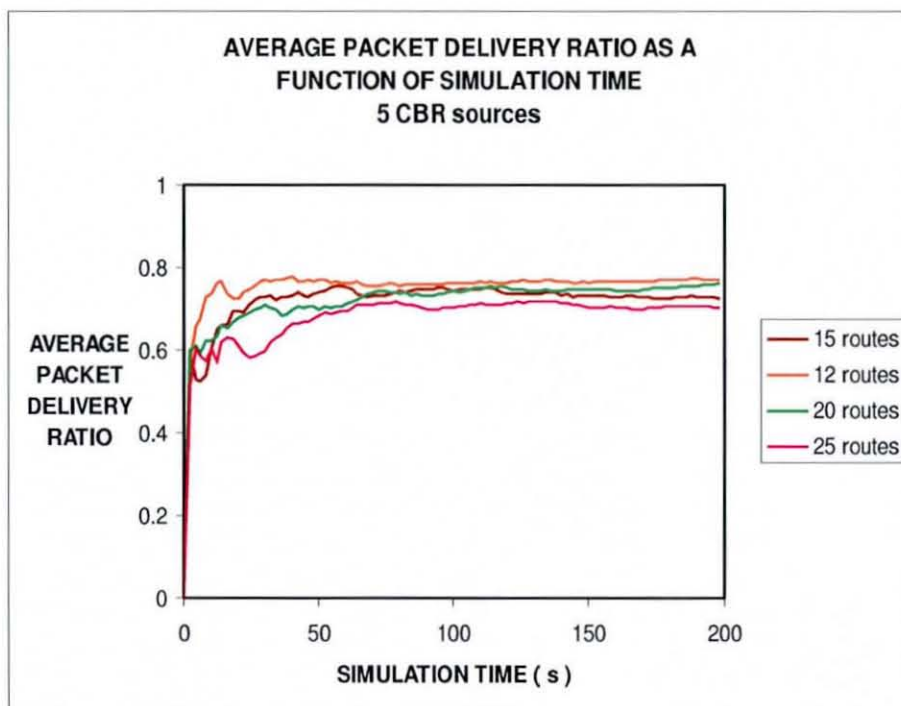
Figure 3-22 Performance of NDMRD Protocol Based on APDR metric for 5 CBR Sources
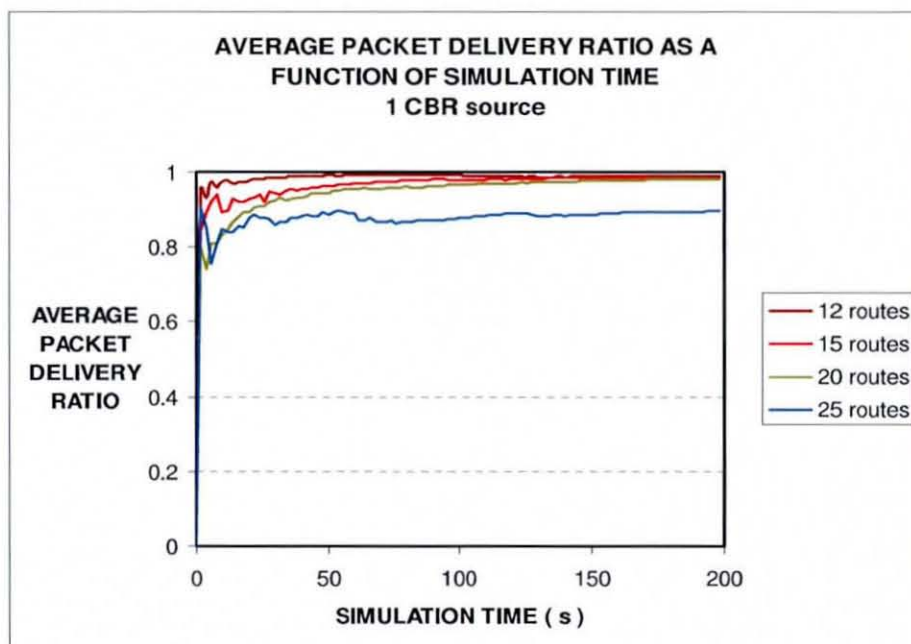
Figure 3-23 Performance of NDMRD Protocol Based on APDR metric for 1 CBR Source

65

This observation is very significant, such that our setting of RAL and RREQ duplicates should reflects the possible 10 % gain in term of APDR. But since the simulation experiments are based on only 40 kbps, 100 kbps and 200 kbps, then as a guideline the value of RREQ duplicates should be set between 5 and 10 duplicates and the value of RAL between 0.08 and 0.1 second.
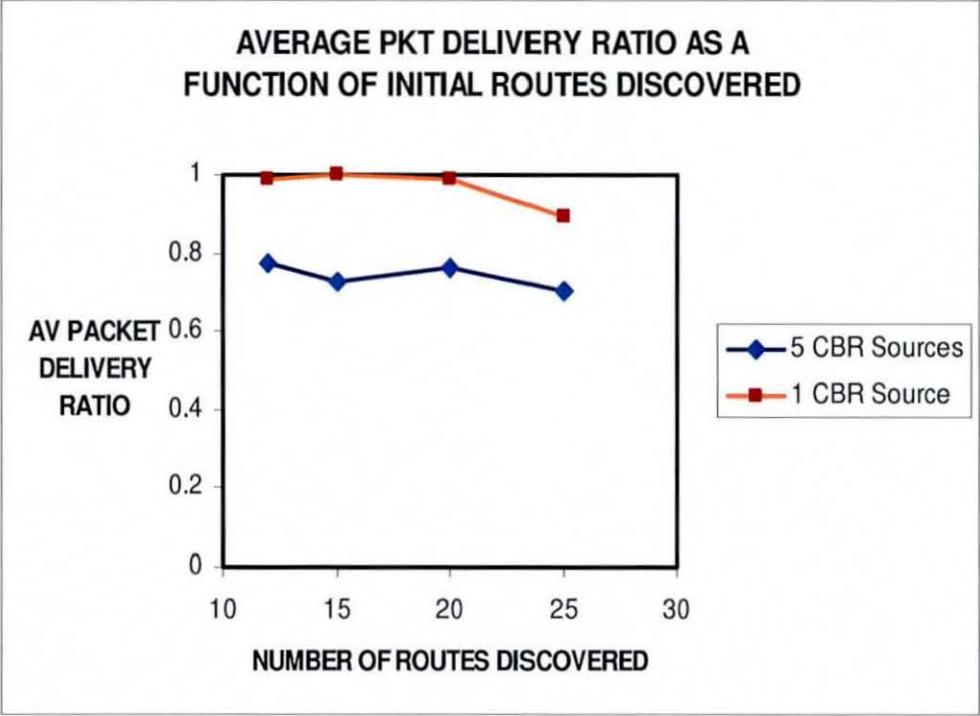
Figure 3-24 Performance of NDMRD Protocol Based on APDR Metrics Against the Number of Routes for One and Five CBR Sources

3.6 SUMMARY

The overall implementation of QoS routing algorithm is presented. The QoS routing protocol, "QoS Routing for MANETs Using a Genetic Algorithm" with its acronym as QOSRGA was defined. The message format and the data structures were formally defined. Then the Non-Disjoint Multiple Routes Discovery (NDMRD) protocol was introduced, which makes up the QOSRGA. The characteristics of the NDMRD are QoS-Aware, since the contents of its route reply packet are QoS parameters. The implementation of the protocol using the

Opnet Modeler, was formally described. The overall flowchart is described. The detailed explanations were given for; (1) the upper layer packet arrival routine, (2) route request sent routine, (3) the received route request routine, (4) send route reply routine, (5) received route reply routine and (6) received data packet routine. Finally a series of performance tests were run to actually see the benefit of multiple routes and routes accumulation. It was shown how to generate non-disjoint multiple routes by using RREQ duplicates and how to limit the initial number of routes by introducing the variable Route Accumulation Latency. The performance test of NDMRD protocol gives an insight into the value of RAL and the number of duplicates.

CHAPTER 4
THE NODE CONNECTIVITY METRIC

4.1   Introduction

The vision of MANET is characterised by mobile, lightweight and personalised computing devices, being collaborative in nature towards a self-establishing network. When network topology changes due to the mobility of nodes, this will result in three node pair scenarios, least-connected, highly-connected or breaking-up. The changes in the set of links of a node affect not only the node's ongoing communication, but may impede the communication of other nodes as well. As the capacity and communication ability of MANETs are dependent on node connectivity, it is important to understand how node connectivity behaves in MANETs. Routing protocols may cause the route to change in response to the variation of quality of node connectivity. The resultant route changes may alter the traffic distribution in the network, causing congestion at a number of nodes with less traffic at other nodes. The network dynamics require a mobility metric which can quantify the node movements and eventual relationship with the neighbours. The final outcome would be how reliable inter-nodal communication is. The stable and durable routes are selected based on the connectivity criteria from a node to its neighbours and successively in pairs towards the target nodes. Hence, QoS routing in this case

is the mechanism to find a feasible route which has overall good connectivity criteria from amongst the routes discovered.

For effective performance of QoS routing protocols, it requires a connectivity metric that can accurately capture the networking dynamics of the node. Effectively it indicates the resulting performance of the QoS routing protocol. Such a metric could provide feedback information which shows the quality of node connectivity. The connectivity metric must be applicable to a real mobile network with real nodes. The first step is to define a set of requirements that must be met to enable effective function of QoS routing protocols. The metrics must be (1) able to compute in a distributed environment, (2) able to adapt a locally measured performance, (3) a numerical quantity which indicates the quality of node-pair connectivity, and (4) capable of processing in a real time implementation. Ultimately, the metric should describe quantitatively the dynamic of the nodes such that it can be of use in the route discovery and selection process to choose the most reliable, longer duration and stable routes. It can also be used to select the cost-effectiveness of the longer duration and stable routes. Lastly, it is expected to generate the available QOS routes.

The rest of the chapter is organised as follows. Section 4.2 outlines the related work on node connectivity. Section 4.3 describes in detail the Node Connectivity Index, where the node mobility is characterised by the expansion and contraction model. Section 4.4 introduces and describes in detail the Node Connectivity Index including performance evaluations. The chapter ends with a summary in Section 4.5.

4.2 Related Work

Route reliability issues have been addressed in several works. In the Associativity-based Routing [32] , the association stability of the links is accounted when choosing routing paths. It attempts a measure of goodness using associativity ticks which indicates how stable a route is. The Signal Strength Adaptive protocol [33] further considers the signal strength in choosing the good paths. With the Route-Lifetime Assessment Based Routing protocol [34] , a parameter called affinity is defined to characterise the signal

strengths and stability of a route. An AODV [10] tries to pre-empt a link failure by using periodic hello packets and propagating failure messages to source using the route. Pre-emptive routing [35] is an enhancement to DSR that enables nodes to estimate the time to link failure and propagate route failure information in advance. All these mechanisms suffer from the fact that they are local to the node and not necessarily good strategy for the whole route. An attempt to view the whole route was carried out [36] using some kind of mobility prediction. This requires a global clock synchronisation and a positioning system for the network. These mechanisms do not capture the aspect of the relative velocity for the route.

4.3   Node Connectivity Metric

4.3.1 Characteristics of Node Movements

The main challenge of QoS routing in mobile ad hoc networks is to handle the topology changes appropriately. The performance of a protocol is greatly determined by its ability to adapt to these changes. Hence, it would be useful to have metrics that characterise the effect of mobility on the connectivity of node pair. Mobility characteristics that affect the connectivity metrics need to be defined. These are the link lifetime, route lifetime, the number of link changes, node degree and route availability. Link lifetime is the duration of connectivity for a single hop node pair. The route lifetime is the duration of full connectivity from source to destination for a certain number of hops. Node degree is the number of neighbouring nodes within the transmission range of source node. Link availability is the fraction of time where there exists a connection between two neighbouring nodes. It would be necessary to consider a metric in terms of a single positive value as node-pair connectivity index, to be incorporated into the route selection algorithm. The metrics are all culminated from the effect of mobility with certain velocities. The velocities of nodes within MANET have the characteristics of spatial and temporal dependence. Spatial dependence is the extent of similarity of the velocities of two nodes that are within the transmission range. Temporal dependence is the extent of similarity of the velocities of a node at two time instances that are not too far apart. Relative

velocity is the velocity difference between two nodes considered. Node pairs moving closer, and node pairs moving apart, experience higher relative velocity. Node pairs moving along side each other or moving in the same direction experience low relative velocity. The relative velocity is chosen to impact on our QoS routing algorithm as it is governed by the strength of connectivity between any set of node pairs within the route. The strength of connectivity between the node pair can be derived from the fact that node-pairs may be moving towards each other in a contraction model or they may be moving away from each other in an expansion model.

4.3.2 Contraction and Expansion Model of Node Movement

Ideally the contraction model emulates the movement of mobile nodes towards a logical centre from all directions. In practical scenarios, at $t = 0$, a node may enters another node transmission range at an angle and move in a direction not towards the other node. Contraction and expansion rates increase as velocity increases. This results in a node reaching the boundary of transmission range after a short time duration. Consequently, this affects the node connectivity time, route lifetime and the quality of node connectivity. Figure 4-1 shows the contraction model. If the movement pattern of the nodes is away from each other, than it is expansion model. Ideally, if at $t = 0$, a node is within a transmission range of the other node, then it moves away from the centre to the edges. In a practical scenario, the mobile nodes may move away from each other at an inclination angle. Incidentally, the starting point of the node is within the transmission radius, in contrast to the contraction model. Figure 4-2 shows the expansion mobility model.

4.3.3 Hybrid Model Characteristics

In this mobility model, the movements of a node may switch from a contraction to an expansion model. These switchable characteristics are due to the relative distance between the node-pair and the time a node is within transmission range of its neighbours. Initially, the node is in the contraction model, having entered the transmission range at an angle of arrival, until it

reaches a perpendicular point towards the other node where it enters the expansion model.
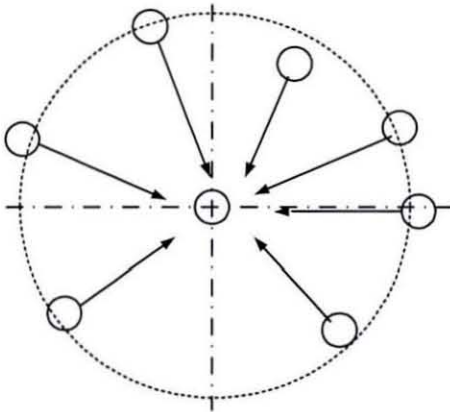
Figure 4-1 Contraction Model of Node Connectivity
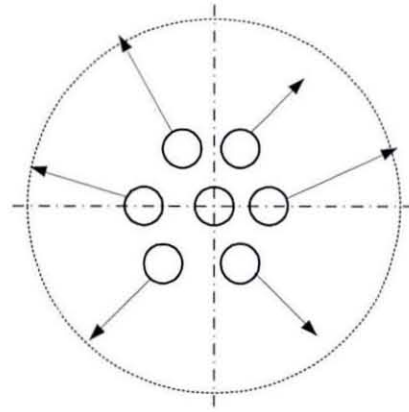
Figure 4-2 Expansion Model Of Node Connectivity

Finally, the node reaches the edge of the transmission range at a point of departure. Considering that the node moves in a straight line motion, at a point in time, the node will exhibit the expansion mobility characteristics. There exists a finite time between the node pair when they will always be in connectivity. The node connectivity time is an important QoS routing parameter for mobile nodes. The connectivity time depends on the angle of arrival and velocity. Figure 4-3 shows the hybrid model, illustrating that the node connectivity time is higher when it is at a particular position within the range for contraction, compared to if it were at a position within the range for expansion.

4.3.4 Analytical Consideration of Node Connectivity Time

4.3.4.1 Expected Node Connectivity Time

Consider two nodes i and j, at time t_1, when the duration of the node connectivity $L(i, j)$ is the time interval (t_1, t_2) during which the two nodes are within the transmission range of each other. However, these two nodes could not be within the transmission range at time $t_1 < t$ and also at time $t > t_2$. Formally, $L(i, j) = (t_1, t_2)$ if and only if, for all values of t, $t_1 \leq t \leq t_2$.

An analytical model is presented to determine the mean connectivity time of a node-pair. The mean connectivity time of a node pair shows the estimated length of time a node pair is in contact with respect to the relative velocity of the nodes. Hence the lower and upper bound of node pair connectivity time which is useful in ensuring a complete communication session can be obtained.
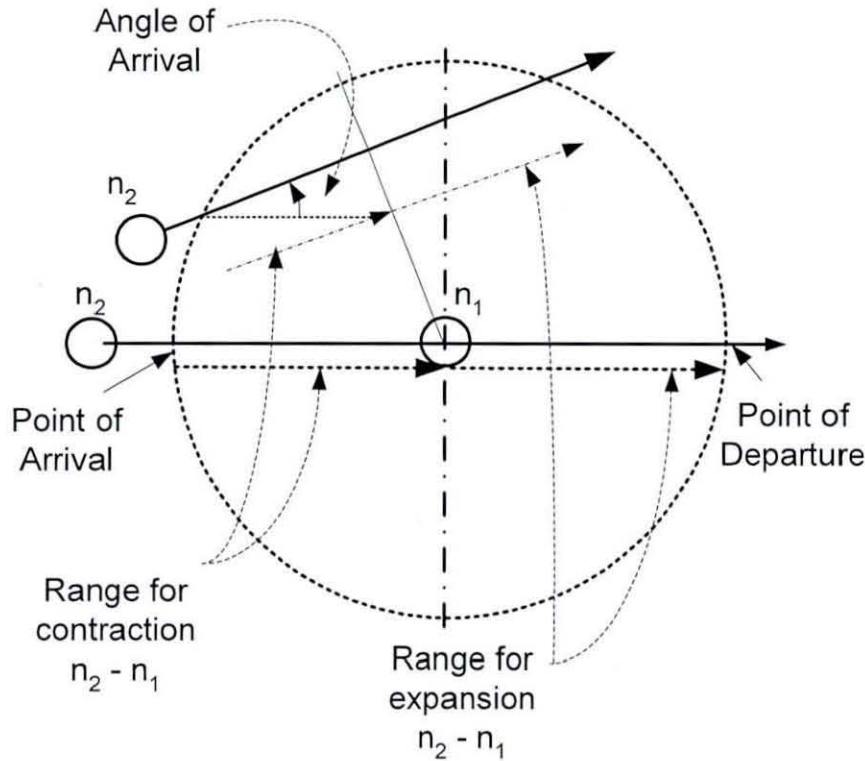
Figure 4-3 Hybrid of Contraction and Expansion Model

The analytical estimate of the node pair connectivity time is obtained using the approach taken by Hong[37] , in the cellular network where the authors derived the expected time of a node residing within a cell. In this case, the connectivity time of a node pair, is estimated, which is the time that a node is always in contact wirelessly with the other node at a velocity and angle of arrival assumed to be uniformly distributed. The transmission range is set to 250m. Next, it is assumed that during the connectivity time, the mobile node moves along the specific direction, with uniform velocity. The velocity distribution of different nodes at different time intervals conforms to a velocity

profile, with uniform distribution. Figure 4-4 shows a mobile node, n_2 moving in the vicinity of another node, n_1 which is assumed to be stationary. In reality, both nodes may be mobile. The mobile node arrives at the boundary of the circular range of the stationary node, at time of arrival, **TOA,** and it keeps on moving until the boundary is reached again, at departure time, **TOD.** The velocity of the mobile node is assumed to be uniformly distributed between **0** and V_{max} m/s. The direction of arrival of node n_2 is given by θ, the value of which is between **-π/2** and **+π/2.** The mobility of the node is then characterised by the velocity PDF, $f_v(v)$ and directional PDF $f_\theta(\theta)$. In order to determine analytically the time range where n_2 resides within the range of n_1, the two PDF are defined. Velocity PDF is given as,

$$f_v(V) = \begin{cases} \dfrac{1}{V_{max}} & for \quad 0 \le V \le V_{max} \\ 0 & otherwise. \end{cases}$$
(Eqn. 4-1)

When the mobile node arrives at time **TOA** and may transverse a distance **X** in any direction with equal probability, the random variable θ, has PDF as,

$$f_\theta(\theta) = \begin{cases} \dfrac{1}{\pi} & for -\dfrac{\pi}{2} \le \theta \le \dfrac{\pi}{2} \\ 0 & elsewhere \end{cases}$$
(Eqn. 4-2)

Consider node n_2 from Figure 5.4, which traverses a distance X, thus,

$$X = 2R \cdot \cos \theta$$
(Eqn. 4-3)

For node n_2 traversing a distance X with time (where $0 \le X \le 2R$) then the density function of X,

$$F_X(x) = \Pr(X \le x).$$
(Eqn. 4-4)

Resolving, for $F_x(X)$,

$$F_X(x) = \begin{cases} 0, & \text{for } x < 0, \\ 1 - \dfrac{2}{\pi} \cdot \cos^{-1}\left(\dfrac{x}{2R}\right), & \text{for } 0 \le x \le 2R, \\ 1, & \text{for } x > 2R. \end{cases} \qquad \text{(Eqn. 4-5)}$$

Then the PDF of X,

$$f_X(x) = \frac{d}{dx} \cdot F_X(x) \qquad \text{(Eqn. 4-6)}$$

reduced to the form,

$$f_X(x) = \begin{cases} \dfrac{1}{\pi} \cdot \dfrac{1}{\sqrt{R^2 - \left(\dfrac{x}{2}\right)^2}} & for\, 0 \le x \le 2R \\ 0, & elsewhere \end{cases} \qquad \text{(Eqn. 4-7)}$$
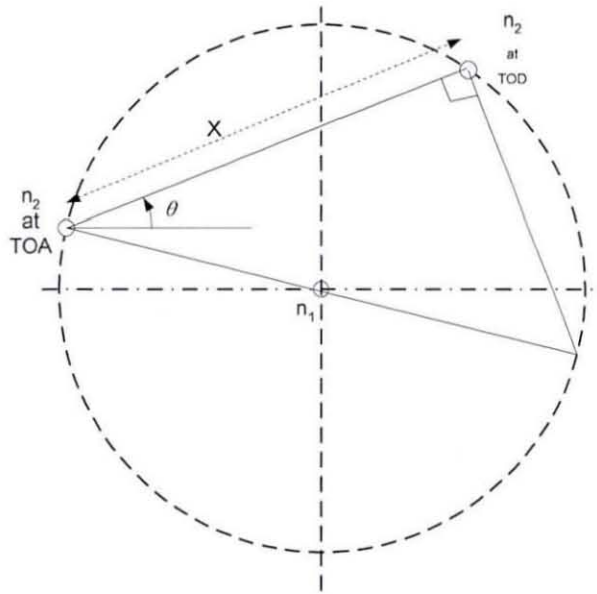
Figure 4-4 Estimating the Expected Node Connectivity Time

Node pair connectivity time is defined as the mean time; a node is within the transmission range of the other node and both are fully connected. The node connectivity time for the mobile node to travel from the point of arrival to the point of departure, a distance of X, with the velocity V, generally is given as,

$$T_{NCT} = \frac{X}{V}$$

(Eqn. 4-8)

The PDF of the node connectivity time can be found using a standard method.

$$f_{NCT}(t) = \begin{cases} \dfrac{4R}{\pi V_m} \cdot \dfrac{1}{t^2} \left[1 - \sqrt{1 - \left(\dfrac{V_m t}{2R} \right)^2} \right] & for \quad 0 \leq t \leq \dfrac{2R}{V_m}; \\[4mm] \dfrac{4R}{\pi V_m} \cdot \dfrac{1}{t^2} & for \qquad t \geq \dfrac{2R}{V_m}; \end{cases}$$

(Eqn. 4-9)

The expected mean node connectivity time for the 2-node is given as,

$$\overline{T_{NCT}} = \int_0^\infty t \cdot f_{NCT}(t)\, dt$$

(Eqn. 4-10)

By solving Eqn. 4-10, the mean node connectivity time, T_{NCT} is now a function of velocity. Figure 4-5 shows the plotting of T_{NCT} against velocity. From the plot the time, T_{NCT} for any given velocity can be explicitly estimated . The lower bound and the expected upper bound of the node connectivity can also be determined.

From the graph, it can be infered that, node connectivity time depends on the velocity of the node. The estimate of T_{NCT} for any velocity, can then be obtained. There is also a limit to the velocity of each node in the system. For this QoS routing simulation, the appropriate velocity is chosen.

4.3.4.2 Node Connectivity Time Obtained Through Opnet Simulation

Figure 4-6 shows the plot of T_{NCT} against velocity. Each reading was taken for a specific angle of arrival. The time difference between the Time of Arrival (**TOA**) and Time of Departure (**TOD**) is taken as the node connectivity time in seconds. Different angles of arrival give a slightly different curve. Low velocity gives higher node connectivity time. The result from Opnet simulation is similar to the one obtained analytically, with a slightly higher connectivity time for a given velocity. Hence we it can be concluded that Opnet simulation can be used to simulate the node mobility model and extract a suitable node

76

connectivity time. The model can be used to estimate node connectivity time for a given velocity and the maximum time for the purpose of QoS routing implementation. Velocity of 25 m/s and 1 m/s resulted in the connectivity duration of about 10 seconds and 225 seconds respectively. Hence the velocity which gives a reliable connectivity duration would be between 0.5 m/s (3.6 km/h) and 25 m/s (90 km/h). The velocity of 0.5 m/s depicts a walking scenario whereas the velocity of 25 m/s depicts a car travelling on the highway. A metric which can describe the relative difference in connectivity time for different node pairs is needed. By comparing the values from different node pairs, those which have surplus connectivity time left can be selected. In the next section a node connectivity index is proposed.

4.4 Development of Node Connectivity Index (nci)

4.4.1 Estimation of the Normalised Relative Velocity

It is not enough to view the network dynamics as having nodes connected or unconnected, but rather how long the connection will last. The longer the connection the better it contributes to the reliability and stability of the whole route. It is proposed to derive a metric which describes the dynamicity of the mobile nodes. In order to compute this metric, the connectivity between two adjacent nodes must be measured. Then the connectivity strength of every node pair is computed to produce a positive numerical index. The index can proportionately describe how reliable a route is. Those low speed nodes which have longer duration and are more reliable are represented by a low value index. Stationary nodes produce a zero index. The rate of separation, expanding and contracting measures, between each adjacent pairs of nodes can be estimated by computing the relative velocity.

Consider node n_1 receiving packets from node n_2 in Figure 4-7. Within the transmission range, node n_2 will always be on the circular disk centred at node n_1. Node n_2 is said to be within the virtual concentric region. The virtual
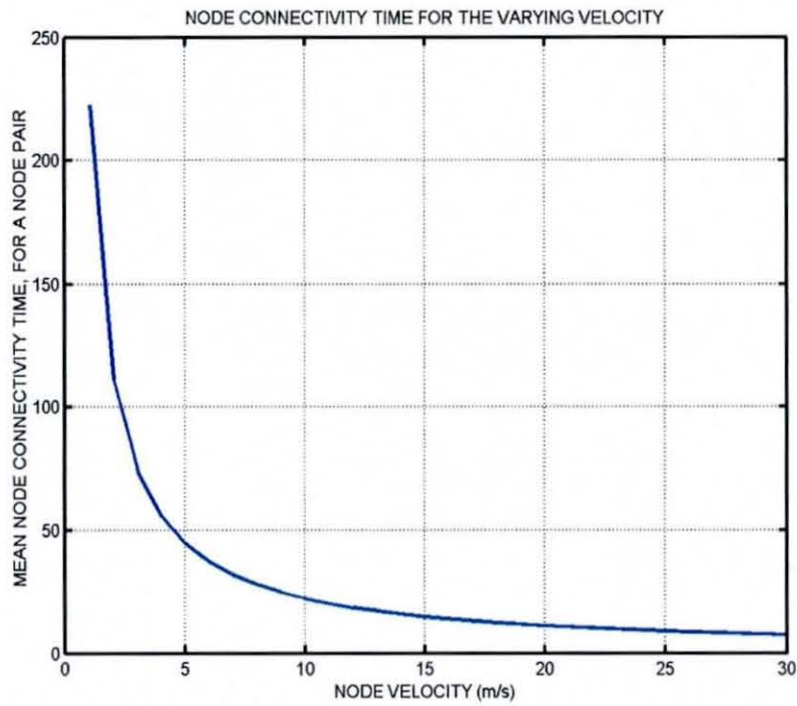
Figure 4-5 Plot of T_{NCT} against velocity from analytical model
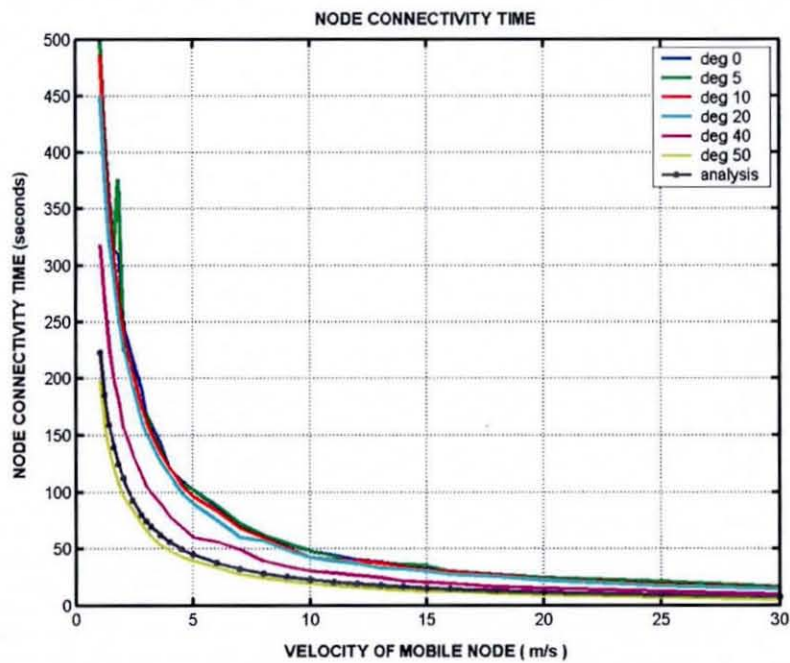
Figure 4-6 Node Connectivity Time against Velocity

78

concentric region is a circular region within the node's (i.e n_1) transmission range, where at any instant its adjacent node (i.e n_2) will always be before reaching point of departure. Let t_1 and t_2 be the times at which the last two packets from n_2 were received. The received power from node n_2 when packets have arrived is denoted as P_1 (at time t_1) and P_2 (at time t_2). From this there exist two possibilities: that nodes are moving closer when $P_2 > P_1$ and nodes moving further apart when $P_2 < P_1$.

Assumptions:

1. The free space path loss propagation model is used [38] hence power received by the receiver antenna due to transmission from the transmitter antenna is given as:

$$P_r = P_t G_t G_r \lambda^2 / (4\pi)^2 d^2 L \qquad\qquad \text{(Eqn. 4-11)}$$

where, P_r is the received power; P_t is the transmitted power; G_t is the transmitter antenna Gain; G_r is the receiver antenna gain; d is the distance between transmitter and receiver; L is the system loss factor and λ is the wavelength in metres.

2. Assuming that all the parameters above, except distance, are the same throughout all nodes, then the received power is inversely proportional to the square of distance, d. Then power, $P_r = k/d^2$, where

$$k = P_t G_t G_r \lambda^2 / (4\pi)^2 L .$$

3 In all cases, the node is assumed to move in a linear motion and posses unlimited battery power.

4 The maximum transmission range is assumed to be 250 m.
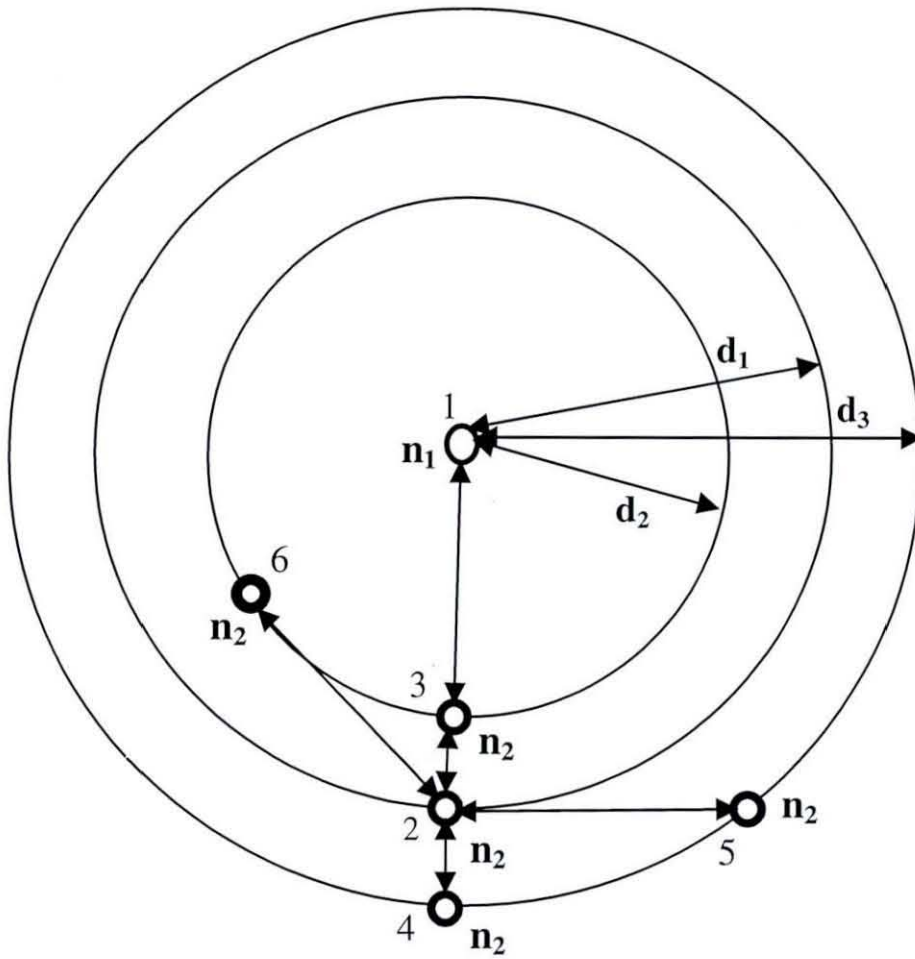
Figure 4-7 The Graphical Illustration of *nci* Calculation

4.4.2 Contraction and Expansion Mobility Model

4.4.2.1 Expansion Model

Referring to the Figure 4-7, consider nodes moving apart at a constant velocity and in linear motion. Two extreme scenarios may occur, that is, n_2 moves from position 2 to position 4 or from position 2 to position 5. Consider, n_2 moves from 2 to 4; distance = d_3 - d_1, then,

$$(d_3 - d_1)/\sqrt{k} = (1/\sqrt{P_2}) - (1/\sqrt{P_1})$$
(Eqn. 4-12)

$$v/\sqrt{k} = (1/(t_2 - t_1))((1/\sqrt{P_2}) - (1/\sqrt{P_1}))$$
(Eqn. 4-13)

Next, consider, n_2 moves from 2 to 5; distance = $\sqrt{(d_3^2 - d_1^2)}$ then,

80

$$\sqrt{(d_3^2 - d_1^2)} \Big/ \sqrt{k} = \sqrt{(1/P_2) - (1/P_1)} \qquad \text{(Eqn. 4-14)}$$

$$v \Big/ \sqrt{k} = (1/(t_2 - t_1)) \sqrt{(1/P_2) - (1/P_1)} \qquad \text{(Eqn. 4-15)}$$

Eqn. 4-15 gives a worst-case scenario, hence this equation is being considered in the calculation of *nci*.

4.4.2.2 Contraction Model

Figure 4.7 shows nodes moving closer with a constant speed in linear motion. Similarly, two scenarios can be used to find the relationship that will deduce the speed of the node. Consider, n_2 moves from 2 to 3: distance $= d_1 - d_2$, then,

$$(d_1 - d_2)/\sqrt{k} = (1/(\sqrt{P_1}) - 1/(\sqrt{P_2})) \qquad \text{(Eqn. 4-16)}$$

and hence,

$$v \Big/ \sqrt{k} = 1/(t_2 - t_1)((1/\sqrt{P_1}) - (1/\sqrt{P_2})) \qquad \text{(Eqn. 4-17)}$$

Next, consider n_2 moves from 2 to 6: distance $= \sqrt{d_1^2 - d_2^2}$

$$v \Big/ \sqrt{k} = 1/(t_2 - t_1)((1/\sqrt{P_1}) - (1/\sqrt{P_2})) \qquad \text{(Eqn. 4-18)}$$

The equation Eqn. 4.18 gives a worst-case scenario, hence it is also being considered in the calculation of *nci*. From these equations we have the estimates of the node-pair relative velocity normalised by the constant due to antenna and propagation factors. The relative velocity can be used to estimate the aggregate values for all node-pairs that are within the source-destination route.

4.4.3 Definition of *npem*, *npcm* and nci

Definition 4.1: Consider a route R which consists of a set of nodes, which are in motion with a velocity, **V** and with varying angle of arrival, then R= {n_0, n_1, n_2,... n_{n-1} }. For R to be a reliable route, each node must be connected to its adjacent node for the whole period of message transmission. Let E denote the

node pairs, (n_j, n_k) such that they are adjacent nodes. If the pairs are moving away from each other, the node pair expansion metric (npem) is given by,

$$npem = (1/(t_2 - t_1))\sqrt{((1/P_1) - (1/P_2))} \qquad \text{(Eqn. 4-19)}$$

Definition 4.2: Consider a route R which consists of a set of nodes, which are in motion with a velocity, v and with varying angle of arrival, R= {n_0, n_1, n_2,...n_{n-1}}. For R to be a reliable route, each node must be connected to its adjacent node for the whole period of message transmission. Let E denote the node pairs, (n_j, n_k) such that they are adjacent nodes. If the pairs are moving toward each other, the node pair contraction metric (npcm) is given by,

$$npcm = (1/(t_2 - t_1))((1/\sqrt{P_2}) - (1/\sqrt{P_1})) \qquad \text{(Eqn. 4-20)}$$

Observe that *npcm* and *npem* are positive quantities but in different directions, *npcm* towards high positive values and *npem* towards low positive values. These two values must be combined to form a single metric to indicate the quality of connectivity between the two adjacent mobile nodes. We note that the node with *npcm* lasts longer than that with *npem*. Hence we need a weighted form of these two values in a single index quantity. This value will indicate a strong connectivity if its value is low and weak connectivity if its value is high. A high value shows that the node is more dynamic and will soon reach the point of departure from the other node's transmission range.

Definition 4.3: Consider a set E which consists of the node pairs, (n_j, n_k) such that they are adjacent nodes. If the pairs are moving towards each other or away from each other, the node pair connectivity index, **nci** is a positive value which describes the quality of connectedness between any two adjacent nodes. The least **nci** value indicates a good quality connection, in which the node pair connectivity time is larger compared to high **nci** value, then **nci** is defined as,

$$nci = \begin{cases} 0.25 - \left[\dfrac{1.0 \times 10^5}{8.0 \times 10^5 - npem} \right], & \text{for } P_2 < P_1. \\[3ex] \dfrac{1.0 \times 10^5}{8.0 \times 10^5 + npcm}, & \text{for } P_2 > P_1. \\[3ex] 0, & \text{for } P_2 = P_1. \end{cases} \qquad \text{(Eqn. 4-21)}$$

The algorithm for calculation of *npem, npcm* and *nci* is part of the whole algorithm for monitoring of MANET. The algorithm for calculation of *npem, npcm* and *nci* is label as Algorithm 4.1.

4.4.4 Performance Evaluation of *nci*

Various simulation experiments are carried out to show the viability of *nci* as a potential comparative measure to compare the connectedness among all the potential node pairs connectivity.

4.4.4.1 First Scenario

Consider a scenario which consist of two nodes, one stationary and the other is mobile, as shown in Figure 4.8. Initially the mobile node is positioned at an angle of arrival to the stationary node. Each node produces a periodic node connectivity packet, i.e. CONN packet, which is a one-hop transmission packet, design to identify nodes that are within transmission range. This node and their parameters are recorded in a form of a node state and saved into a node state cache. Readings of *nci* were taken for various speeds and for various angles of arrival. The results were tabulated for *nci* with speeds of 1 m/s, 2 m/s, 3 m/s, 4 m/s, 5 m/s, 6 m/s, 10 m/s and 15 m/s and graphically presented as shown in Figure 4.9 through to Figure 4.16 respectively. For each graph values of *nci* for each angle of arrival of 0, 30, 45, 60, 70 and 80 degrees were drawn. For each speed, the time of departure is less for small angles of arrival. The *nci* on the other hand showed that as the time of departure is decreasing, the value of *nci* is increasing. For the purpose of data transmission, it would be convenient to choose a node for which *nci* is the lowest among the entire neighbouring

83

nodes. As can be seen, as *nci* increases, the expected connectivity time is reduced.
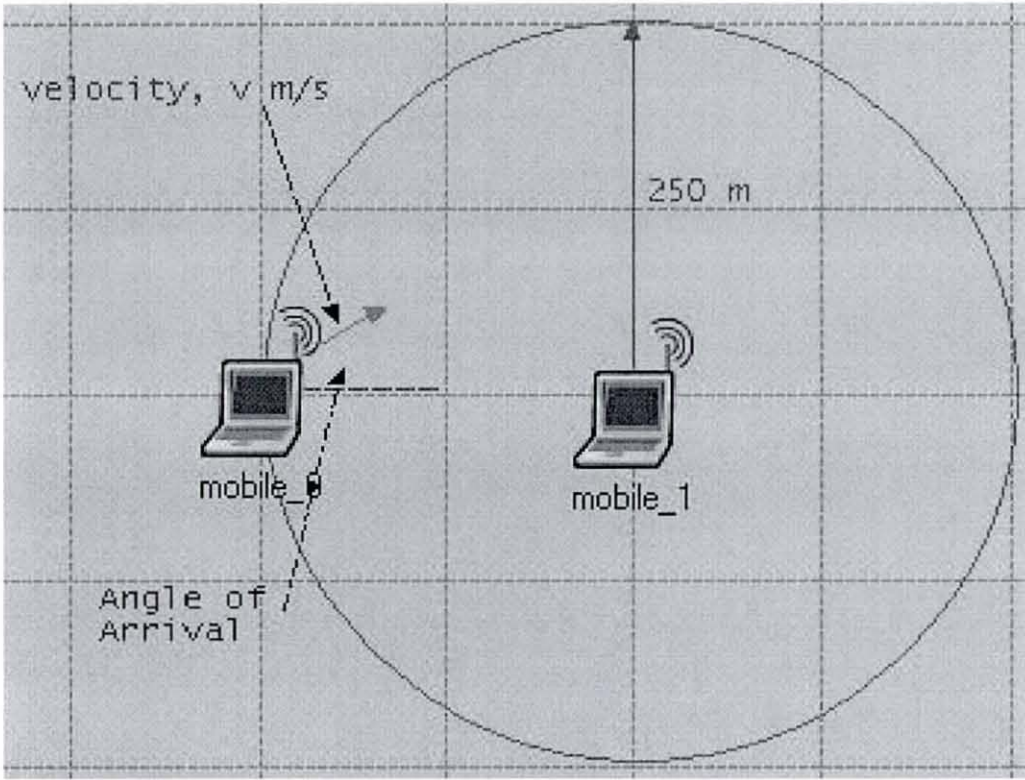
Figure 4-8 Scenario to Determine *nci*

The results for the *nci* values for a number of velocities that were tabulated in Figure 4-9 through to Figure 4-16 are now redrawn to show the relationship between *nci* and different velocities. Figure 4-17 shows the redrawn graph. It illustrates the effect of velocity on *nci*. It can be concluded that the greater the velocity, the higher the *nci*. If there are two node-pairs with different relative velocities, then the one which has the least *nci* value will be chosen at that instant as its performance is the best in terms of how long the node-pair connection lasts. Hence this confirms that *nci* could be a unique comparative index, to select which node pairs that provides longer node connectivity.
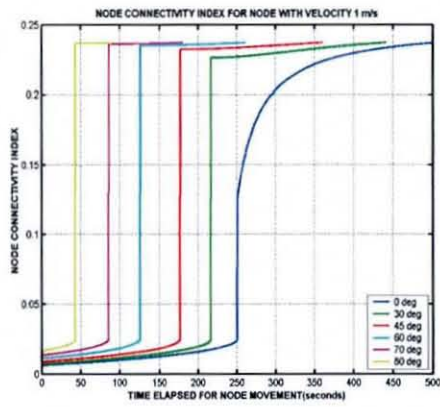
Figure 4-9 *nci* for velocity 1 m/s
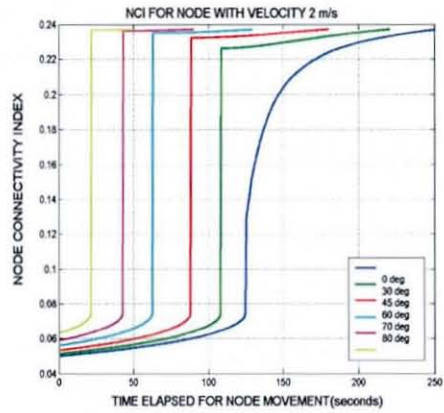
Figure 4-10 *nci* for velocity 2 m/s
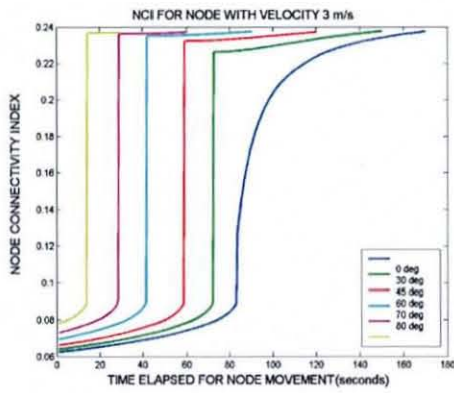
Figure 4-11 *nci* for velocity 3 m/s
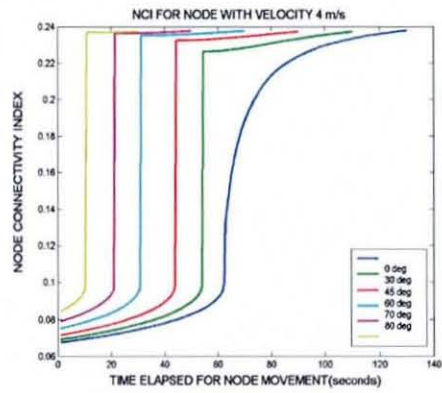
Figure 4-12 *nci* for velocity 4 m/s
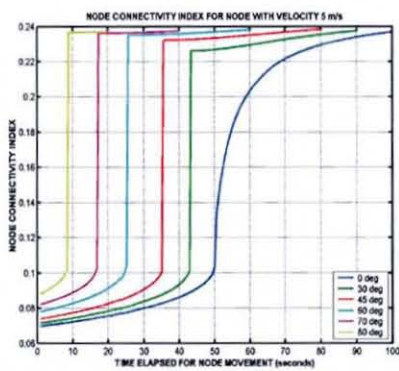
Figure 4-13 *nci* for velocity 5 m/s
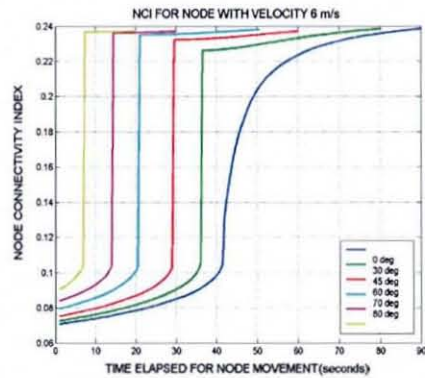
Figure 4-14 *nci* for velocity 6 m/s

85
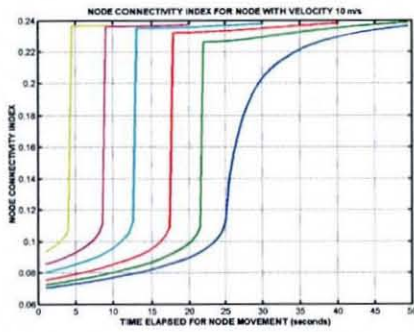
Figure 4-15 *nci* for velocity 10 m/s   Figure 4-16 *nci* for velocity 15 m/s
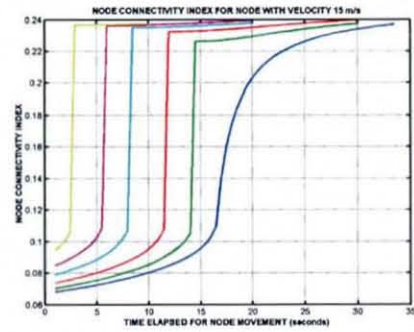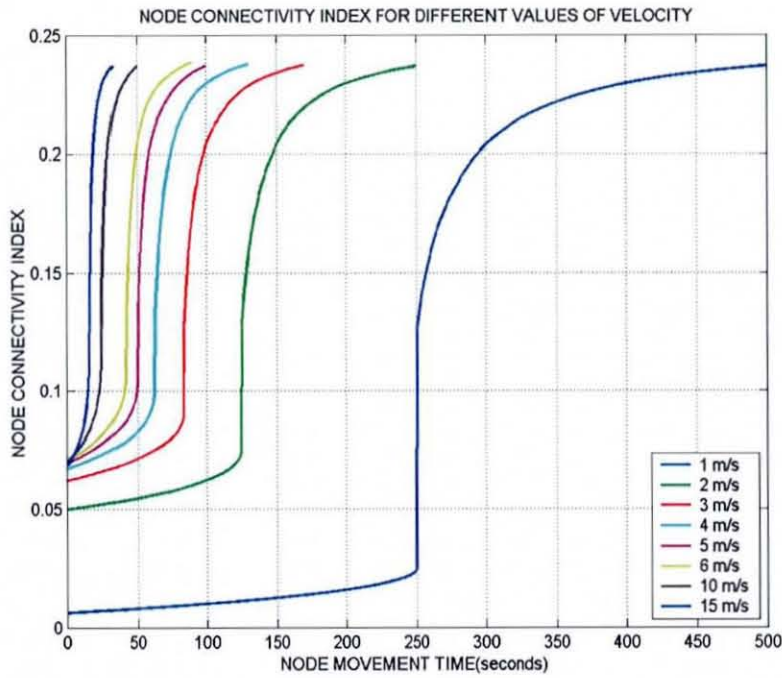
Figure 4-17   NCI for various node velocities

*4.4.4.2 Two Scenarios to Demonstrate the Effect of **nci***

The first scenario consists of one mobile node, **n1** and two stationary nodes, **n2** and **n3** as shown in Figure 4.18(a). The mobile node is set to move in a straight line in the direction where it enters the radio transmission range of **n2** and **n3** at time 0 second and 280 seconds respectively. At time 0 second, **n1** and **n2** are connected since **n1** is just within the transmission range of **n2**. At that instant **nci** is about 0.01. From 0 second to 280 seconds **n1** is always in connection with **n2** such that the value of **nci** (n1<->n2) varies from ~0.01 to ~0.23. If the selection process occurs during this period, no change of path will take place. Upon reaching the 280th second, **n1** enters **n3** connectivity range.

The value **nci** (n1<->n3) is ~0.01. At this stage **n1** has two choices of connection. In the event there is a selection process, **nci** (n1<->n3) is much less than **nci** (n1<->n2), hence the node-pair connection changes to n1<->n3.

The graph in Figure 4.18(b) shows the reading of **nci** for node **n1**. The **nci** values for node **n1,** are the **n1** node states for neighbours **n2** (n1<->n2) and **n3** (n1<->n3). From the graph in Figure 4-18(b) , at time ~280 seconds, the **nci** (n1<->n2) is ~0.23 and **nci** (n1<->n3) is ~0.01. If the line of **nci** (n1<->n2) is extrapolated from that point until the end, the time left is ~180 seconds. On the other hand, if the line of **nci** (n1<->n3) is extrapolated from that point until the end, the time left is ~ 460 seconds. Hence it can be deduced that the element of node state represented by **nci** , does infer connectivity between node-pair. Next, any node-pair connectivity may or may not be part of the potential routes.
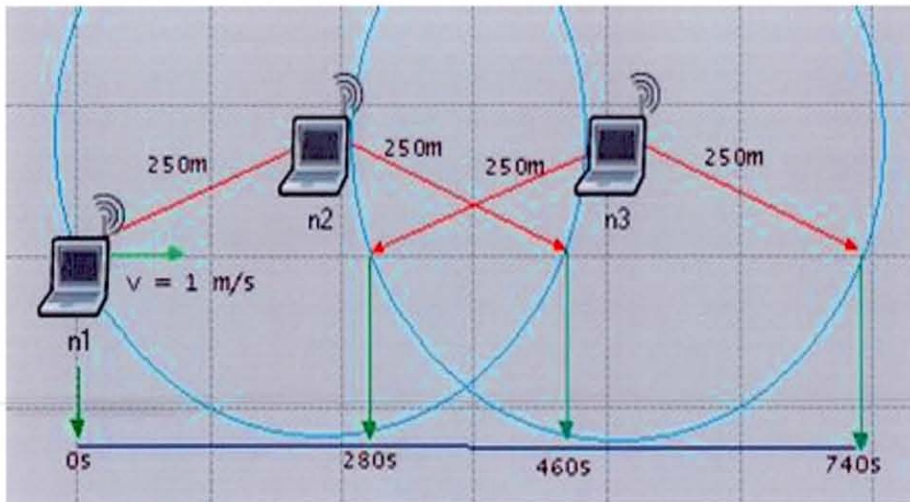
(a)
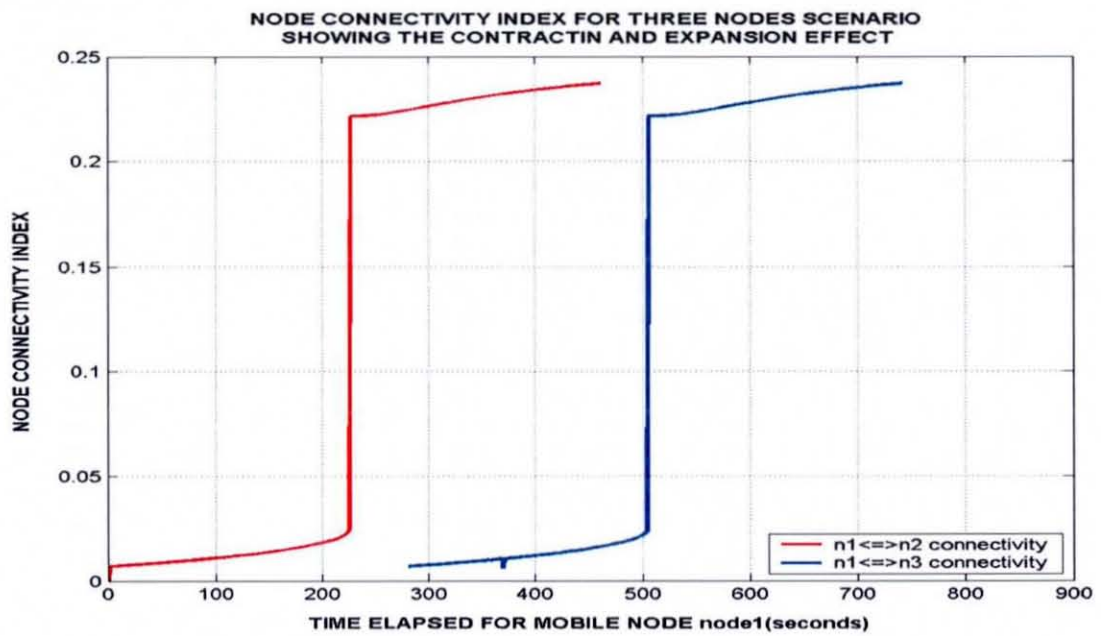
(b)

Figure 4-18 Two stationary nodes and one mobile node
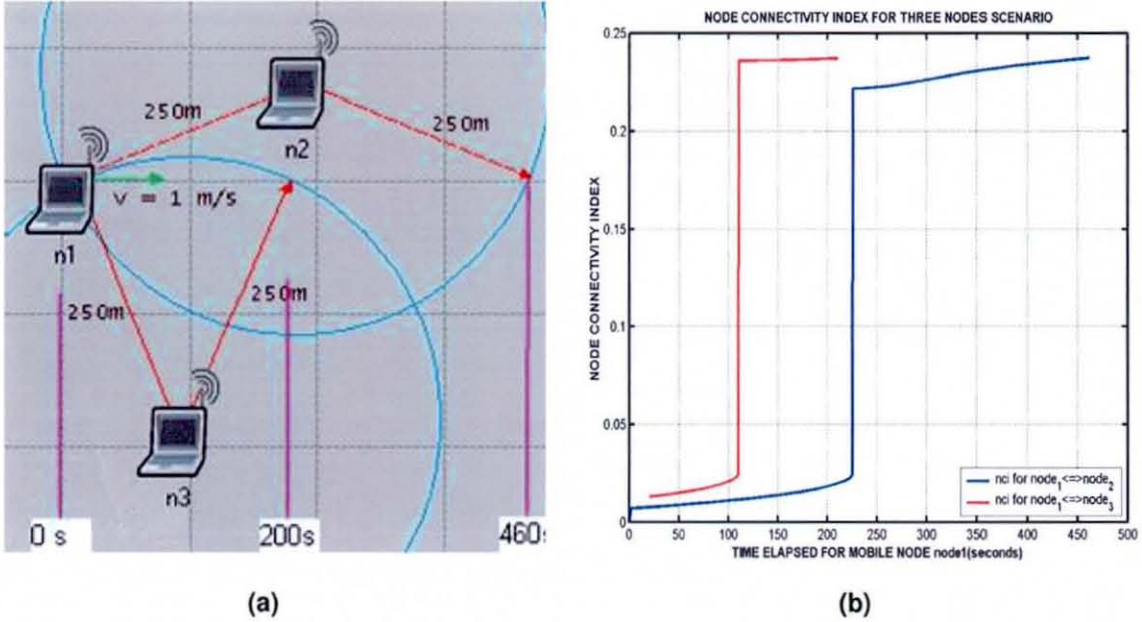
Figure 4-19 The Effect Of Two Stationary Nodes In Vicinity Of One Mobile Node

The second scenario consists of three nodes, one mobile node and two stationary nodes, with the setup as shown in Figure 4.-19(a). The mobile node, **n1** initiates its movement in a straight line with a speed of 1 mps. It immediately enters the transmission range of **n2** and then a little while later into the range of **n3**. The graph in Figure 4.19(b) shows the readings of **nci** against the time elapsed for mobile node **n1**. From the graph **nci** (n1<->n2) is less than the **nci** (n1<->n3) from the time it enters the range of **n3** until the time of departure at 200 seconds. Node **n1** connected to **n3** traversed a shorter distance compared to **n1** connected to **n2**. Connectivity time for **n1** to **n2** is much longer, hence when **nci** for both connections are compared, the one which has longer connectivity in the expansion region offers the least **nci**. In the event, the selection process occurs at any instance between 0 second and 200 seconds, the connection of (n1<->n2) would be chosen.

It is noted that the value of **nci** (n1<->n3) just before **n1** is within the transmission range of **n3** , is not zero but that it simply did not exist, since both are out of range of each other. At that point in time **n3** is removed from the **n1**'s neighbour's list.

89

Algorithm 4.1 : Calculating the npcm, npem, nci

---

00 $S \leftarrow \{i, H_i, B_{AVA}, D_{ij}, nci, t_{CUR}, t_{PRV}, P_{CUR}, P_{PRV}\}$;

01 If ($P_{PRV} > P_{CUR}$) then

02 $npem = \left(\dfrac{1}{t_{CUR} - t_{PRV}}\right) \times \left(\dfrac{1}{P_{PRV}} - \dfrac{1}{P_{CUR}}\right)$;

03 $nci = 0.25 - \left(\dfrac{1.0 \times 10^5}{8.0 \times 10^5 - npem}\right)$

04 elseif ($P_{CUR} > P_{PRV}$) then

05 $npcm = \dfrac{1}{t_2 - t_1}\left(\dfrac{1}{\sqrt{P_{CUR}}} - \dfrac{1}{\sqrt{P_{PRV}}}\right)$;

06 $nci = \dfrac{1.0 \times 10^5}{8.0 \times 10^5 + npcm}$;

07 endif;

08 $S \leftarrow \{i, H_i, B_{AVA}, D_{ij}, nci, t_{CUR}, t_{PRV}, P_{CUR}, P_{PRV}\}$

4.5 Summary

In this Chapter, the idea of formulating the node connectivity into a form suitable for the purpose of monitoring the dynamic of any given node pair was proposed. It had been able to show that the node connectivity time constitutes the length of time where node-pairs are connected and communication sessions can be established. By analytical calculation and simulation, the node connectivity time depends on the relative velocity of the node-pair. It shows that there is an upper limit and a lower limit for the connection time. Between

these limits communication sessions can be carried out. In order to monitor the quality of connection between a node-pair and for the whole route which is made up of a collection of node-pairs, *npem, npcm* and **nci** are introduced. The **nci** is shown to be a viable way to estimate the quality of node-pair connection in terms of its timeliness. In the next chapter **nci** will be used as one of the fitness metric for GA to identify and select the best path to the destination.

CHAPTER 5
NODE STATE MONITORING PROTOCOL FOR MOBILE AD HOC NETWORKS

5.1 Introduction

The goal of QoS routing is to provide an application with a connection which can sustain the requested bandwidth and delay requirement. In MANET, a wider range of environment parameters are encountered, and a different approach using QoS routing techniques is called for. QoS Routing has received considerable attention by a number of researchers [39][40][13][41] . A number of approaches have dealt with QoS support in multichannel wireless networks such as TDMA and CDMA. Shared channel networks such as the IEEE 802.11 [42] do not perform well for QoS routing. The characteristics of the shared channel do not provide a unified view of the medium to all nodes. The following issue must be addressed: the nodes must obtain information about their environment and must react to topology changes. Focusing on the first issue, this must be addressed before applying various node information triggered by topology changes. The node must capture the network information instantaneously and then be saved as a Node State. The monitoring function ultimately would improve the process of QoS route selection. This chapter

focuses on the resource information that would be used to parametricise QoS route selection mechanism. Generally, the QoS routing mechanism used resource monitoring for admission control and for QoS routing enforcement. In our work we will be applying this information as an input to the GA module for the computation of route selection.

The remainder of the chapter is organised as follows. Section 5.2 defines the Node State, outlining its usefulness in the QoS routing and discusses its advantages and how it infers node connectivity. Section 5.3 describes in detail the process of monitoring and capturing all the node state information. In Section 5.4 discusses the techniques of node bandwidth measurement . Section 5.5 describes the delay estimation. Section 5.6 describes the Node State Monitoring protocol. The Chapter ends with a summary in Section 5.7.

5.2 Node State as a Component for QoS Routing

5.2.1 Definition of Node State

Definition 5.1: The Node State is defined as any attribute of a node which can be measured, quantified and maintained. When network topology changes, the Node State needs to be updated. In general, the node's attributes would be able to describe the node's characteristics instantaneously.

Within the routing taxonomies, protocols are classified as link state, distance vector or flooding algorithms [13]. They are further categorised into various schemes such as proactive schemes, reactive routing and predictive routing scheme, ticket-based routing scheme and scheme based on bandwidth calculation [5] . As a result of frequent topology changes in MANET, the emphasis shifted from ways to calculate routes to finding ways to discover and disseminate topology information efficiently.

Node connection is normally understood based on the discrete link between a pair of nodes. The node pairs usually advertise the existences of links and are connected. In link-based protocols, this information would be used to calculate the routes, joining all node pairs, thus forming feasible paths. The concept of Node State is proposed. Node State is a general concept similar to link state for the purpose of analysing networking environment. The Node State requires two

capabilities, location awareness and signal strength measurement. Hence, relative position, power measurements and relative velocity provide sufficient information to determine the node-pair connectivity. Due to mobility, Node State changes as frequently as the link breakage rate. Node State can provide the capability to predict connectivity between mobile nodes.

5.2.2 Advantages of Node State

In this thesis the Node State is regard as the basis of a QoS routing protocol where it has advantages over the link-state mechanism in several ways.

One State per Node

It is a one state per node. In a link-based protocol, there is one state for a pair of nodes. For **n** nodes there are **n(n-1)/2** link states, but only **n** Node States.

Node State as Information Cache

Node States could generate more information about the network topology that is relevant to QoS routing. The contents are updated when network topology changes, as well as being updated periodically. No restrictions are imposed on the number of Node States to be collected, cached and disseminated. The information is normally gathered during the Non-Disjoint Multiple Routes Discovery protocol operation, and during any packet arrival event.

Prediction of Future Topology

The Node State can be used to predict the node departure time where the node connectivity will be broken. Hence this is used as a QoS metric whereby the measure of node connectivity is transformed into an index to indicate any length of time before the connection breaks. The index would be useful for the QoS route selection using GA.

Quality of Nodes

When a node moves, its state also changes. If the node is stationary and other nodes have moved, then only those mobile nodes need to change their

state. Most of the states of the stationary node do not change and so do not need to be disseminated as often.

Node State as Support for QoS Routing

The Node State provides a unified mechanism to collect and to disseminate the required state information. Thus it supports the realisation of QoS routing algorithms. The proposed GA based route selection algorithm would consider the fitness of each route. These fitnesses are derived from all node states in the potential routes.

5.2.3 Inferring Connectivity and Topology Changing

Inferring connectivity involves predicting the node-pair connectivity index, ***nci***. Three observations on signal propagation are relevant to the understanding of the approach taken in this thesis to predict the node-pair connectivity index: (1) power measured at a node generally increases as a power law function of distance, (2) power may vary over short distances due to multipath effects, (3) radios can receive and detect signals with strengths that vary over a wide dynamic range. With the assumption that all nodes operate in the free space environment, then an approximation that is conservative in its estimate is suitable. The free space propagation model[38] can be used to provide such an estimate. As with most large scale radio wave propagation models, the free space model predicts that the received power decays as a function of the transmitter-receiver separation distance raised to some power, following a power law function. Measuring power loss is tractable, so long as sources can reliably specify the power they are using when transmitting, and destinations can determine the range between nodes and the strength of the signals they receive. Inferring quality connectivity is a three step process. The propagation model provides the loss of received signals, in term of power measured when a packet was received. The second step determines whether it is within the power received threshold. The third step is how to associate the quality connectivity with the time a node is within the transmission range of its neighbours. **Table 5.1** provides some examples of Node States. These

objectives may be combined to form additional metrics. In this section the used of Node State as part of our QoS routing protocol is proposed.

Table 5-1 List of Node State and Their Definitions

Name	Definition
Node Address	IP Address of this node (IPv4)
Neighbours Address	IP Address of the neighbour node (Ipv4)
Node Degree	The number of neighbour nodes in the vicinity of this node.
Node Bandwidth Consumed	The amount of bandwidth that has been used for flows via this node and neighbours node
Node Bandwidth Available	The amount of bandwidth that is free to be used for future flows via this node and neighbours node
Node Bandwidth Capacity	The total amount of bandwidth available for usage.
Node Connectivity Index	Node Connectivity Index is a positive value which describes the quality of connectivity between any two adjacent nodes.
Delay	Represents the end to end delay of all the packets received by the wireless LAN MAC of this node from its neighbour and forwarded to the higher layer.
Medium Access Delay	The total of queue and contention delays of data packets received by WLAN MAC from the higher layer. For each packet, the delay is recorded when the packet is sent to the physical layer for the first time.
Velocity	The rate of movement of the node in m/s.
Power Due To Curr Received Packet	Signal Strength measured due to the arrival of a packet from a neighbouring node.
Power Due To Prev Received Packet	Signal Strength measured due to the arrival of a packet from a neighbouring node, before the arrival of current packet.
Current Packet Arriv. Time	Current packet arrival time.
Previous Packet Arriv. Time	Previous packet arrival time.
Packet Size	Size of payload packet that arrive at the MAC layer.

5.3 Node State Monitoring Model

QoS routing performance is difficult to predict. Many discussions occur on the effects of these higher layer parameters on the QoS performance, such as link capacity, traffic pattern and even different transmitting schemes. Similarly, the

effects of the low layer parameters can be considered as unpredictable. On the other aspects, QoS routing can be applied with imprecise information, which is modelled by probability density functions [43].

5.3.1 System Model for QoS Parameters Monitoring

5.3.1.1 Network Model

A Mobile Ad Hoc Networks (MANET) is modelled as an undirected graph $G = (E, Q\{nci, B_{AVA}, D_{E2E}, D_{MAC}\})$ where E is the set of all mobile nodes, and Q is a set of QoS parameters that determine the quality of the routes. The symbol nci, B_{AVA}, D_{E2E} and D_{MAC} represent node connectivity index, bandwidth available, end to end delay and MAC delay respectively. A node-pair connectivity (u, v) is said to exist between nodes (u, v), if both are in the transmission range of each other. The quality of the connectivity is given by nci. The quantities E and Q are ever-changing, since nodes may join and leave the network.

5.3.1.2 Node Mobility Model

Mobility characteristics, M_i, of the node i, is characterised by its relative velocity $V(i,j)$ with its neighbour j, the initial direction θ_i and the movement function F_{RWP} can be represented by

$$M_i = (v(i, j)), \ \theta, \ \{F_{RWP}\} \ )$$

(Eqn. 5-1)

The movement function represents the random waypoint mobility model (RWP) [11], which is available within OPNET Modeler. The mobility model will affect the node connectivity and result in variation of node-pair connectivity time, depending on the direction of the node's movement towards each other or away from each other. It is termed as a contraction and expansion metric. The contraction and expansion metric leading to the derivation of node connectivity index, nci has been considered in **Chapter 4**.

97

5.3.1.3 Node Communication Coverage Model

The node's communication coverage model is represented by a circular area assuming transmission from isotropic antennae. The node's transmission range $R(i, j)$ from node i to node j is used to represent the coverage area. The circular area is determined by the transmitter power P_t of node i and receiving power P_r of node j. Nodes situated within this coverage range are considered neighbours. The relationship between the node and its neighbours, which is represented by the Node State, can be monitored and various QoS parameters can be extracted. The transmission range of the node is assumed to be 250m. Additionally, the free space path loss propagation model is assumed throughout this thesis. The Node State is dominated by two factors: (1) the lower layer parameters, such as node mobility and radio channel characteristics; and (2) the higher layer parameters, such as offered traffic load in the form of packet size and packet interarrival rate. During the exchanging of the neighbour's local information in the network model, a node's lower layer parameters can be transferred to the network layer.

5.3.1.4 Node Pair Connectivity Model

The node pair connectivity model is used to predict the local performance whereby a QoS routing module can react upon it. If the distance between two nodes i and j in a route is denoted as d_{ij}, then the condition for the existence of connectivity between the two mobile nodes is $d_{ij} \leq R$. The distance between any two mobile nodes is obtained from the radio device on each node, based on signal strength. From Eqn. 4-11,

$$d_{ij} / \sqrt{k} = 1 / \sqrt{P_i} \qquad \text{(Eqn. 5-2)}$$

where
$$k = (P_t G_t G_r \lambda^2) / ((4\pi)^2 L). \qquad \text{(Eqn. 5-3)}$$

Using the above relationship, the quality of node connectivity was modelled in terms of a numerical index, *nci*, which is described in **Chapter 4**.

5.3.2 Monitoring Scheme

In this section the protocol is described in terms of its mechanism for monitoring the node. The monitoring scheme involved the monitoring of packet types, the Node State parameters extraction and QoS parameters distribution.

5.3.2.1 Monitoring the arrival of various packet types

There are five types of packet that arrive at a node: Data, RREP, RREQ, RERR, ACK and CONN. By recording two consecutive packet arrivals from the same neighbour node, the neighbour's node IP address, may be identified, the signal strength due to the arrival of each packet, the node degree, the distance to the neighbour's node and the relative velocity of the node. The monitoring protocol is installed on every node and monitors the nodes that receive any type of packet from its neighbour. No extra packets need to be used for the monitoring process. The QoS parameters are distributed by piggybacking on the RREP packet. The information collected by each node is maintained appropriately. The temporary characteristic of the monitoring parameters is determined by the monitoring requirements. Based on its temporary characteristics, all information collected from the node neighbours is time dependent information. When the node moves, all the collected parameters need updating while stale information is deleted.

5.3.2.2 Extracting Node State and Distributing QoS Parameters

The parameters of concern are derived mainly from the MAC layer and the physical layer. These parameters are the results of the model's interaction with the neighbouring nodes. The inter-node interactions are calculated. The parameters collected will be transferred to a higher layer for the calculation of QoS route performance. These parameters include the bandwidth consumption (B_{CON}), bandwidth available (B_{AVA}), end to end delay (D_{E2E}), medium access delay (D_{MAC}), the relative velocity (V_{REL}), the power due to current (P_{CUR}) and previous packet (P_{PRV}) arrival and the time when the current (t_{CUR}) and previous (t_{PRV}) packet arrived. The QoS parameters are collected by the RREP packet. It follows the routes discovered by the RREQ packets but going in

reverse route. It attaches the QoS parameters into the RREP packet and is carried towards the source

5.3.2.3 Update Mechanism For Node State

The node state information which is being collected and cached, reflects these activities of the node and its surroundings. Each node updates all the Node State parameters regularly. The update mechanism operates at two levels; periodic and aperiodic. The periodic mechanism updates at a constant time interval of one second. It is triggered by the Non-Propagating RREQ packet, by setting the TTL field to 1. The packet is the connectivity packet, which is labeled as CONN packet. This periodic packet which causes the update, only operates when nodes have no packet in the queue to send. A periodic updating is triggered by the arrival of various kinds of packets, RREQ, RREP, DATA, ACK and RERR, from the node's neighbours.

5.4 The Bandwidth Measurement

To design an efficient QoS routing, it is very important to get accurate information on the consumed bandwidth and available bandwidth. In MANET environment, this estimation is not so easy to compute, as the perception of the medium used is different from one mobile node to another. Therefore, to determine precisely the available bandwith on its own, a node has to know the bandwidth available to the nodes which it share the medium with, in order not to penalise them.

5.4.1 IEEE 802.11 DCF MAC Protocols

IEEE 802.11 uses CSMA/CA implementing MAC layer [44] . Stations participating in the network use the same CSMA/CA to coordinate access to the shared medium. A station that wishes to transmit must first listen to the medium, to detect if another station is using it. If so, it must defer until the end of that transmission. If the medium is free, then that station may proceed. In IEEE 802.11, carrier sensing is performed by the method of *physical carrier sensing* and *virtual carrier sensing*. Physical carrier sensing detects the presence of WLAN users by analysing all detected packets, and also detects

channel activity from relative signal strength from other sources. The virtual mechanism is referred to as the Network Allocation Vector (NAV). The NAV is a way of blocking other nodes from transmission. A node's medium is considered busy if either its virtual or physical carrier sense mechanisms indicate busy. Before a station can transmit a frame, it must wait for the medium to have been freed for some minimum amount of time known as the Inter-frame Space (IFS). A priority mechanism for access to the shared medium can then be established. For the DCF, these are the Short IFS (SIFS) and the DCF IFS (DIFS). The MAC protocol defines instances where each IFS is used to support a given transmission priority. A node wishing to transmit a packet must first wait until its carrier sense mechanism indicates a free medium. Then, a DIFS will be observed. After this, the node then waits an interval amount of back-off time, BO, before transmitting. This additional deferral will minimise collisions between the nodes that may be waiting to transmit after the same event. Before a node can transmit a frame it must perform a backoff procedure. The node first waits for a DIFS time upon noticing that the medium is free. If the medium is still free the node computes the Backoff Timer. The node will wait either until this time has elapsed or until the medium becomes idle, whichever comes first. If the medium is still free, the node begins its transmission. The receiving nodes respond back to the sending nodes with an ACK packet to indicate a successful reception. A lack of ACK indicates unsuccessful transmission to the sending node. The timing diagram for a basic DCF is shown in Figure 5-1. An interesting fact about NAV is that it can be used to monitor when the channel is either busy or idle.

5.4.2 Contention in Wireless Network

5.4.2.1 Theoretical Maximum Throughput

In order to calculate the capacity bound for the constraint on the GA QoS route selection algorithm, a series of simulation experiments were conducted using OPNET Modeler. The aim was to find the maximum throughput a node

can have in the shared medium environment, and how it relates to the number of nodes and the offered traffic load.

In the simulations, the free space propagation model was used and omni-directional antennas were assumed. The wireless channel capacity was set to 5.5 Mbps. Six nodes moved randomly in an area of 250m by 250m. The transmission range of each node is set to 250m. The routing protocol used is the Best Effort DSR and MAC protocol CSMA/CA IEEE802.11. Throughput readings are considered as the bandwidth consumed by the nodes. The traffic used in the simulation is generated by a video connection, modeled as an 819.2 kbps video conference sending 512 byte packets at a rate of 200 packets per second. The requested bandwidth is therefore 819.2 kbps. The total throughput was expected to be 4.9 Mbps, which is less than the channel capacity. The video connections were established between any two nodes selected randomly. The numbers of connections were increased one by one until all connections are active. The whole setup is shown in Figure 5-2. The total bandwidth consumed by every node is shown as in Figure 5-3. The maximum throughput obtained from the graph is approximately 2.2Mbps. This is much less than expected. This is due in part to the fact that a portion of each transmission is performed at lowest data rate, 1 Mbps. In addition, interframe spacing and control packet overhead further decrease the effective bandwidth.
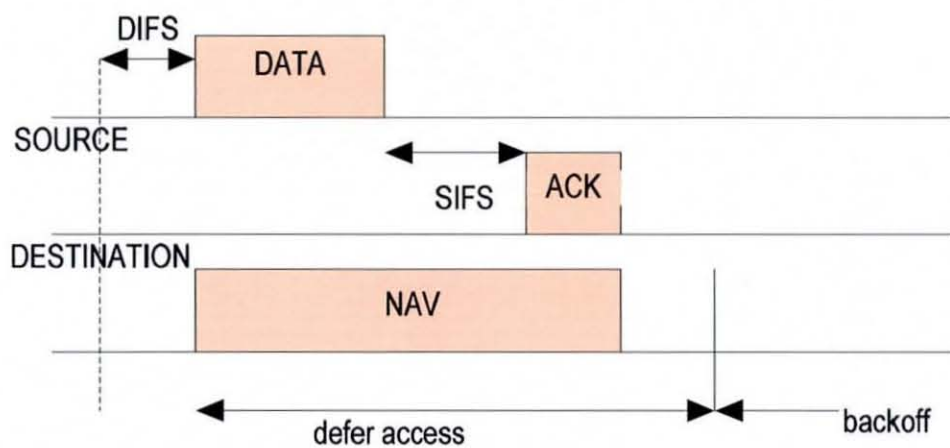
Figure 5-1 DCF Basic CSMA/CA Timing Diagram Showing NAV [44]
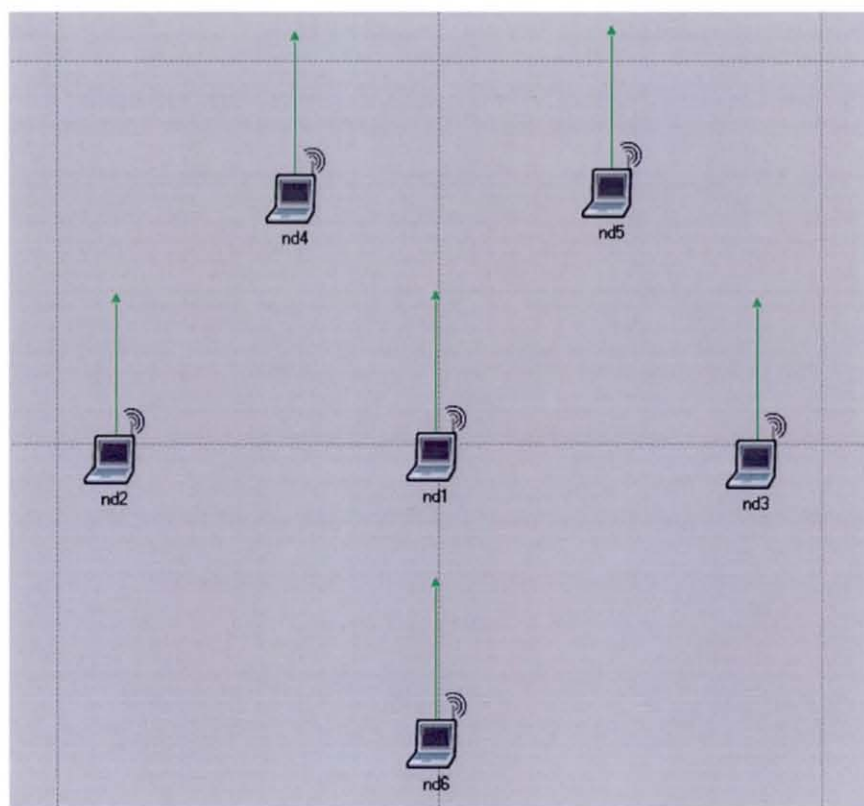
Figure 5-2 Mobile Nodes for the Calculation of Bandwidth Consumed
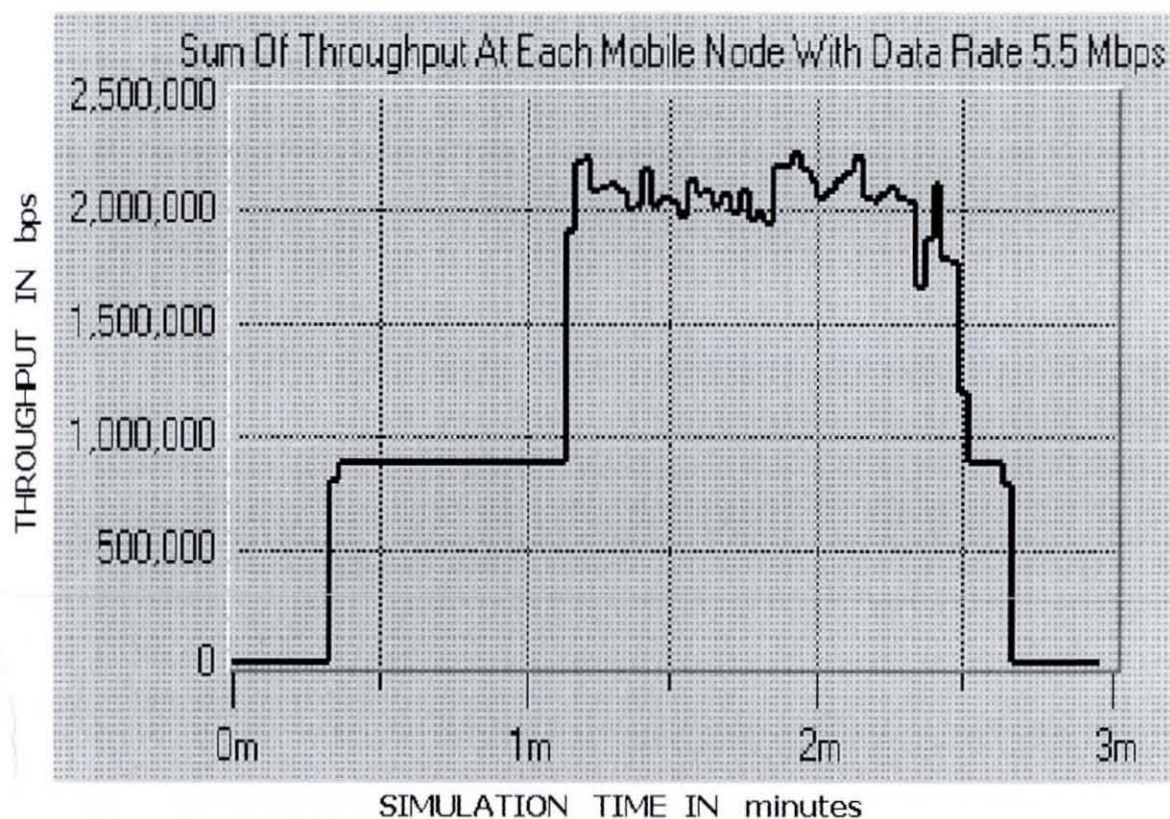
Figure 5-3 Sum Of Throughput from Each Mobile Node

Also, in this scenario where there are multiple sources, there are periods when no transmission occurs, since all nodes are idle or backing off. This characterises the contentious behaviour of nodes. Jun *et. al* [45] has produced a general result relating to this effect, and is stated as the Theoretical Maximum Throughput. Using this result for an offered load with a given MSDU, the maximum throughput can be calculated. The theoretical maximum throughput (TMT) [45] of the IEEE 802.11 MAC is given by

$$TMT(x) = 8x/(ax+b) \quad Mbps \qquad \text{(Eqn. 5-4)}$$

where x is the MSDU size in bytes, **a** and **b** are parameters specific to different MAC schemes and spread spectrum technologies. Table 5-2 provides the example of TMT values for stated data rates of 1, 2, 5.5 and 11 Mbps as applied to a node with CSMA/CA operating under DSSS and HR-DSSS modulation scheme and MSDU of 512 bytes.

104

Table 5-2 TMT Values For CSMA/CA, DSSS and HR-DSSS Scheme,
with MSDU 512 Bytes [45]

Data Rate	a	b	TMT
1 Mbps	8	1138	0.78 Mbps
2 Mbps	4	1002	1.343 Mbps
5.5 Mbps	1.45455	915.45	2.467 Mbps
11 Mbps	0.72727	890.73	3.243 Mbps

In MANET, there is a need to consider the effect of interference of traffic from neighbouring nodes that are possibly more than one hop away, but within the transmission range. This is the reason why the effective throughput is less than TMT. From Figure 5-3 the estimated maximum throughput is approximately 2.2 Mbps. Hence it can be concluded that, when calculating the bandwidth available the TMT value should be used.

5.4.2.2   The Need for Bandwidth Estimation

A robust and timely estimation of available instantaneous bandwidth or throughput at a node is necessary for effective control mechanism, in order to support the QoS routing algorithm. In a MANET, with CSMA/CA based MAC layer IEEE802.11, the bandwidth available at a node at any given time is affected by fading and shadowing effects in the wireless channel, changes in the network topology and variations in the traffic due to the neighbouring nodes. Hence, an estimation technique is needed that is timely, responsive to bandwidth changes and non-intrusive. Noticeably these requirements are not sufficiently well addressed in the existing bandwidth estimation techniques [46] [47] designed for wired networks.

5.4.2.3   The Contention Experiment

The first challenge of QOSRGA routing protocol is the ability to measure the available bandwidth instantaneously. Each node sees a different channel state and the available bandwidth in the network is not as simplistic as for only a single node concept. To understand this complexity, bandwidth availability may

be classified as local available bandwidth and neighbours' available bandwidth. The former is due to a given node, and the latter is due to its neighbour's node. Neighbours' node bandwidth is the maximum amount of bandwidth a node can use, without depriving the existing flows within the node transmission range. Local available bandwidth is the amount of unconsumed bandwith as seen by the given node.

To demonstrate this relationship, a simple simulation experiment is shown using the Opnet Modeler. The MAC layer protocol is the 802.11 with basic access CSMA/CA. The configuration and the setting of the nodes are shown in Figure 5.4 and Table 5.3 respectively. Three flows are established, each of 200 packets/s, CBR traffic with packet size of 512 bytes. The simulation was run for 600s. At time 40s, **node_1** initiated flow 1 to **node_0**. At 100s **node_3** initiated flow 2 to **node_2** and at 160s **node_5** initiated flow 3 to **node_4**. The throughput, queue length and the delay statistics were obtained as shown in Figure 5.5, Figure 5.6 and Figure 5.7 respectively. Figure 5.6 shows the queue length, to indicate that there are always packets to be transmitted. It is in a saturated state of the node after 100s.
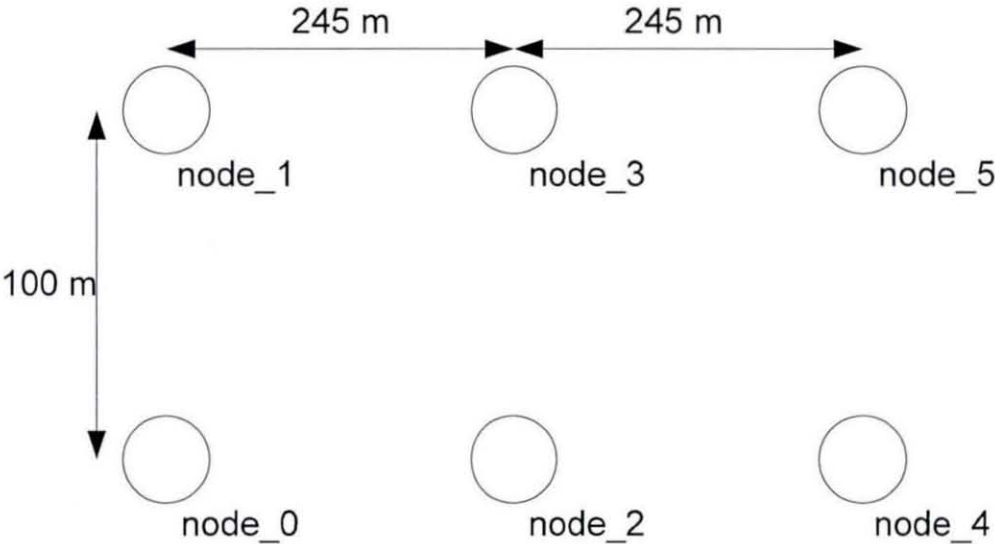
Figure 5-4 Simulation Topology

Table 5.3 Setting of the Nodes For The Simulation Experiment

Transmission Range	250m
Data Rate	2 Mbps
Traffic Models	CBR, 200 packets/c, Packet Inter-arrival Time 0.005 s, Packet Size 512 bytes, TX Data Rate 820 kbps
Simulation Time	600s
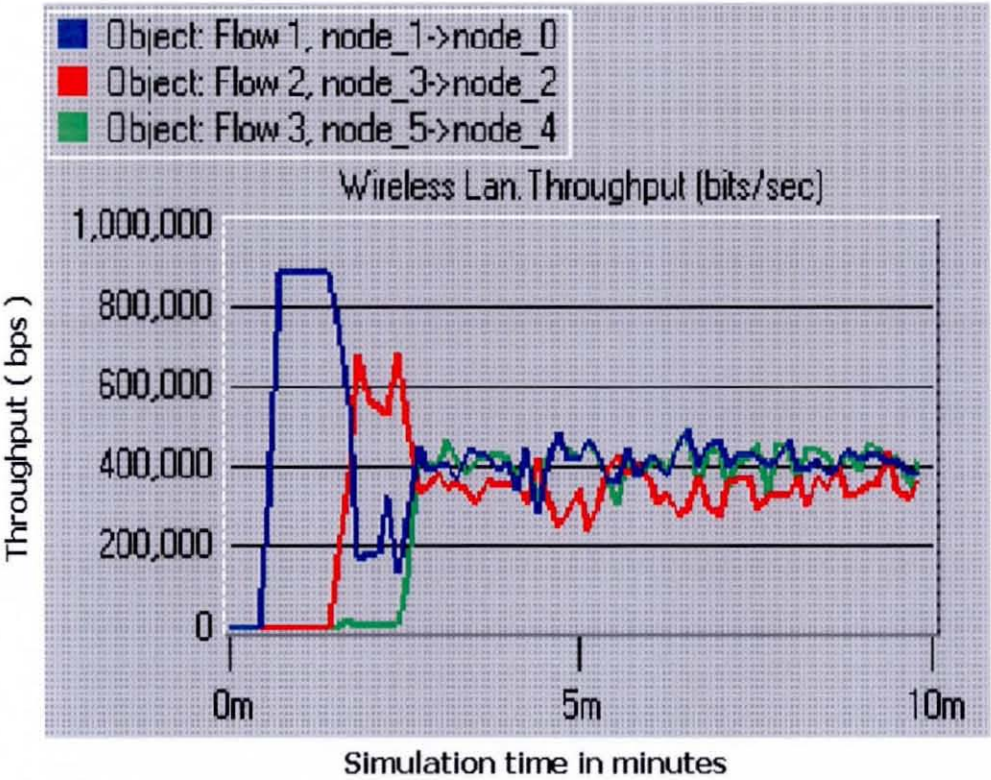
Figure 5-5 Throughput for Flows 1, 2, 3
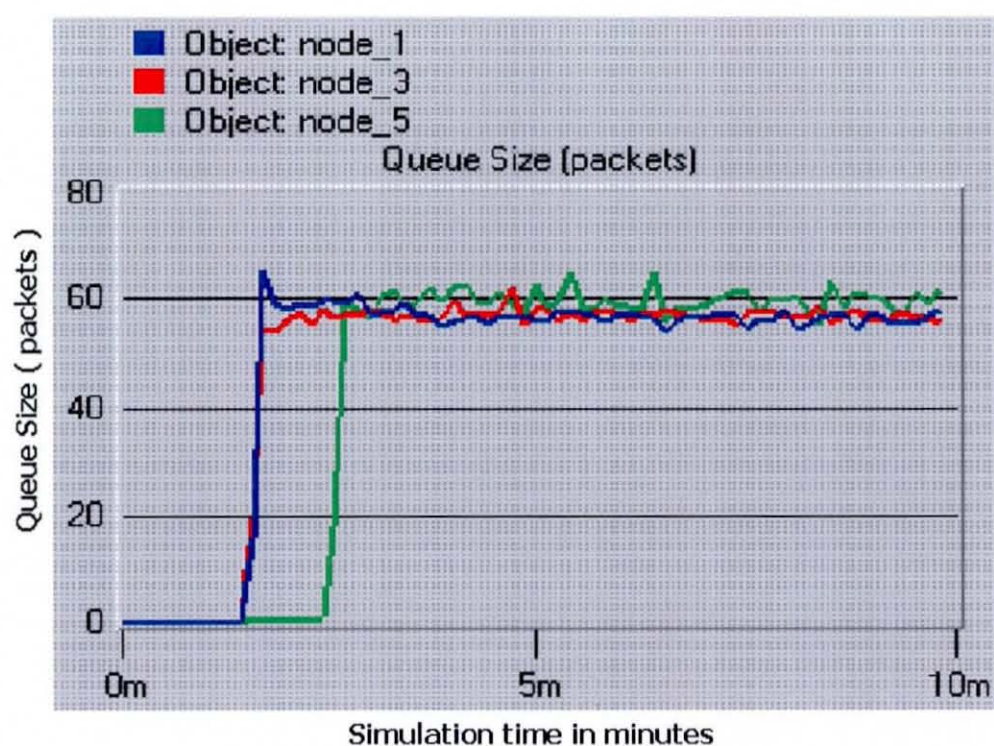
Figure 5-6 Queue at the Node's Transmitter
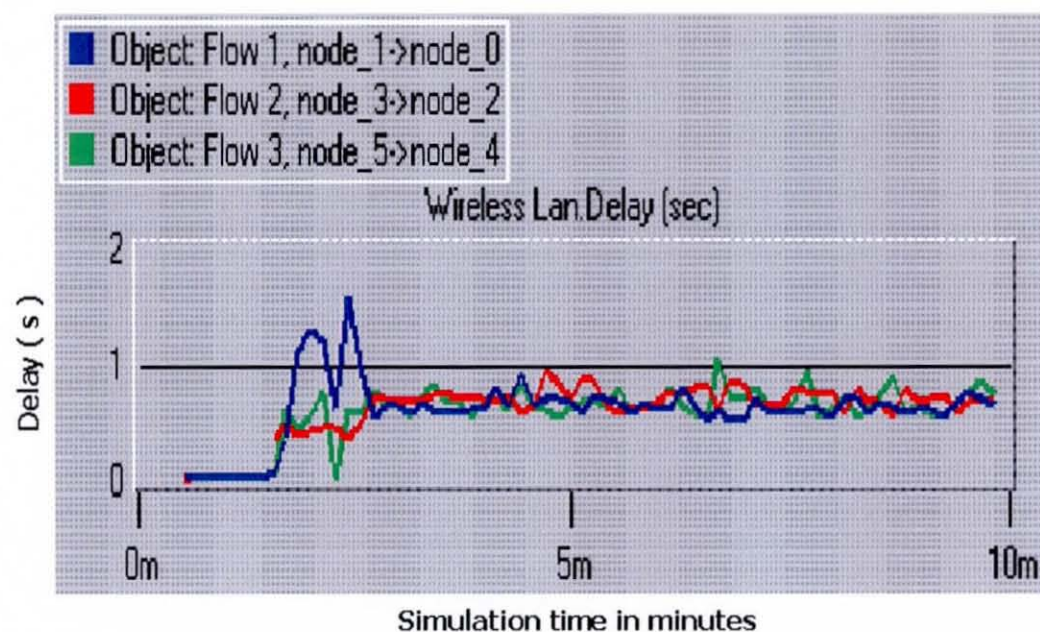
Figure 5-7 Delay for flows 1,2,3

108

At time 40s, when flow 1 initiates, with bandwidth demand B_{REQ} of 820 kbps, bandwidth available B_{AVA} at node_0 is approximately 1.3 Mbps due to TMT as in Table 5.3. No contention of the channel occurs, since it is the only node that is active and the throughput is 0.82 Mbps. At 100s, flow 2 is initiated from node_3 towards node_2. At that instance B_{AVA} for flow 1 is approximately 0.2 Mbps and flow 2 is 0.6 Mbps. At 160s, flow 3 is initiated and contended with node_2. Flow 3 and flow 2 throughputs are reduced to approximately 0.4 Mbps each. But node_0, still contending with node_2, produces the throughput of 0.4 Mbps. It may be deduced that the prediction of instantaneous B_{AVA} must take into account the concept of neighbours' node, where each node with packet ready for transmission always tries to contend for the shared channel. Each node within the transmission range needs to share their instantaneous bandwidth. In order to take into account the shared bandwidth, it is necessary to be able to predict the instantaneous bandwidth which considers the active network. In the next section, an algorithm is developed to predict the instantaneous local bandwidth and simultaneously include the neighbourhood bandwidth.

5.4.3 The Node Bandwidth Estimation By Means Of NAV Duration

5.4.3.1 The NAV Duration

The requirements to support QoS routing in MANETs can only be met through a proper measurement of the parameters concerned. The nature of the wireless channel requires that network and MAC layer interact in order to provide necessary information to support QoS routing. The proposed GA-based route selection largely depends on the information that was extracted from the MAC layer. One of these is the bandwidth. Estimating the available bandwidth for the IEEE 802.11 MAC in MANETs is a challenging task due to the shared medium. During the available bandwidth estimation, we have to consider the activities of the neighbours' nodes. The available bandwidth is estimated, based on the channel status of the medium and computation of the busy time of the shared channel. By using this method, the bandwidth consumed by the activities of the

neighbours, may be considered simultaneously. Any send or receive packets from other nodes will affect the channel status throughout the transmission range.

Utilisation represents a measure of the consumption of an available channel bandwidth. Each node listens to the channel to determine the channel status and computes the busy duration for a period of time $T_{SAMPLING}$. The IEEE 802.11 MAC [44] , utilises both a physical carrier sense and a virtual carrier sense. The basic CSMA/CA transmission is used to determine the two states: busy state (transmitting, receiving and carrier sensing channel) and idle state. Each node will constantly monitor when the channel state changes; it starts counting when channel goes to busy state from idle state and stops counting when channel state changes from busy state to idle state. The busy time, T_{BUSY} is composed of **k** busy instances during an observation interval $T_{SAMPLING}$, defined as,

$$T_{BUSY} = \sum_{i}^{k} t_i \qquad , \qquad \text{(Eqn. 5-2)}$$

which is the total amount of time within the $T_{SAMPLING}$ where the channel is busy. It is a summation of channel occupancy time of the i-th transmits-receive activities that occupy the channel. The node adds all the busy instances to compute the total busy time. The busy time ratio, is calculated for each period of time $T_{SAMPLING}$ as the instantaneous utilisation of wireless channel, and are defined as,

$$U_{INST} = {T_{BUSY}}/{T_{SAMPLING}} \times 100\% \qquad \text{(Eqn. 5-3)}$$

The average utilisation is then estimated using a weighted moving average, as,

$$U_{AVG}(j) = U_{AVG}(j-1) \times \alpha + (1-\alpha) \times U_{INS}(j) \qquad \text{(Eqn. 5-4)}$$

where α is the smoothing factor, between 0 and 1. The local bandwidth consumption is then given as,

$$B_{CON}(j) = U_{AVG}(j) \times TMT \qquad \text{(Eqn. 5-5)}$$

and local bandwidth available as,

$$B_{AVA}(j) = (1 - U_{AVA}(j)) \times TMT \qquad \text{(Eqn. 5-6)}$$

After the node finishes computing the available bandwidth during a period of time $T_{SAMPLING}$ at the MAC layer, it sends the information of the available bandwidth to the network layer and starts computing the available bandwidth during this period. The timing diagram for a basic IEE802.11 DCF function is shown in Figure 5.1. NAV can be used to monitor the channel as either busy or idle. NAV is a timer that indicates the amount of time the medium will be reserved. Nodes set the NAV to the time for which they expect to use the medium, including any frames necessary to complete the current operation. Other nodes count down from the NAV to 0. When NAV is non-zero, the sensing function indicates that the medium is busy; when NAV reaches 0, the sensing function indicates that the medium is idle. A node will copy the other nodes NAV when these other nodes actively occupy the medium. This provides an indication that the neighbour node is busy.

5.4.3.2 Opnet Modeler Implementation of Bandwidth Estimation

Opnet implement MAC layer as a **wlan_mac** process model (**Chapter 3**). It implemented all the specifications for the IEEE802.11 MAC. The standard **wlan_mac** was modified by adding extra functions, to implement the bandwidth estimation using the NAV duration facility. The model was designed such that it can update and interpret the NAV variables as follows.

1. NAV is represented by variable **nav_duration** in **wlan_mac** process model. It is measured in absolute time from the beginning of the simulation.

2. Each node has its own **nav_duration** with the initial value of zero.

3. The **nav_duration** represents the length of time the medium is likely to remain busy.

4. Unit of time of **nav_duration** is seconds.

5. The **nav_duration** is updated based on the channel occupancy.

6. The **nav_duration** in the nodes other than sending nodes, is updated using the *duration* field in a received control frame and proceeds as follows:

    nav_duration = max (**nav_duration**, duration value received in a control frame).

7. If **nav_duration** $= T_{CURRENT}$, it means no pre-announced transmission is going on in the near future. $T_{CURRENT}$ is defined as the instantaneous simulation time.

8. Idle medium is defined as absence of carrier AND **nav_duration** = $T_{CURRENT}$.

9. NAV is set to one, if **nav_duration** is set to the absolute simulation time at the point where the medium busy is expected to end.

10. NAV is reset to zero, if at any point in time, **nav_duration** is set to current time. Logically it is shown in the Figure 5.8 and **nav_duration** updates are shown in Figure 5.9.

Set NAV = 1	Reset NAV=0
medium busy	medium idle
nav_duration > current_time	nav_duration = current_time

Figure 5-8 Logical Setup of NAV

nav_duration updates that causes the setting of NAV=1
nav_duration $= T_{CURRENT} + T_{SIFS} + T_{ACK}$.
nav_duration $= T_{CURRENT} + 2(T_{SIFS} + T_{ACK}) + T_{SIFS} + T_{DATA}$.
nav_duration $= T_{CURRENT} + 2T_{SIFS} + T_{ACK} + T_{DATA}$.
nav_duration $= T_{CURRENT} + T_{SIFS} + T_{DATA}$.
nav_duration $= T_{CURRENT}$ + contention_free_period_length.
nav_duration = recv_idle_time $+ T_{EIFS} - T_{DIFS}$.
nav_duration $= T_{CURRENT}$ + received nav_duration.
nav_duration = next beacon transmission time. nav_duration = recv_idle_time.

nav_duration updates that causes the resetting of NAV = 0
nav_duration $= T_{CURRENT}$

Figure 5.9 The Updates of nav_duration that cause the set and reset of NAV

T_{BUSY} can be calculated by an algorithm monitoring the setting and resetting of NAV. Figure 5.10 shows the point in the *wlan_mac* process model where monitoring of set and reset of **nav_duration** takes place. Algorithm 5.1 calculates the channel busy time. It is executed during the **nav_duration** set or reset process in the *wlan_mac* and occurs according to Figure 5.10. Algorithm 5.2 performs the sampling of the busy time, according to a set value of $T_{SAMPLING}$.
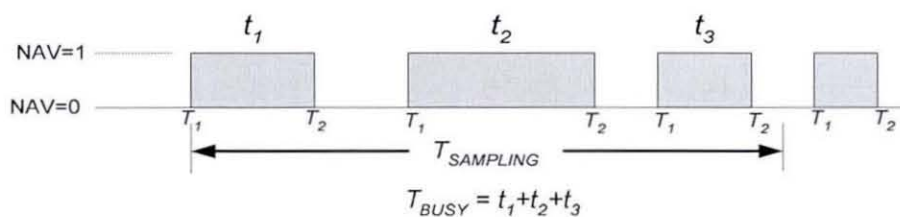
Figure 5.10 The Occupancy of Wireless Medium

Algorithm 5.1 : Calculation of Busy Time

procedure calculate_busy_time;

T_1 : time transition NAV=0 to NAV=1;

T_2 : time transition NAV=1 to NAV=0;

$T_{CURRENT}$: current time;

T_{BUSY} : busy time;

```
01    begin
02      if (nav_duration > T_CURRENT) then
03      if (nav_on_off=0) then
04      nav_on_off=1;
05      T_1 = T_CURRENT;
06      elseif (nav_on_off=1) then exit
07      endif
08      else
09      if (nav_duration=T_CURRENT) then
10      if (nav_on_off=1) then
11      nav_on_off=0;
12      T_2 = T_CURRENT;
13      T_BUSY = T_BUSY + (T_2 - T_1);
14      elseif (nav_on_off=0) then exit;
15      endif;
16      endif;
17      endif;
18      end:
```

Algorithm 5.2: Calculation of Bandwidth Consumed

procedure sample_busy_time

$T_{CURRENT}$: current_time;

$T_{SAMPLING}$: time to sample nav_on_off duration;

T_1 : time transition NAV=0 to NAV=1;

$U_{AVERAGE}$: average utilisation;

$B_{CONSUMED}$: the amount of bandwidth occupy by this node;

TMT : theoretical maximum throughput;

$U_{INST} = 0$, $B_{CON} = 0$;

```
00 begin;
01      if (nav_on_off=1) then
02      if ( T₁ ≤ (T_CURRENT − T_SAMPLING) ) then
03      U_INST =1;
04      else
```

$$05 \quad U_{INST} = \frac{1}{T_{SAMPLING}} * \left(T_{BUSY} + (T_{SAMPLING} - T_1)\right) ;$$

```
06      endif;
07      T₁ = T_CURRENT;
08      elseif (nav_on_off=0) then
```

$$09 \quad U_{INST} = \frac{T_{BUSY}}{T_{SAMPLING}} ;$$

```
09      endif;
10      U_AVG(j) = U_AVG(j−1) * α + (1−α) * U_INST ;
11      B_CON = U_AVG(j) * TMT;
12      T_BUSY = 0;
13      reschedule next interrupt;
14 end;
```

115

5.4.3.3 Bandwidth Measurement Performance

In order to demonstrate the ability to determine the bandwidth consumed using busy time measurement technique, network simulations were performed. The setup consists of 5 source nodes (mn0, mn2, mn4, mn6, mn8), 5 receiver nodes (mn1, mn3, mn5, mn7, mn9) and a single node (mn10) which measures the medium utilisation as a result of interaction between these nodes. The source nodes were located in the middle of a circle of radius 250 m, set by the **RxGroupConfig**, which represent the maximum transmission range of the nodes, as shown in Figure 5.11. The transmission is fixed as a source-destination pair. The measurement node is not the source or the destination of any CBR traffic. Simulation is done with an aggregate traffic of 0 kbps to 2000 kbps. The transmitter node emits a series of 512 bytes packet at constant rate set to give traffic 100 kbps through to 2000 kbps. For each offered load, the simulation runs for 30 seconds, with packet transmission starts at 10 seconds. The data rate is set at 2 Mbps with basic CSMA/CA.
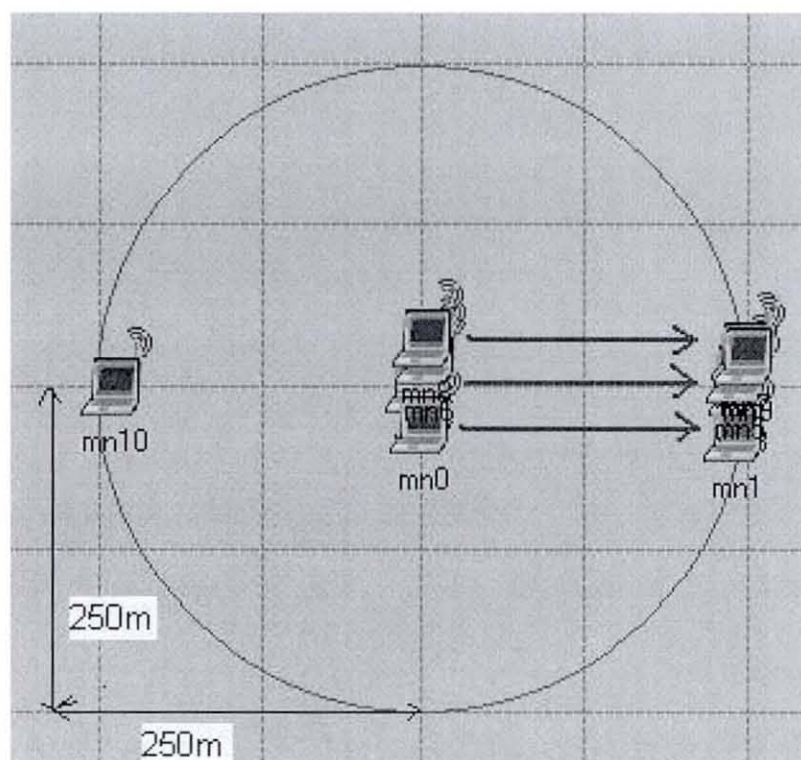
Figure 5-11 Network Configuration for Bandwidth Measurement

The busy time method provides utilisation which varies from zero (fully idle) to one (fully busy). In this scenario, the theoretical maximum throughput (TMT) [45] is set at 1343 kbps. The raw data of the network utilisation for the traffic from 100 kbps to 2000 kbps is shown in Figure 5.12(a)–5.12(p). An average value of the utilisation is estimated from the graph, and then a graph of medium utilisation versus offered load is produced. The graph shows a remarkable increase in utilisation as the offered load is increased. After 1000 kbps the utilisation dropped. With any measurement technique, it is common that instantaneous values vary, sometimes widely as can be seen in the graph of Figure 5.12. The sampling time, $T_{SAMPLING}$ is chosen so as to reduce this variation. We chose $T_{SAMPLING}$ as 20 ms, which is the same value set for beacon transmission rate in WLAN MAC. A graph shown in Figure 5.13 is of the estimated available bandwidth versus offered load and shows the amount of available instantaneous bandwidth at the node. At any instance the value represents node state parameter and will be stored in the node state cache.

In this section, the channel busy time calculation was shown to offer a good measure of network utilisation. It takes into account the contention effect among the neighbouring nodes in a shared medium. It gives an instantaneous measurement and hence can easily be used as a QoS parameter controlling the packet flows across the MANET. In this approach the stored bandwidth available were extracted from the node state cache and used in the fitness calculation of the QoS route selection using the Genetic Algorithm.
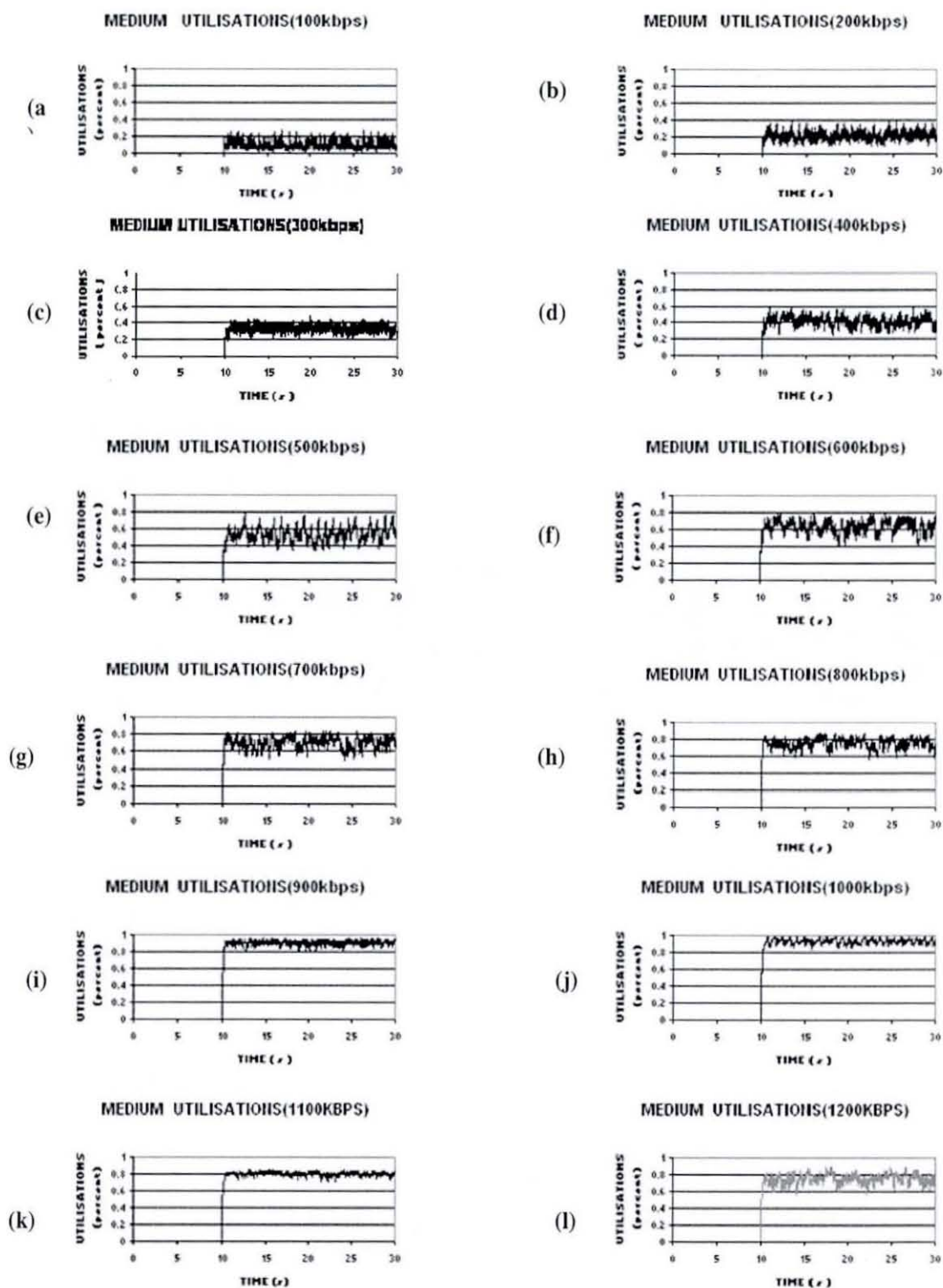
Figure 5.12 Graph of Percentage Utilisation Measured Using Algorithm 5.1 and 5.2

MEDIUM UTILISATIONS(1400kbps)

MEDIUM UTILISATIONS(1600kbps)

(m)

(n)

MEDIUM UTILISATIONS(1800kbps)

MEDIUM UTILISATIONS(2000kbps)

(o)

(p)
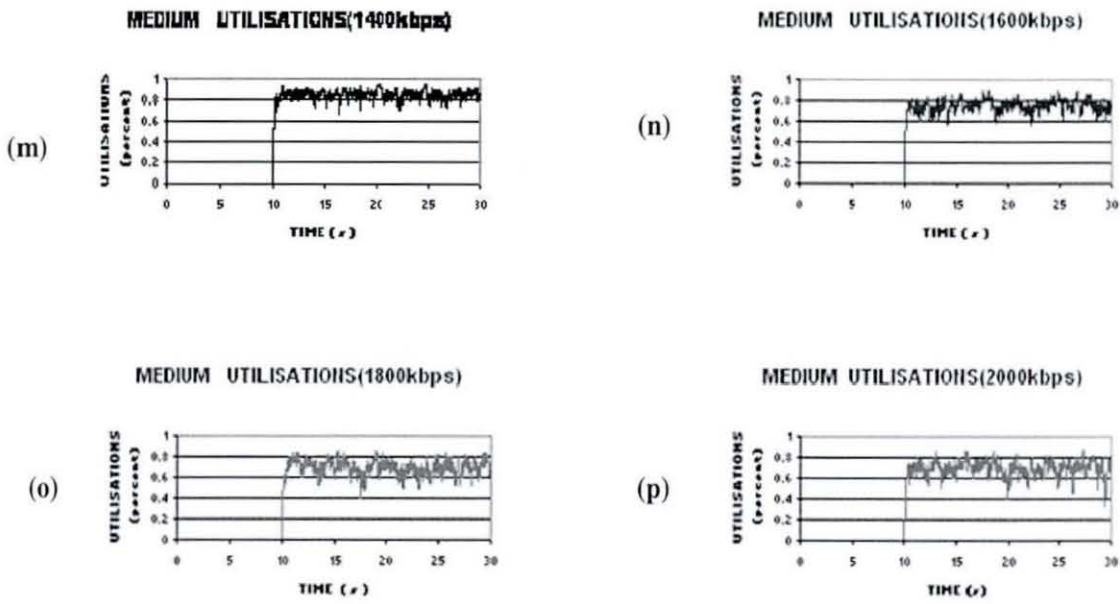
Figure 5.12 (cont) Graph of Percentage Utilisation Measured Using Algorithm 5.1 and 5.2

MEDIUM UTILISATIONS VERSUS OFFERED LOAD
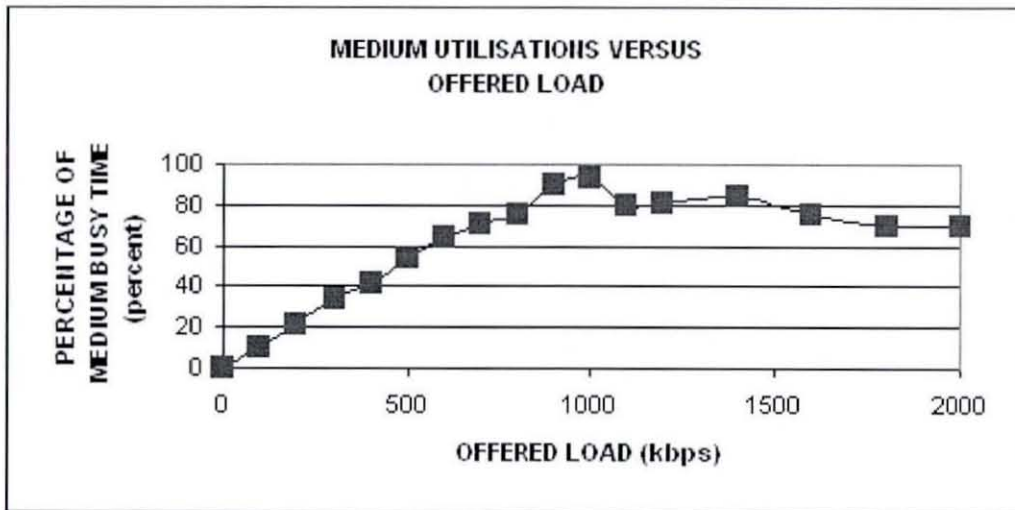
Figure 5.13 Graph of Average Percentage Utilisation

119

Figure 5.14 Graph of the Available Bandwidth for the Offered Load

5.5   The Delay Measurement

While throughput-sensitive realtime flows, such as on-demand multimedia retrieval or video/audio broadcasting, require only throughput guarantees, delay-sensitive realtime flows, such as video/audio teleconferencing, require both throughput and end-to-end delay guarantees. Many studies have focused on providing delay and throughput guarantees in ad hoc networks based on IEEE 802.11. In this approach, the packet delay was measured on a per-hop basis as a packet travelled from one node to the next. The delay measured is cached in the node with respect of where the packet comes from. It is saved in the node state cache, for future usage by the genetic algorithm during the route selection process.

The delay requirement of an application is typically stated in terms of end-to-end packet delay. Table 5.4 shows the delay requirements for the multimedia transmission. End-to-end packet delay is the aggregation of the delays from each hop of the flow. For this algorithm to satisfy effectively the end-to-end delay requirement, the packet delay at each hop must be known and then be able to select the appropriate route which give the required delay. The summation of per-hop delay measurement from source to destination must be

120

less than the end-to-end delay requirement of a flow. However, the per-hop delay at a node is composed of multiple components

Per hop delay component at node **i**, d_i , is composed of three components: the queuing delay, the contention delay at the MAC layer and the transmission delay. The queueing delay, d_{qi} is the interval between the time that the packet arrives at node **i** and the time that the packet becomes the head of line (HOL) packet in node **i**'s queue. The contention delay, d_{ci}, is the interval between the time that the packet becomes the HOL packet and the time that the packet actually starts to be transmitted on the physical medium. This contention delay is unique for contention-based channel access schemes, CSMA/CA. It captures the fact that when a packet becomes the HOL packet at node **i**, node **i** may need to backoff before transmitting the packet on the physical medium.

Table 5.4 Multimedia Services: QoS Requirements

Multimedia Services Key Characteristics	Voice	Internet	VoD	Video Conferencing
Delay	<250 ms	<100 ms	<100 ms	<500 ms
Jitter	<100 ms	<50 ms	< 50 ms	<100 ms
BER	10E-4	10E-6	10E-4	10E-4
Bandwidth	8-16 kbps	500 kbps	10 Mbps	64 kbps-1.544 Mbps

The measured delay is obtained by a node state monitoring algorithm which captured the time stamped of a packet at the source node or at an intermediate node last one hop. The time difference ($T_R - T_s$) is the measured delay, as shown from Figure 5.15.
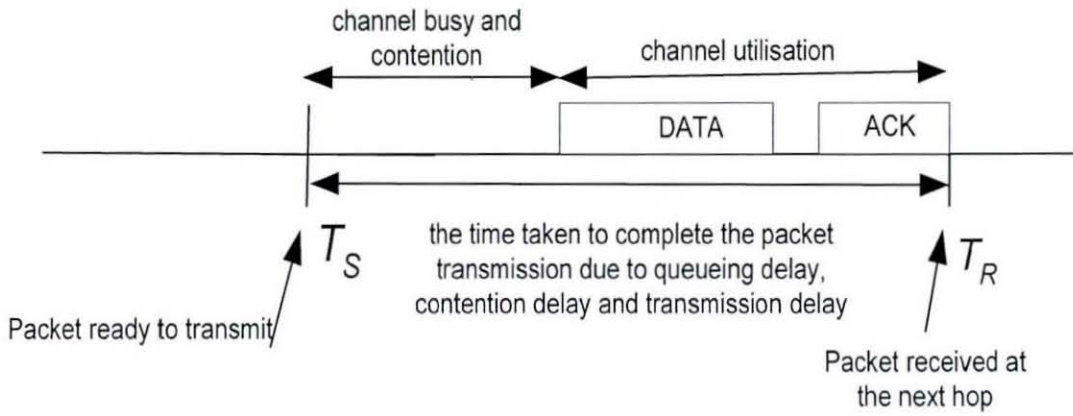
Figure 5-15 Channel Activities For IEEE802.11 For DCF Basic Access

5.6 Implementation of Node State Monitoring Protocol

5.6.1 Updating, disseminating and accumulation of QoS Parameters

One of the functions of the Non-Disjoint Multiple Route Discovery (NDMRD) protocol is to ensure that the QoS parameters that were measured during the monitoring operation of MANET were cached. Figure 5-16 shows the updating of QOS parameters. When any packet type **RREQ, RREP, DATA, RERR** and **CONN** arrived at a node from its neighbour, it triggers the Node State Monitoring protocol whereby the following instantaneous information is obtained: (1) time of packet arrival; (2) time of packet transmit; (3) power measurement due to the arrival of a packet; (4) this node address; (5) the neighbour node address, from where the packet arrived; and (6) the number of neighbours.

All these information are actually the Node States that were listed and defined in Table 5.1. The bandwidth consumed (B_{CON}), bandwidth available (B_{AVA}) and medium access delay (D_{MAC}) for each node is calculated at a regular interval in the wireless LAN MAC and saved in the Node State cache in each node. From the measured Node State, other Node States were then derived as follows: (1) end-to-end delay; (2) normalised velocity; (3) node-pair contraction metric; (4) node-pair expansion metric; and (5) node connectivity index.

These parameters were saved in the Node State table. From this set of parameters, Node Connectivity Index, end to end delay, bandwidth available

122

and medium access delay are resolved as the QoS parameters. It is then copied to the other nodes along the reverse route of **RREQ** by piggybacking on the **RREP** packet. Only the **RREP** packet is used to extract the Node State from each node traversed by **RREP** and piggybacking it towards the source node.
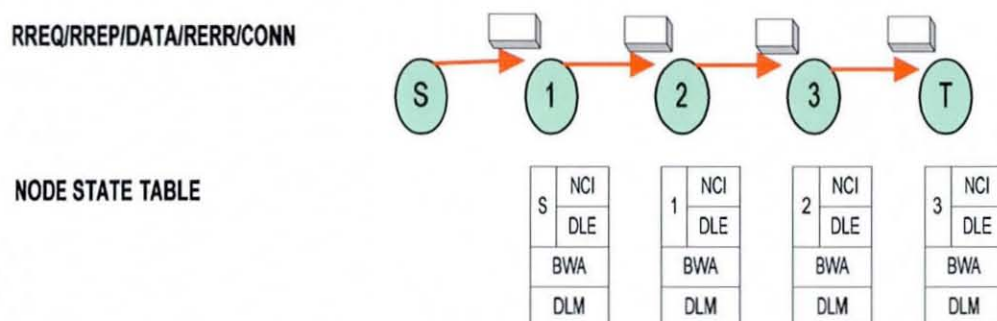
Figure 5.16 Updating Of Qos Parameters Triggered By Packet Arrival
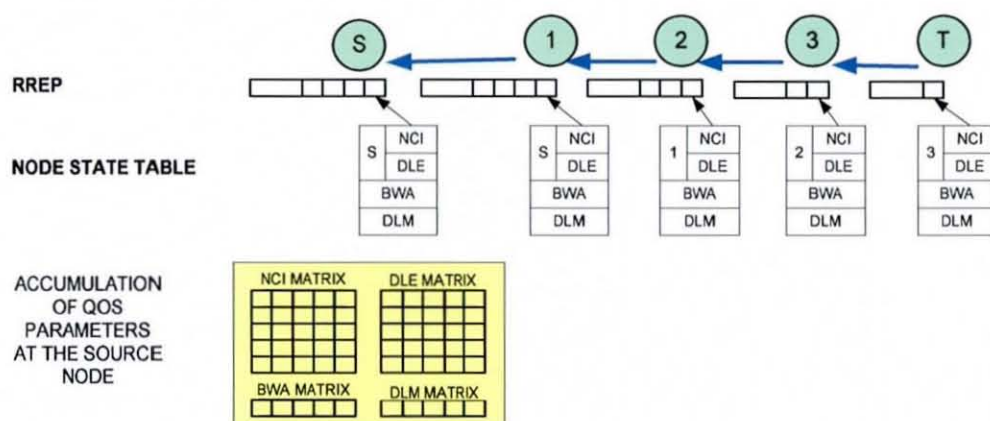
Figure 5.17 Collecting and Disseminating of QoS parameters

Figure 5.17 shows how the **RREP** packet from destination traversed back on the reversed route to the source. For each traversed address, it collects all QoS parameters from the Node State Table. Each time **RREP** reach a node, *nci*, *dle*, *bwa* and *dlm* are being inserted into the RREP packet.

5.6.2 Flowchart of Node State Monitoring Protocol

The flowchart of Figure 5.18 illustrates in detail the Node State monitoring protocol. The protocol initially checks and removes the stale Node State Table entry. The information is considered stale if it resided in the Node State Table more than 50 ms or the existing neighbour node is out of range. It then identifies the packet type and extracts all the relevant Node State information. The protocol identifies the relative movement of the node, either expansion or contraction. With this information the node connectivity index is calculated.

5.7 Summary

In this chapter, the usage of Node State has been proposed. Node State consists of a collection of attributes which defined the particular node at a particular instant in time. Its usage and usefulness is deliberated. The QoS parameters, Delay, Bandwidth and nci are the most important elements of Node State. The *nci* value infer mobility and connectivity of a node and its neighbour. The bandwidth and delay infer the resources being utilised. Next the description of the process of monitoring and capturing of node state information were given in detail. The two most fundamental metrics for QoS routing, that is, the bandwidth and delay, are given special interest. The estimation of node bandwidth is done using the NAV component of the 802.11 MAC protocol and is discussed thoroughly. It has shown tha,t at any instant, the algorithm rightly estimates the node bandwidth. The implementation of the monitoring process with the flowchart is described. The benefit of node state monitoring is that it could extract, process and disseminate the node state information for calculating QoS routing.

NODE STATE MONITORING

Begin

Check the Node State Cache. Delete the expired entry. The entries were expired if (a) the neighbour node is out of transmission range, (b) the last time information were accesed from the cache is more than 50ms.

Access the content of the incoming IP datagram. Get THIS node address. Extract the QOSRGA packet from the IP datagram. Get the option from the QOSRGA packet.

Get the neighbour address for the following packet type received:
1. Route Request Pkt (RREQ) 2. Route Reply Pkt (RREP)
3. Route Error Pkt (RERR)   4. Data Pkt (DATA)
5. Connectivity Pkt (CONN)

Access the content of IP datagram. Get THIS node address

Get the following information:
1. time when the current packet received, T2.
2. power received due to current packet, P2.
3. medium access delay,
4. end to end delay.

Is the Node State Cache empty ?

YES → Initiate new Node State Cache, with all the node state obtained for this neighbour.

End

NO

Get the neighbour address due to current packet from the Node State Cache

Is the neighbour addr exist in the Node State Cache?

NO → Save the new neighbour in the Node State Cache. Defer calculation of *nci* to the next packet arrived from this neighbour.

End

YES

Extract from the Node State Cache of this node,
1. time when previous packet was received, T1.
2. power received due to previous packet, P1.

Is P1>P2?

NO

YES

calculate
$$aem = ((1/\sqrt{P2}) - (1/\sqrt{P1})) * (1/(T2 - T1))$$
$$nci = 0.25 - (1/((8 \times 10^5 - aem)/10^5))$$

Is P2>P1?

NO → Set *nci*=0

YES

Calculate
$$acm = (\sqrt{(P2 - P1)/(P2 * P1)})/(T2 - T1)$$
$$nci = (1/((8 \times 10^5 + acm)/10^5))$$

Save *ALL* node states information obtain here in the Node State Cache

End

Figure 5-18 Flowchart of Node State Monitoring

CHAPTER 6
QOS ROUTE SELECTION USING GENETIC ALGORITHM

6.1   Introduction

This chapter presents in detail the proposed GA route selection approach to QoS routing problem in the MANET system. The proposed GA technique which is based on source routing, effectively selects the most viable route in terms of bandwidth availability, end-to-end delay, media access delay and the sum of *nci*. The Non-Disjoint Multiple Routes Discovery algorithm initially determined a number of potential routes. The returning RREP packets extract the QoS parameters from each node along the routes. GA then operates on this set of routes as an initial population and the corresponding set of QoS parameters.

This chapter is organised as follows. Section 6.2 describes the network model for our MANET environment with QoS routing function. Section 6.3 introduces the steps involved in building GA based QoS route selection mechanism. Section

6.4 describes how the mechanism for searching the best routes in MANET were implemented. It was implemented online, doing the route calculation as needed. Section 6.5 describes the parametric evaluations in order to choose the proper value for the GA control parameters P_m, P_c and the population size. Section 6.6 describes the performance of QOSRGA and Section 6.7 concludes the chapter.

6.2 The Network Model

The underlying topology of mobile ad hoc networks is modeled as a graph G = (E, Q { *nci*, B_{AVA}, D_{E2E}, D_{MAC} }) where E is the set of all mobile nodes, and Q is a set of QoS parameters that determine the viable QoS connectivity between the nodes. Each mobile node *i* $\in$ E has a unique identity and moves arbitrarily. A radius *R* defines a coverage area within which every node can communicate with each other directly. Neighbours of node *i* are defined as a set of nodes *i* # *j*, which are within radius *R* and reachable directly from the node *i*. Every pair of neighbours can communicate with each other in both directions. Hence, there exists a connectivity between neighbours *i* and *j* with the index of *nci*. This connectivity constraint may appear and disappear in the *nci* matrix due to node mobility. A route *P* from source, *s* to destination, *t* is defined as a sequence of intermediate nodes, such that *P(s, t)* = {*s*,... *i, j, k, l, ...t*} without loop. The connectivity constraint, *nci*$_{(i,j)}$ associated with the node pair is transmission cost. It is specified by the connectivity matrix C=[*nci*$_{(i,j)}$] , where *nci*$_{(i,j)}$ represents the node pair connectivity index. It is described as follows,

$$
C = \begin{bmatrix} nci_{0,0} & \cdots & nci_{o,k-1} \\ \vdots & \ddots & \vdots \\ nci_{k-1,0} & \cdots & nci_{k-1,k-1} \end{bmatrix}
$$

(Eqn. 6-1)

The connectivity matrix is built at the source, upon receiving the RREP packets from the destination after a certain period of time of Route Accumulation Latency of $nci_{(i,j)}$. The values continue to change as the topology changes. The protocol also removes the outdated values to ensure that the contents remained current. A connection indicator $L_{i,j}$ mapped the connected nodes forming a chromosome. L_{ij} provides the information on whether the link from node i to node j is included in the routing path. It is defined as follows,

$$L_{i,j} = \begin{cases} 1 & if \ there \ exist \ connectivity \ (i,j) \ . \\ 0 & if \quad otherwise. \end{cases}$$ (Eqn. 6-2)

The diagonal elements of L must always be zero. Another formulation in describing the MANET topology is node sequence in the routes, such that,

$$N_k = \begin{cases} 1, & if \ node \ N_k \in \ route. \\ 0, & \text{............} \ if \ otherwise. \end{cases}$$ (Eqn. 6-3)

Using the above definitions, MANET QoS routing can be formulated as a combinatorial optimisation problem minimising the objective function. The sum of nci of the selected route should be minimum, since this would be the most preferred route due to the higher probability of being connected longer with next hop neighbours. Then, the formulation statement is to minimise the sum of node connectivity index of the route,

$$C_{sum \ (S,T)} = \sum_{i=S}^{T} \sum_{\substack{j=S \\ j \neq i}}^{T} C_{ij} \cdot L_{ij}$$ (Eqn. 6-4)

The sum of nci of the route $P(s, t)$ constitutes the "cost" of the packet transmission process. In this approach, the "cost" of transmission is due to the lifetime of the node pair connection. The longer the connectivity lifetime, the

lower the "cost" of the route is. The node pair connectivity index indicates the estimated length of time a given node pair was in connection. The most important features of *nci* are the velocity and position of a node with respect to the other neighbour node. A node has longer connectivity time if its *nci* is smaller than the other node pair. Detailed descriptions and analysis of node pair connectivity index, *nci* was described in **Chapter 4**.

The operation of GA will minimise the sum of node connectivity index of the route, $C_{sum(S,T)}$, subject to the following constraints:

(i) that it must avoid looping. This constraint ensures that the computed result is indeed an existing path and without loops between a source, **S** and a designated destination, **T** such that,

$$\sum_{\substack{j=S \\ j\neq i}}^{T} L_{i,j} - \sum_{\substack{j=S \\ j\neq i}}^{T} L_{j,i} = \begin{cases} 1 & if & i=S \\ -1 & if & i=T \\ 0 & otherwise. \end{cases}$$ (Eqn. 6-5)

(ii) That the packet transmission can accommodate the available node bandwidth. This constraint ensures that the node bandwidth can manage the request bandwidth such that,

$$B_{AVA,i} \geq B_{REQ}$$ (Eqn. 6-6)

and for the whole route,

$$B_{REQ} \leq min\ (B_S,\ ...,\ B_i, B_j,\ ...B_T\ )$$ (Eqn. 6-7)

where B_{REQ} is the bandwidth of the transmitted message. The node bandwidth must be greater than the demand bandwidth. Generally, for QoS operation to be effective, the bandwidth available for the node in question must be considered. Since the shared medium is being dealt with, CSMA/CA, as the link layer of the mobile ad hoc network, the problem of medium contention among the nodes within the transmission range must be taken into account. Hence, it

is necessary to estimate the instantaneous $B_{AVA,i}$ and $B_{CON,i}$ for the node concerned. The mechanism for estimating the node bandwidth was outlined in **Chapter 5**.

(iii) Constraints in terms of link delay and node delay.

$$D_w \geq \{ \sum_{i=1}^{m} \sum_{\substack{j=1 \\ j\neq 1}}^{|S\rightarrow T|} D_{i,j} \cdot L_{i,j} + \sum_{i=1}^{|S\rightarrow T|} D_j . N_j \} \qquad \text{(Eqn. 6-8)}$$

If several routes exist, then the total delay for a route to be selected is the one that is the least. The mechanisms of calculating the node delay and link delay were outlined in **Chapter 5**.

6.3 Outline of GA-based QoS Route Selection Algorithm

6.3.1 Goal of Qos Routing

The goal of QoS routing is to find a feasible path through the network between the source and destination that possessed the necessary resources to meet the QoS constraints. The notion of feasible routes are summarised according to the following statements:

- The minimum bandwidth for the particular QoS requirement is satisfied by every node throughout the route.

- The end-to-end delay requirement is met. The route with the least delay is chosen.

- The route chosen must be such that it has the longest node connectivity time and measured according to the **nci**, the node pair connectivity index. Then the route with its sum of **nci** is the least is chosen due to the fact that it is highly probable that it has the longest node connectivity time.

In the following subsection, the route selection procedure, network model, coding scheme and GA operators were explained.

130

6.3.2 The general outline of route selection procedure

The outline of the GA based route selection procedures are as follows:

Step 1: A route from source to destination is represented as variable length chromosomes and the nodes within the route as genes. In this procedure the initial routes were not generated randomly, but obtained from the result of Non-Disjoint Multiple Route Discovery (NDMRP) protocol, as outlined in **Chapter 3**. By using this approach, most of the infeasible or bad chromosomes have been eliminated.

Step 2: Generate an initial population of chromosomes of size k, from an input matrix R of routes from source to destination. Matrix R is of dimension [$m \times k$]. In this case m is the population size with even value and k is the number of columns. For each chromosome, the length n, differs, such that $0 \le n \le k$.

Step 3: From the matrix R, a node connection matrix L , and node connectivity matrix C, is generated. Node connection matrix L , provides instantaneous information regarding the state of the network whether it is connected, a '1' or unconnected, a '0'. Node connectivity matrix C, consists of the node pair connectivity index nci, which indicates the state of node connectivity. The algorithm for calculating nci was described in Chapter Four. The other two matrices generated are the bandwidth matrix B, and delay matrix D.

Step 5: Route validation and loop free check ensures that the chromosome that is infeasible is removed.

Step 6: Evaluate the fitness of each chromosome using the fitness functions described below. The fitness function depends on the nci, the end-to-end delay, node delay and node bandwidth.

Step 7: Operate the GA operator on matrix R. The operators are the selection function, the crossover function and the mutation function.

Additionally, the crossover and mutation function depends on the crossover rate, P_c and mutation rate, P_m.

Step 8: Repeat for a number of generations or when the solution converged. Typically, the generations are limited up to twenty. Figure 6-1 shows a flow diagram illustrating the working model of genetic algorithm.
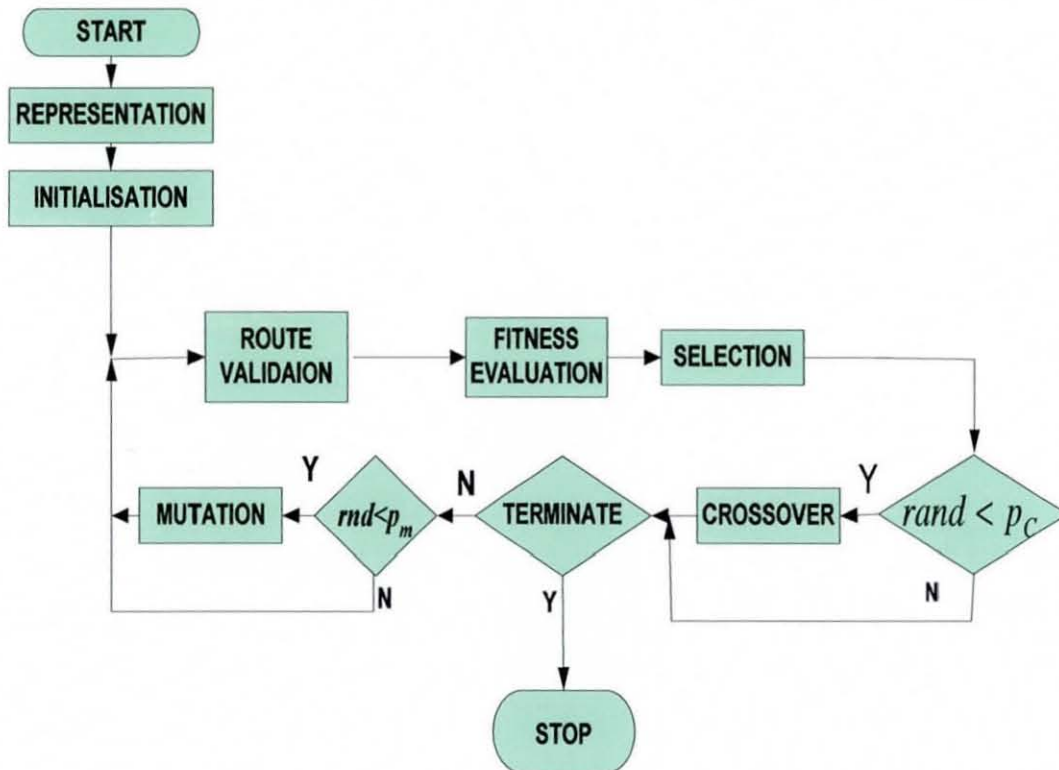
Figure 6-1 Flow Diagram of GA Process

6.4 Implementation

6.4.1 Chromosome Representation

The chromosome consists of sequences of positive integers, which represent the identity of nodes through which a route passes. Each locus of the chromosome represents an order or position of a node in a route. The gene of the first and the last locus is always reserved for the source node, S and destination node, T respectively. The length of the chromosome is variable, but it should not exceed the maximum length which is equal to the total number of

nodes in the network [27] . It is unlikely that more genes are needed than the total number of nodes to form a route. A chromosome which represents the route encodes the problem by listing up node identity from its source node to its destination node based on node information monitored from the network. The information can be obtained and managed in real-time by MANET monitoring algorithm and the non-disjoint multiple routes discovery protocols which have being described in **Chapter 3.**
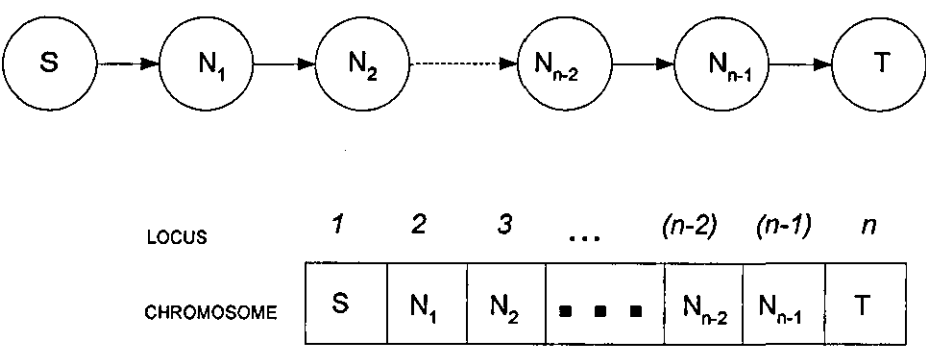
Figure 6-2 Chromosome Representation of S-T Route

An example of route chromosome encoding from node **S** to node **T** is shown in Figure 6.2. The chromosome is essentially a list of nodes along the constructed path, **S** to **T**. In Figure 6.2, *n* represents the total number of nodes forming a path. The gene of the first locus encodes the source node, and the gene of second locus is selected from the nodes connected with the source node. A chosen node is always checked to prevent a node from being selected twice in the same chromosome, thereby avoiding loops in the route. This process continues until the destination node is reached. It is noted that an encoding is possible only if each step of a route passes through a physical link in the network.

6.4.2 Limited Population Initialisation

GA process typically starts with a large number of initial populations. A large number of population results in better chances of getting good solutions. Generally, the initial populations were obtained by generating the chromosome randomly. In QOSRGA, the initial populations were gathered as a result of NDMRD protocol. In a MANET system, with 5 nodes, the possible number of

solutions were calculated as 10 according to the formula $n(n-1)/2$ [18]. One approach is to generate the initial solutions randomly and then remove the invalid solutions before being fed to the GA module. Furthermore, the infeasible solutions can only be eliminated after the connectivity matrix is obtained by the multiple route discovery algorithms. The number of possible solutions increased enormously as the network gets bigger, as shown in Table 6.1. Another approach is to produce an initial population by extracting the existing potential solutions from the result of NDMRD protocol. Clearly, a set of useful solutions are extracted before being processed by the GA module. This set of solutions has the characteristics of non-disjoint multiple routes, whereby each chromosome starts with the source node and ends with the target node. No looping is possible as it was done by the multiple routes discovery procedure and each intermediate node must pass the two node connectivity test.

Table 6.1 An Example of Search Space for Different Network Sizes

No Of Nodes	7	10	15	20
Links, L=n(n-1)/2	21	45	105	190
Search Space, 2^L	2.10×10^6	3.51×10^{13}	4.05×10^{31}	1.56×10^{57}

6.4.3 Fitness Calculation

Fitness calculation is most crucial in the GA operation, whereby the best route can be identified. In this case the least value of fitness constitutes the lowest cost and the one that is to be chosen. The fitness value of routes is based on various QoS parameters: bandwidth, node delay, end to end delay and the node connectivity index, nci. Clearly it can be classified as multiple-objectives optimisation problem. According to M. Gen $et\ al.$ [18] , each objective function can be assigned a weight and then the weighted objectives combined into a single objective function. For this MANET QoS routing protocol, the weighted-sum approach can be represented as follows. The fitness function operates to minimise the weighted-sum F, which is given as,

$$F = \alpha.F_1 + \beta.F_2 + \gamma.F_3 ,$$

(Eqn. 6-9)

where F_1, F_2 and F_3 are the objective functions which describe *nci*, delay and bandwidth respectively. F_1, F_2 and F_3 are given as follows,

(a)
$$F_1 = \sum^{|s \to t|} C_{ij} \cdot L_{ij} ,$$

(Eqn. 6-10)

(b)
$$F_2 = \left(\sum_{j=1}^{|S \to t|} D_{ij} \cdot L_{ij} + \sum^{|s \to t|} d_j \cdot N_j \right)$$

(Eqn. 6-11)

(c)
$$F_3 = \begin{cases} 1/B_i & if \quad B_i - B_{QOS} > 0 \\ 1000 & if \quad B_i - B_{QOS} \le 0 \end{cases}.$$

(Eqn. 6-12)

The weights α, β and γ are interpreted as the relative emphasis of one objective as compared to the others. The values of α, β and γ are then chosen to increase the selection pressure on any of the three objective functions. The fitness function [18] measures the quality and the performance of a specific node state. The fitness function includes and correctly represents all or at least the most important parameters that affect QoS Routing. Having described these parameters, which are the bandwidth, *nci*, medium access delay and end to end delay, the next issue is the decision on the importance of each parameter on the QoS Routing protocol as a whole. The significance of each parameter is defined by setting appropriate weighting coefficients to α, β and γ in the fitness function that will be minimised by the GA operations. The values of these coefficients (Table 6.2) were determined based on their equal importance towards the overall QoS Routing performance.

Table 6.2 Weighting Coefficients of GA Fitness Function

Weighting Coefficients	Equal Importance Values
α	10^{-3}
β	10^{-4}
γ	10^{-3}

Concerning the function which involved bandwidth, we need to find the minimum bandwidth among the nodes and compare this with the demand bandwidth, B_{QOS}. If the minimum bandwidth is less than the B_{QOS}, the fitness is set to a high value so that in the selection process it will be eliminated. By doing so, all the nodes where the bandwidth is limited, will have been eliminated simultaneously, the total delay being more than the typical delay and when the node pair connectivity index is high. Algorithm 6.1 shows the fitness calculations.

Algorithm 6-1: Fitness Calculation

Input:

$$R = \begin{bmatrix} n_{0,0} & n_{0,1} & n_{0,2} & \cdots & n_{0,u-1} \\ n_{1,0} & \cdots & \cdots & \cdots & \cdots \\ n_{2,0} & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ n_{v-1,0} & \cdots & \cdots & \cdots & n_{v-1,u-1} \end{bmatrix} \qquad C = \begin{bmatrix} nci_{0,0} & \cdots & nci_{o,k-1} \\ \vdots & \ddots & \vdots \\ nci_{k-1,0} & \cdots & nci_{k-1,k-1} \end{bmatrix} \qquad D = \begin{bmatrix} D_{0,0} & \cdots & D_{o,k-1} \\ \vdots & \ddots & \vdots \\ D_{k-1,0} & \cdots & D_{k-1,k-1} \end{bmatrix}$$

$$L_{i,j} = \begin{cases} 1 \\ 0 \end{cases} \qquad\qquad BW = [B_0\ B_1\ B_2 \cdots B_{k-1}] \qquad d = [d_0\ d_1\ d_2 \cdots d_{k-1}]$$

00 begin;

01 $F_1 = 0; F_2 = 0;\ F_3 = 0; B = 1.343;$

02 $\alpha = 0.001;\ \beta = 0.0001;\ \gamma = 0.001;$

03 for (i=0 to m)

04   *for (j=0 to | S$\rightarrow$T |)*

05 $F_1 = F_1 + C_{i,j}$; $F_2 = F_2 + D_{i,j} + d_j$; $B = \min((B_{QoS} - B_j.N_j),B)$

06   *endfor*

07   *if (B_{QOS} < B)*

08 $F_3 = 1/B$

09   *else*

10 $F_3 = 1000$

11   *endif*

12 $F[i] = \alpha F_1 + \beta F_2 + \gamma F_3$

13 endfor

14 end

6.4.4 Mobile Nodes Crossover

Crossover examines the current solutions in order to find better ones. Physically, the crossover operation in the QoS routing problem plays the role of exchanging each partial route of two chosen chromosomes in such a manner that the offsprings produced by the crossover represent only one route. This

dictates selection of one-point crossover as a good candidate scheme for the proposed GA. One partial route connects the source node to an intermediate node, and the other partial route connects the intermediate node to the destination node. The crossover between two dominant parents chosen by the selection gives a higher probability of producing offsprings having dominant traits. But the mechanism of the crossover is not the same as that of the conventional one-point crossover. In the proposed scheme, the two chromosomes chosen for crossover should have at least one common gene, except for source and destination nodes. It is not a requirement that they be located at the same locus. That is, the crossover is independent of the node position in routing paths. Figure 6-3 shows an example of the crossover procedure.

Crossing Sites Available: (4,3) and (6,5)
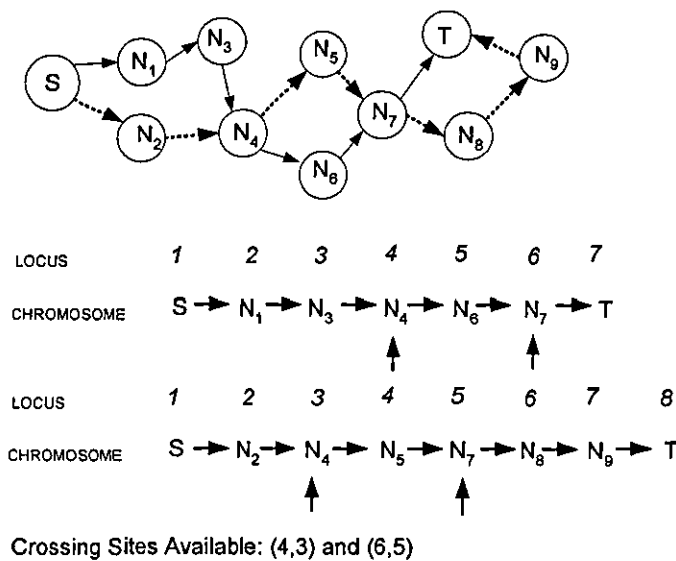
Figure 6-3 Route Crossover

It shows a set of pairs of nodes which are commonly included in the two chosen chromosomes but without positional consistency being first determined. Such pairs are called potential cross sites. Then, one pair (4,3) is randomly chosen and the locus of each node becomes a crossing site of each chromosome. The crossing sites of two chromosomes may be different from each other. Each partial route is exchanged and assembled, eventually leading

to two new routes. It is possible that loops are formed during crossover. A simple restoration procedure is designed in this regard such that it can improve the rate of convergence and the quality of solution. Of course, such chromosomes will gradually be weeded out in the course of a few generations because the traits of those chromosomes drive fitness values from bad to worse. Restoration function can be used to eliminate the infeasible chromosomes.

The procedure for crossover operator:

Step 1: Input a matrix which consists of the route array and the number of nodes in the environment.

$$ROUTE\_ARRAY = \begin{bmatrix} n_{0,0} & n_{0,1} & n_{0,2} & \cdots & n_{0,k-1} \\ n_{1,0} & \cdots & \cdots & \cdots & \cdots \\ n_{2,0} & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ n_{m-1,0} & \cdots & \cdots & \cdots & n_{m-1,k-1} \end{bmatrix} \begin{cases} chromosome\_1 \\ chromosome\_2 \\ \quad \cdots \\ \quad \cdots \\ chromosome\_M \end{cases} \quad \text{(Eqn. 6-13)}$$

Step 2: Test for crossover rate. Initialise the random number generator and the new route array. The population size must be positive and even.

Step 3: Consider a pair of chromosomes denoted as parents, V_1 and V_2, starting from the last chromosome within the population. The length might not be the same as originally stated.

Step 4: Locate the potential pair of crossing sites, by searching for nodes common to both chromosomes.

Step 5: If more than one pair of crossing sites exist, apply a random number to establish one particular pair of crossing sites, i.

Step 6: Perform crossover of V_1 and V_2, by exchanging all nodes after the crossing site i. Two offsprings, V_1' and V_2' were produced.

6.4.5 Restoration Function

The crossover operation may generate infeasible chromosomes that violate the constraints, causing loops to be generated in the routing paths. It must be noted that none of the chromosomes of the initial population or after the

crossover are infeasible because, once a node is chosen, it is excluded from the candidate nodes forming the rest of the path. The restoration method is employed in the proposed GA which eliminates the lethal genes. It thus can cure all the infeasible chromosomes. The proposed restore function is shown in Figure 6.4. It shows how one of the offsprings produced after crossover becomes infeasible because the new route contains the loop N_7, N_6, N_4 and N_5. The restoration function detects the loop by a simple search described below.

Procedure for Restoration function:

Step 1: Consider the resultant pair of chromosomes from the crossover, the children, V_1' and V_2'.

Step 2: Locate the genes that occur more than once within V_1' and V_2', labelled as elimination index.

Step 3: If elimination index is more than one, remove the redundant mode.

Step 4: Repeat until all the nodes are restored.

Algorithm 6-2: Route Crossover

Input: 1) population matrix

$$R[x, y] = \begin{bmatrix} n_{0,0} & n_{0,1} & n_{0,2} & \cdots & n_{0,k-1} \\ n_{1,0} & \cdots & \cdots & \cdots & \cdots \\ n_{2,0} & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ n_{m-1,0} & \cdots & \cdots & \cdots & n_{m-1,k-1} \end{bmatrix}$$

2)p_c = crossover probability
3)m = population length
4)k = total number of nodes
5)parent1, parent2 = two input chromosomes for crossover
6)offspring1, offspring2 = two output chromosomes

```
00  begin;
01  if ( random(0,1) ≥ pc ) exit;
02  Rnew[x,y] ← [0];
03  for (i=0 to m/2)
04      for (j=0 to k)parent1[j] ← R [(2i - 2),j]
05      parent2[j] ← R [(2i - 1),j]
06      length1 ← sizeof(parent1[j]>0)
07      length2 ← sizeof(parent2[j]>0)
08      endfor;
09          if (parent1 ∩ parent2 ≠ 0 )
10              { x0, ...xr, ...xp } ← { parent1 ∩ parent2 };
11              { y0, ...yr, ...yp } ← { parent2 ∩ parent1 };
12              yr, xr ← sizeof({ x0, ...xr, ...xp } ) * random(0,1);
13              offspring1 ← { parent1(s, ..... xr ) ∪ parent2(yr+1 ....... yp) };
14              offspring2 ← { parent2(s, ..... yr ) ∪ parent1(xr+1 .......xp) };
15              if ( offspring1 ∩ offspring2 ≠ 0 )
16                  remove looping genes from offspring1
17                  remove looping genes from offspring2
18              endif;
19              for (j=0 to k)
20                  new_route[2i-2][j] ← offspring1;
21                  new_route[2i-1][j] ← offspring2;
22              endfor;
23          else
24              for (j=0 to k)
25                  new_route[2i-2][j] ← parent1;
26                  new_route[2i-1][j] ← parent2;
27              endfor;
28          endif;
29  endfor;
30  end;
```
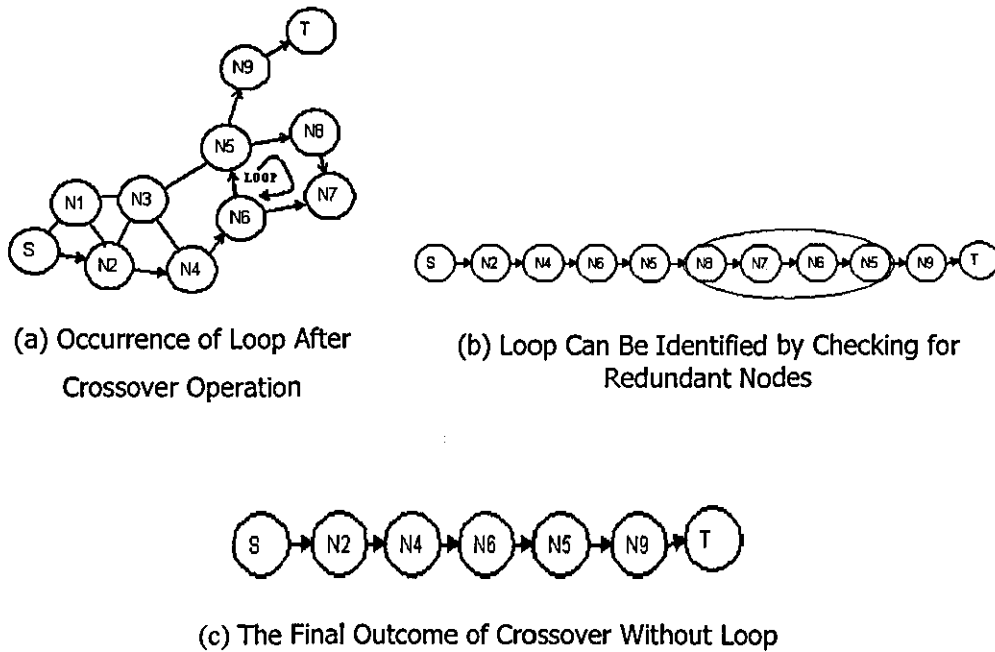
(a) Occurrence of Loop After Crossover Operation

(b) Loop Can Be Identified by Checking for Redundant Nodes

(c) The Final Outcome of Crossover Without Loop

Figure 6-4 Restoration Operation

6.4.6 Route Mutation

Mutation is used to change randomly the value of a number of the genes within the candidate chromosomes. It generates an alternative chromosome from a selected chromosome. It can thus be seen as an operator charged with maintaining the genetic diversity of the population, thereby keeping away from local optima. Mutation may also induce a subtle bias in which it generates an alternative partial route from the mutation node to the destination node. However, this small bias does not affect the performance of the algorithm. This is explained as follows: (1) mutation leads to an infinitesimal increase in the probability of inducing the bias; and (2) selection and crossover strongly influence the way this bias operates. Indeed, by the process of mutation, harmful effects may vanish altogether. Figure 6.5 shows the procedure of the mutation operation.

The procedure for the mutation process can be outlined below:

Step 1: Input two matrices: (1) the population matrix, Eqn. 6-14 and (2) the connectivity matrix, Eqn. 6-15. The population matrix consists of a collection of

142

chromosomes which result from previous generation where each chromosome represents a QoS route from source to destination.

$$POP\_MATRIX = \begin{bmatrix} n_{0,0} & n_{0,1} & n_{0,2} & \cdots & n_{0,k-1} \\ n_{1,0} & \cdots & \cdots & \cdots & \cdots \\ n_{2,0} & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ n_{m-1,0} & \cdots & \cdots & \cdots & n_{m-1,k-1} \end{bmatrix}$$

(Eqn. 6-14)

$$CONNECTIVITY\ MATRIX,\ L_{i,j} = \begin{bmatrix} l_{1,1} & l_{1,2} & l_{1,3} & \cdots & l_{1,n} \\ l_{2,1} & \cdots & \cdots & \cdots & \cdots \\ l_{3,1} & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ l_{n,1} & \cdots & \cdots & \cdots & l_{n,n} \end{bmatrix}$$

(Eqn. 6-15)

Step 2: Select randomly a parent chromosome V, from the *POP_MATRIX*. It is selected with the probability **P_m**.

Step 3: Randomly select a mutation node i from V, .

Step 4: Generate the first subroute r_1 from source node, S to node i by deleting a set of nodes in the upline nodes after the mutation node.

Step 5: Generate a second subroute r_2 from i to the destination node T. It is done as follows.

Step 5-1) Determine node degrees of i, deg(i), neighbours of i.

If deg(i)=1 and { deg(i) } = T, then terminate the search, since the second subroute consist of T.

If deg(i) =1 and { deg(i) } # T, then terminate the mutation process.

If deg(i) > 1 go to Step 5-2.

Step 5-2) Select node {1, 2, 3, ... deg(i) }. If deg(1)=1 and {deg(1)}=T then second subroute is generated. Proceed with 2 and so on.

If deg(1)=1 and {deg(1)}# T, proceed with 2 and so on.

If deg(1)>1 go to Step 5-3.

143

Step 5-3) Select node { 1, 2, 3, ... deg(1) }. If deg(1)=1 and

{deg(1)}= T then second subroute is generated. Proceed with 2 and so on. If deg(1)=1 and {deg(1)}# T, proceed with 2 and so on.

If deg(1)>1 terminate. We search for the second subroute up to two stages so that the effort will not take much processing time.

Step 5-4) If the number of second subroute generated is more than

one, then choose the least hop.

Step 6: Combine the first subroute and second subroute forming a new route. Add to the *POP_MATRIX*.

Step 7: If any duplication of nodes exists between r_1 and r_2, discard the routes and do not perform mutation. Otherwise, connect the routes to make up a mutated chromosome. However, nodes already included in an upper partial-route should be deleted from the database so as not to include the same node twice in the new route. The upper partial-route represents the surviving portion of the previous route after mutation; it is the partial chromosome stretching from the first gene to the intermediate gene at the mutation point.

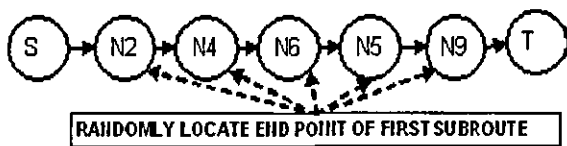Algorithm 6.3 shows the operation of route mutation in detail.

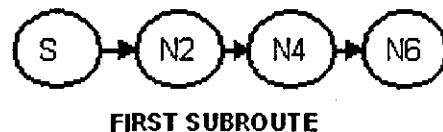Fig.6-5(a) The End Point of First Subroute is Determined Randomly

Fig.6-5(b) An Example of First Subroute
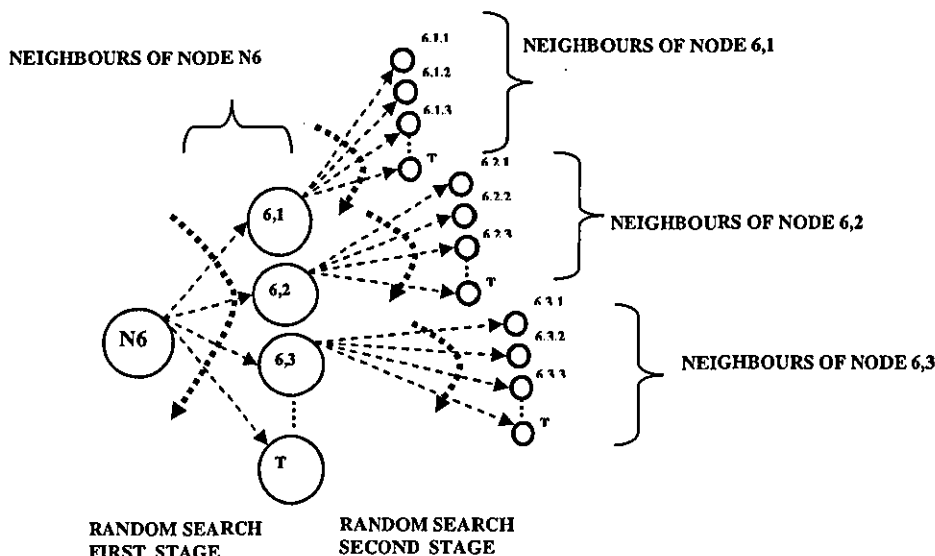
Fig. 6.5(c) The process of obtaining second subroute using a two stage search. The search stops on finding destination node T and repeats the search for another subroute. The search completes after all possible combination of nodes that make up the second subroute are covered. From the list of the second subroutes, the algorithm chooses the least hop.

Fig.6-5(d) Combining first subroute and second subroute.

Fig.6-5(e) New route obtained after mutation process.

Figure 6-5 Route Mutation Operation

145

Algorithm 6-3: Mutation

Input: 1) population matrix   3) chromosome

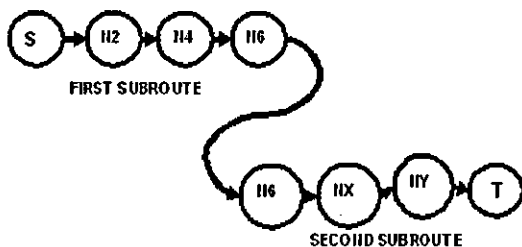$$R = \left(C_0, C_1, C_2, \cdots, C_{m-1} \right) \qquad C_j = \left(g_0, g_1, g_2, \cdots, g_{k-1} \right);$$

2) connectivity matrix   4) g_m = mutation node

                                  5) p_m = mutation rate

$$L_{xy} = \begin{bmatrix} l_{0,0} \cdots & l_{0,k-1} \\ l_{k-1,0} \cdots & l_{k-1,k-1} \end{bmatrix}$$

6) m=population length

7) M = mutation chromosome

```
00   begin;
01   for ( i = 0 to m)
02       [C i] = random   (R[ C0 , C1 , C2 , ··· , C m-1 ]
03       if ( random( R[ Ci ] ≤ pm ) then M = R[ Ci ]
04           endif
05   endfor
06   Subroute1, S1 =[ g0 , g1 , ... , g m ]
07       for ( i = 0 to deg(gm) )
08           if  node(i) # T
09               for (j=0 to deg(node(i)) )
10                   if node(j) # T
11                       for (k=0 to deg(node(j))
12                           if node(k)#T  , break;
13                           generate Subroute2;
14                       endfor;
15                   endif;
16                   generate Subroute2
17               endfor;
18               generate Subroute2
19           endif;
20       endfor;
21   Subroute2, S2[0] = [ gm, gm+1, gm+2,..., T ]
             o  - - - - - -
             S2[X] = [ gm, gm+1, gm+2,..., T ]
22   Select S2 the least hop.
23   new_route = S1+S2
24   remove redundant node from new_route
```

6.4.7 Route Selection Schemes

There are a few selection schemes to be considered. They are the tournament, elitism, stochastic universal selection and roulette wheel selection. Each has its own merit and can be examined to find the most useful for an application. The schemes are briefly described and in the next section we examine and chose one of them.

6.4.7.1 Tournament Selection

In tournament selection, a number of individuals are chosen randomly from the population and the best individual from this group is selected as a parent. This process is repeated as often as individuals are to be chosen. These selected parents produce uniform random offspring. The tournament selection method is shown in Algorithm 6.4.

6.4.7.2 Roulette Wheel Selection

The simplest selection scheme is roulette-wheel selection. It is a stochastic algorithm and involves the following technique. The individuals are mapped to continuous segments of a wheel, such that each individual's segment is proportional in size to its fitness. A random number is generated and the individual whose segment spans the random number is selected. The process is repeated until the desired number of individuals is obtained. Selection is random but biased towards individuals with least cost or higher fitness.

6.4.7.3 Elitism

In elitism, the individuals that are the fittest fraction μ of the population are retained for the next generation. The remaining fraction of $(1 - \mu)$ of the population is selected for crossover using the tournament selection.

6.4.7.4 Stochastic Universal Selection

A perceived drawback of the roulette wheel selection is its high degree of variance. An unfit individual could by chance, reproduce more times than a fitter one. The mechanics of stochastic universal selection overcomes this effect. Individuals are allocated a proportion of a circumference of a wheel according to their fitness value. Only one spin is required in order to select all

147

reproducing individuals and so the method is less computationally demanding compared to RWS [42].

Algorithm 6-4: Tournament selection

Input: Population Matrix   Fitness Function Matrix   m=population length

$$R = \begin{bmatrix} n_{0,0} & n_{0,1} & n_{0,2} & \cdots & n_{0,k} \\ n_{1,0} & \cdots & \cdots & \cdots & \cdots \\ n_{2,0} & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ n_{m-1,0} & \cdots & \cdots & \cdots & n_{m-1,k-1} \end{bmatrix} \qquad F = \begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_{m-1} \end{bmatrix}$$

00   begin;

01   for i=0 to m, selection_index=floor(m*random());

02   for i=0 to m

03   index1=m-i;

04   index2=selection_index[i];

05   if (F [index1] ≤ F [index2])

06   new_R[i, *] = R[index1, *];

07   else

08   new_R[I, *] = R[index2, *];

09   endif;

10   endfor;

148

6.5 GA Parametric Evaluations and Preferences

Selecting genetic algorithm parameters such as population size, mutation rate and crossover rate is a very difficult task. Each combination of parameters may produce a variety of outcomes. Haupt [49] outlined a general procedure for evaluating these parameters, after which reasonably suitable parameters are adopted for the specific application. In this case, four selection methods were considered, namely the roulette wheel selection (RWS), tournament selection (TS), stochastic universal selection (SUS) and elitism technique. Next, the parameters P_c, P_m and population size were considered. It is necessary to examine the performance of each and select according to preferences. Matlab was used to initially design a GA-based routing algorithm without the QoS function. The route selection was based on the shortest path without considering the bandwidth, delay and node connectivity index, *nci*. The cost for each path was randomly generated. The main objective was to examine all the GA parameters that are useful for the protocol design and use them in the design of the QoS route algorithm. Hence, in this section a mobile network was considered consisting of 20 nodes, randomly distributed within a perimeter of 1000m by 1000m. Each node had a transmission range of 250m. In each selection, a method using various population sizes was explored to find the best point of convergence corresponding to the lowest measure such as minimum cost and average minimum cost. For each reading taken, 10 simulation runs had to be done and the results were averaged. Table 6-3 outlines the variables for the optimisation on the GA parameters.

Table 6-3 Variables Used for the Simulation Runs

Selection Methods	RWS, TOUR, SUS, Elitism
Population Sizes	10 to 2000 individuals
Fixed Mutation Rate	0.01 to 0.5
Crossover Rate	0.80 – 1.00
Generations	10 – 10000 generations

6.5.1 Population Size

The effect of population was investigated by fixing the mutation rate (P_m = 0.01) and changing the population size. The simulation was run for 2000 generations.

6.5.1.1 Average Minimum Cost, C_{AMC}

The minimum cost in each generation was recorded and the average minimum cost C_{AMC} was evaluated over the range from 0 until the 2000th generation. Figure 6.6 plots C_{AMC} for the four different selection methods (with μ = 0.05 for Elitism). It shows that in RWS, a population size in excess of 700 produces a significantly low cost. This is because, with a large population, the RWS method finds it easier to choose the low cost individuals. Consequently, the probability of a low cost individual being selected becomes low. Apart from this, with a large population size there are too many sectors within the wheel, making the probability of selecting each sector smaller. The most significant result is that of the tournament selection and elitism. With a population size of approximately 10, it produces very low C_{AMC}. Hence the best choice of selection method would be the tournament selection and elitism. Figure 6-6 is re-plotted in Figure 6-7, concentrating on a population size below 100. The results reinforce the view that a population size below 100 is appropriate for both the Elitism and Tournament selection. In fact, a population as low as 20 could be used and still produce good fitness. The Tournament selection was chosen and the Elitism method was left for future work.

6.5.2 The Fitness Function Compared to the Generations

Another necessary parametric study for the GA process was to find the most reasonable number of generations that could produce a satisfactory result. Again, this is done offline. The mutation rate was fixed at 0.1 and the crossover rate at 0.7. A simulation was run for different sizes of populations. The fitness measures were taken for different population. The result is shown in Figure 6-8. It shows the lowest cost as a function of the number of generation for Tournament selection. This figure also shows that a larger population size generally produces lower cost solutions. However, the rate of convergence is

about the same when the population exceeds 20. The use of a larger population requires greater processing power and therefore is not desirable. From the graph it can also be inferred that convergence could occur in less than 20 generations.
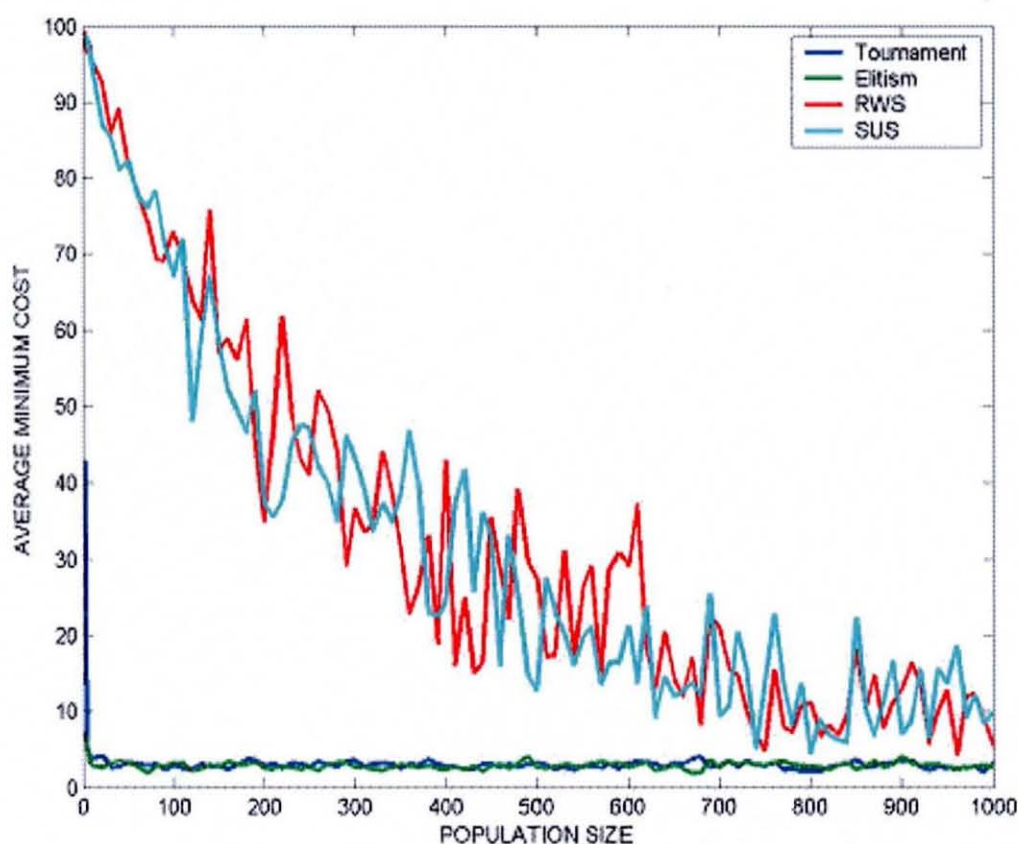

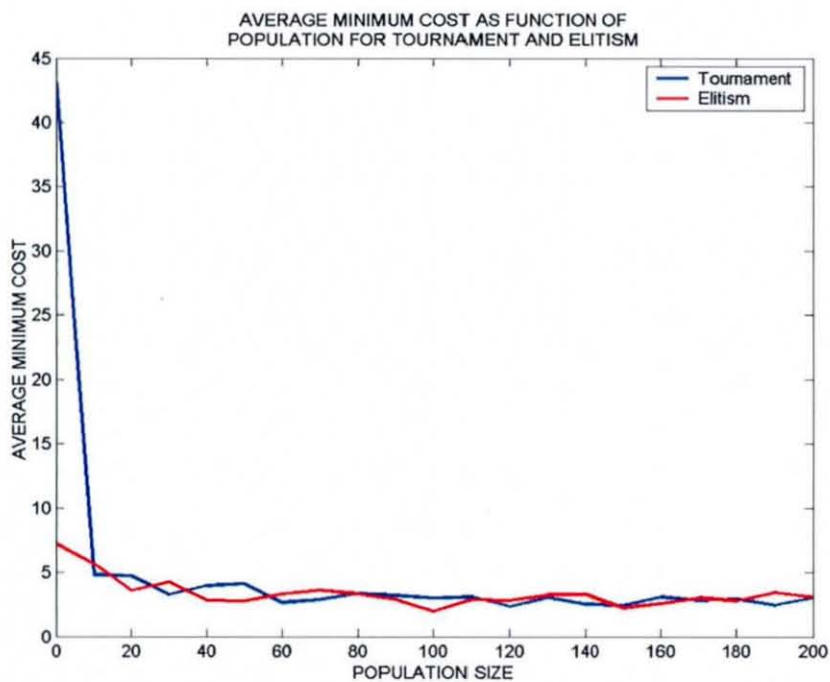Figure 6-6 Plot Of Average Min Cost for Various Selection Methods

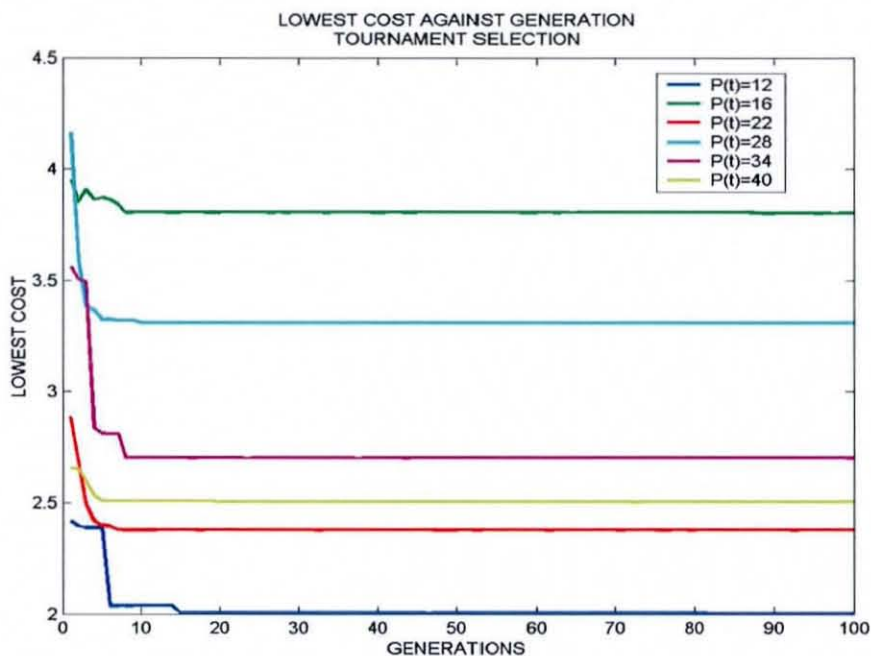Figure 6-7 Plot of C_{AMC} as a Function of Population Size – Tournament and Elitism (0.05)

Figure 6-8 Lowest Cost against Generation for Tournament Selection

6.5.3 Crossover and Mutation Probability

Another set of very important parameters for the GA implementation are the crossover probability P_c and the mutation probability P_m. The parameters determine how many times crossovers occurred and how many times mutations occurred within a transmission period. The occurrence of crossover and mutation increases the convergence rate. De Jong [49] , tested various combinations of GA parameters and concluded that mutation is necessary to restore lost genes, but this should be kept at a low rate, for otherwise the GA degenerates into a random search. Further study by Schaffer *et al.* [50], suggest that the parameters should have these recommended ranges: population size of 20 ~ 30, mutation rate of 0.005 ~ 0.1 and crossover rate of 0.75 ~ 0.95. Another study by Haupt [51] concluded that the best mutation rate for GAs lie between 5% and 20% while the population size should be less than 16. For this thesis, where GA operation is done online, the value of P_c and P_m is taken to be between 0.4 and 0.9 and between 0.05 and 0.2 respectively. The choice of these parameters should produce a reasonably high efficiency packet transmission. The population size is limited up to the number of routes discovered. The limit is also imposed on the number of generations. Haupt [49] provide useful guidelines on when to stop the algorithm. It would be better to stop the algorithm when the value of the selected route does not change or when it reached the maximum number of generations. We designed the algorithm so that it stops when the value of the route does not change and also we restricted the maximum number of generations up to 20.

Simulation experiments were run online by considering a MANET scenario running QOSRGA protocol, with 20 nodes placed within an area of 1000 meter x 1000 meter. Each node has a radio propagation range of 250 meters and channel capacity of 2 Mbps. Up to 10 sources were initiated transmitting CBR with a data payload of 512 bytes. The nodes move randomly with random waypoint mobility model. For each point, 10 simulations run were carried out and for each run it executed for 200 seconds. Two sets of simulation experiments were conducted, one for calculation of crossover probability and the other for mutation probability. The experiment was conducted using the

source traffic rate of 40 kbps, 100 kbps and 900 kbps. The aim of the first set is to identify exactly the possible values of P_c which would give the best results. The metric is the transmission efficiency. It is defined as the ratio of average throughput of all nodes to the average load of all the nodes in the network. We varied P_c but set P_m constant as 0.1. The results are shown in Figure 6.9. For P_c with values from 0.4 to 0.8 the transmission efficiency is more than 80%. For 100 kbps CBR source, the maximum efficiency occurred when P_c is approximately 0.65 and for 40 kbps CBR source it is 0.4. The 900 kbps CBR source, the efficiency does not deviate very much. Hence, as a general guideline the value of P_c as 0.7 was chosen for all future simulation experiments.

For mutation probability, a similar simulation was run, this time the crossover probability was fixed at 0.7 and the mutation probability varies from 0.04 to 0.8. Figure 6.10 shows the result of mutation probability. Mutation probability produces the highest transmission efficiency when it is 0.1, which is more than 80 % for all three different traffic rates. Hence it is concluded that the crossover probability and mutation probability can be taken as 0.7 and 0.1 respectively.

6.6 Performance of QOSRGA

In this section, an extensive performance evaluation is presented through simulations for two important objectives: (1) to determine the behaviours of QOSRGA as CBR sources are varied, and (2) to determine the effect of velocity on QOSRGA for two different CBR sources. The performance of QOSRGA is influenced by the number of routes, number of CBR sources and the respective data rate. The source data rate controls the amount of packets being transmitted from a node. Multiple routes are generated as a result of NDMRD protocol. Subsequently multiple CBR sources could also generate traffic congestion within the network. The metrics measured are the Average Packet Delivery Ratio (APDR), Average Packet Delay (AETED), Average Throughput (ATPUT), Average Number of Hops (ANH) per destination and the Average Routing Load Ratio (ARLR). APDR is defined as the ratio between the average total traffic received and average total traffic sent. AETED is defined as the
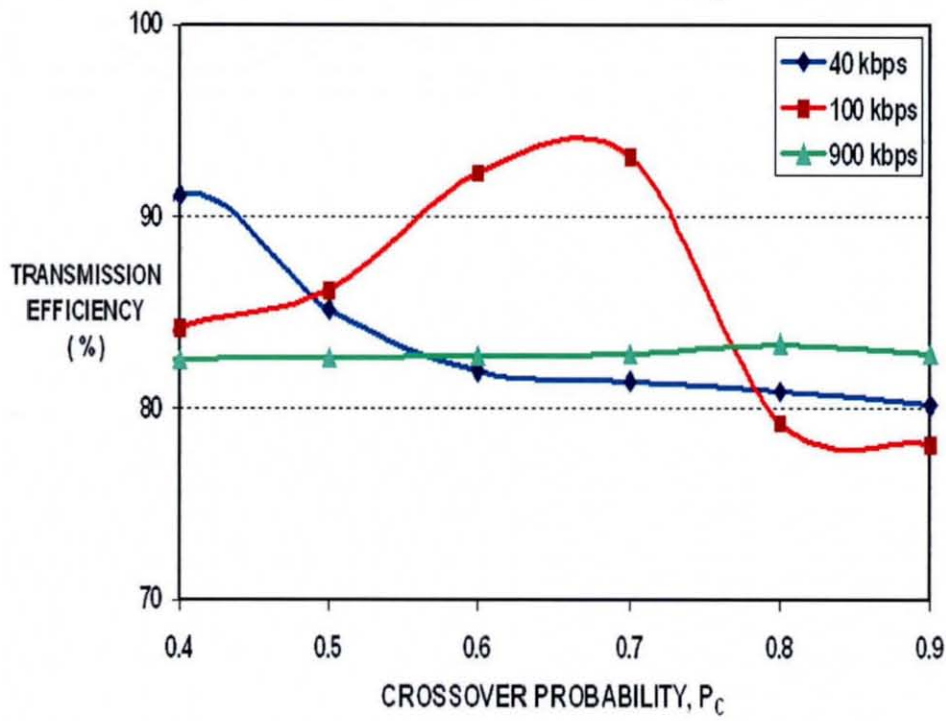
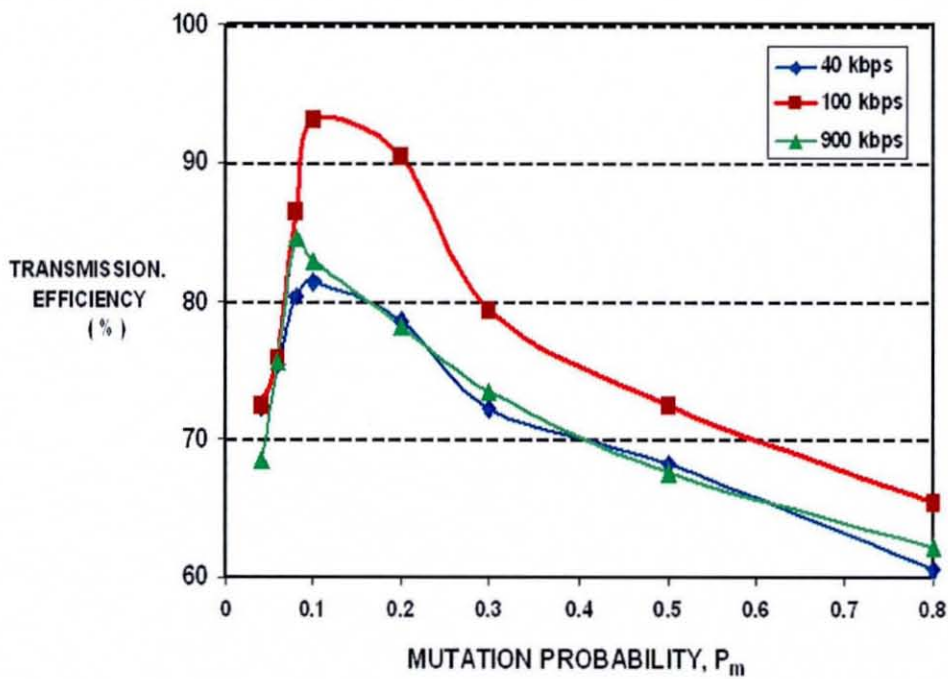Figure 6-9 Transmission Efficiency as a Function of Crossover Probability

Figure 6-10 Transmission Efficiency as a Function of Mutation Probability

average end to end packet delay for the whole network. ATPUT is defined as the average total bit rate transferred to all nodes in the network. ANH is defined as the average number of hops from source to destination. ARLR is defined as the ratio of routing traffic send for every data packet received at the final destinations.

6.6.1 Simulation Parameters

Simulations were conducted using Opnet Modeler [29] that includes the wireless module which was used in prior performance studies [52][53] . The network was made up of 20 mobile nodes in a field configuration of 1,000 m x 1,000 m area. Nodes movement is according to random waypoint model [54] . It is configured into our model through the Mobility Profile Definition. All nodes moved within the boundary of the field configuration. The pause time was kept constant at 1 second [53][54] for all the simulation experiments. This gives the consistency in the nodes' movement for all the scenarios. The start time was set to zero until the end of simulation. The only variation on the RWP model that was made is the maximum velocity. The velocity was configured according to uniform distribution between zero and V_{max}. Traffic sources with 512 bytes data packets [52][53][55][56] were the CBR type and the packet sizes were configured to vary according to exponential distribution. The packet inter arrival rates were also configured to vary according to exponential distribution. Figure 6-11 and Figure 6-12 shows the scenario of the simulation experiment for initial setup and after the simulation ends at 200 seconds. Each reading is a result of averaging 10 simulation runs using a different seed value.
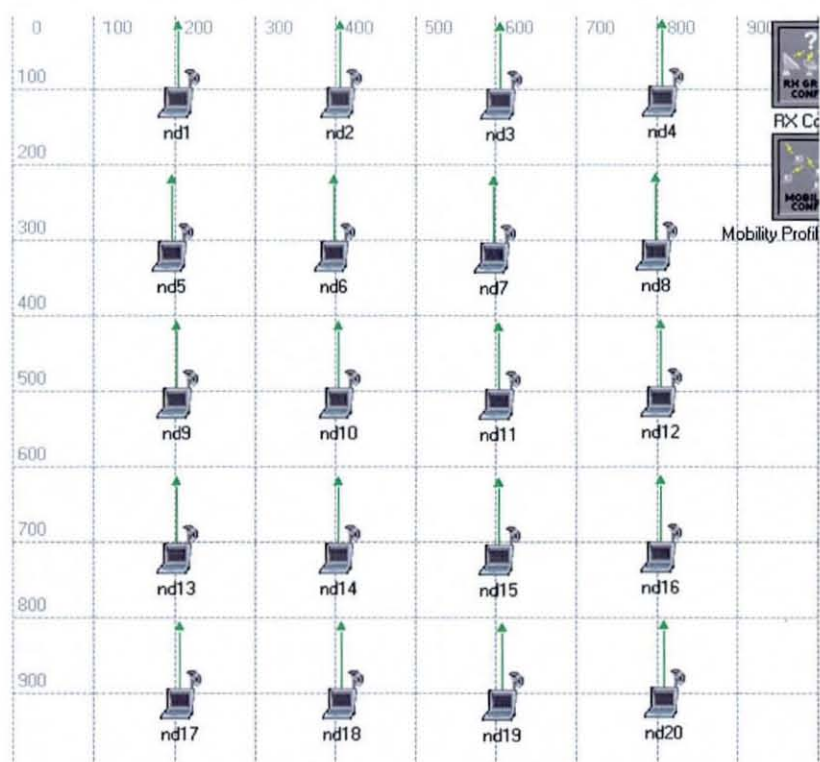
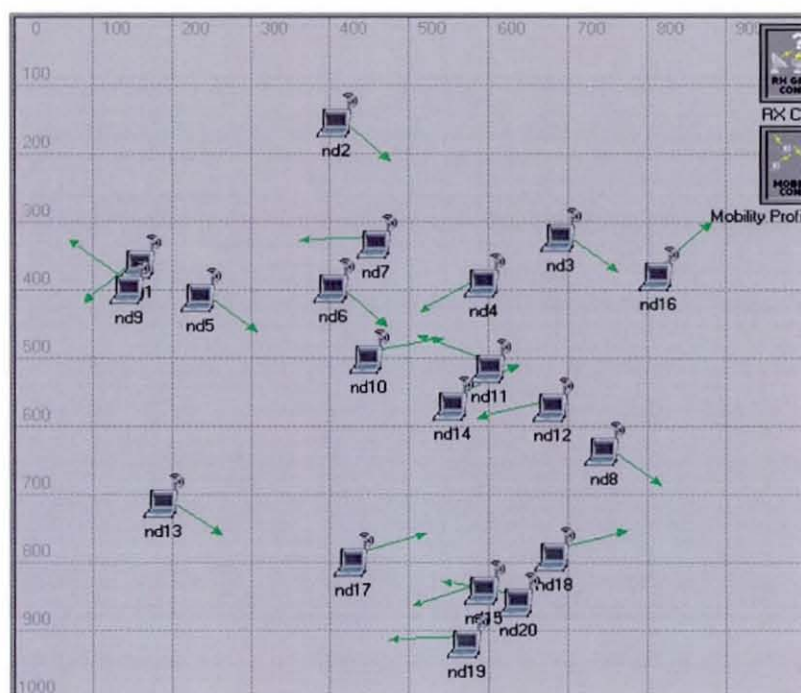Figure 6-11 Scenario of 20 Nodes with RWP Mobility at Time 0 second.

Figure 6-12 Scenario of 20 Nodes with RWP Mobility at Time 200 seconds

157

6.6.2 Effect of the Number of CBR Sources on Performances

The performance of QOSRGA was analysed when the network was congested. The performance metrics were measured for one, three, four and five CBR sources during the simulation period. The effect of congestion on performance was studied by varying the number of data sources. The results of Average Packet Delivery Ratio (APDR), Average Packet Delay, Average Throughput and Average Number of Hops are shown in Figure 6-13, 6-14, 6-15 and 6-16 respectively. When the number of sources was increased, the network becomes loaded with data packets for gaining access to the medium. Also, there will be an increase in probability of control packets that were broadcasted and collided with the unicast packets from each of the destination nodes. In such scenarios, data packets will be dropped and thus limits the data packet received. The Average Packet Delivery Ratio (APDR) was shown in Figure 6-13, and the highest APDR was a single-source scenario, and the least is the five-sources scenario. Furthermore, the delay of data packets was higher when the network was highly congested, as shown in Figure 6-14. The plot in Figure 6-15 illustrates the variation of throughput with time. The throughput measurement was taken as the total number of bits transferred to the destination nodes at physical layer. It shows the usage of the shared channel overwhelmingly heavy for the five-sources scenario, compared to a single-source scenario. Figure 6-16 relates the average number of hops from source to destination for each scenario. It was observed that the hop count does not vary very much during the simulation time. It indicates that QOSRGA maintained a route long enough and possibly, could maintained for the whole sessions.
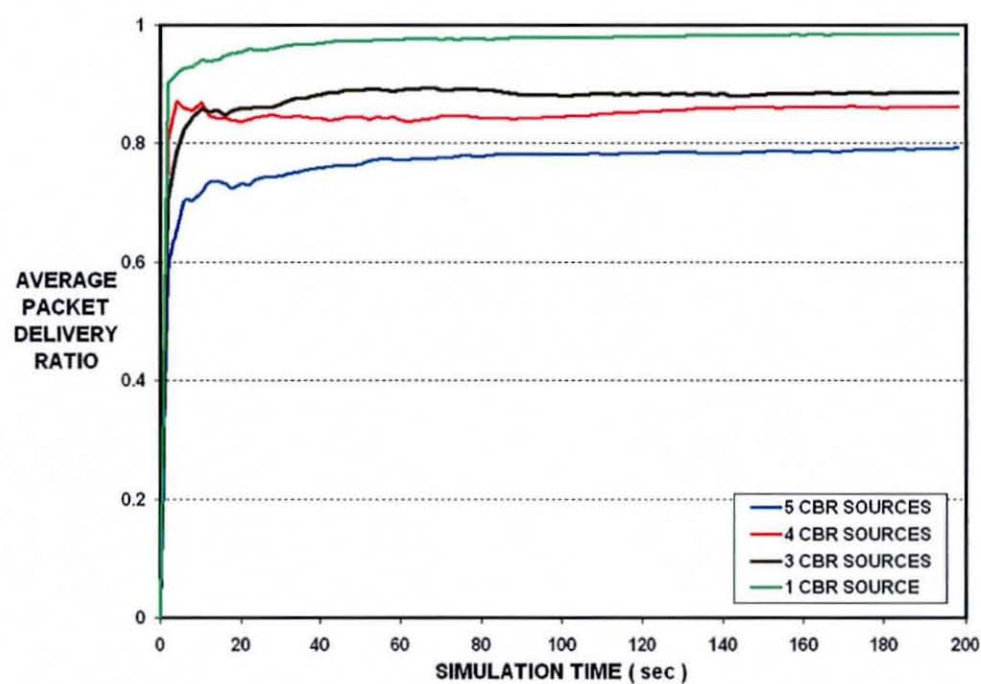
Figure 6-13 Average Packet Delivery Ratio as a Function of Simulation Time
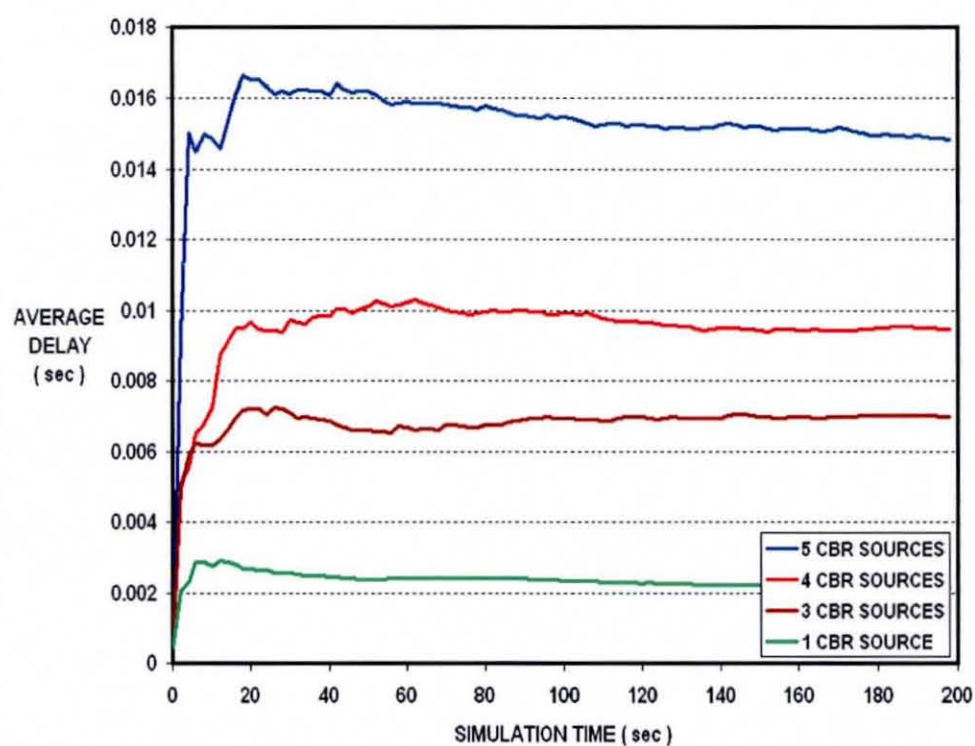

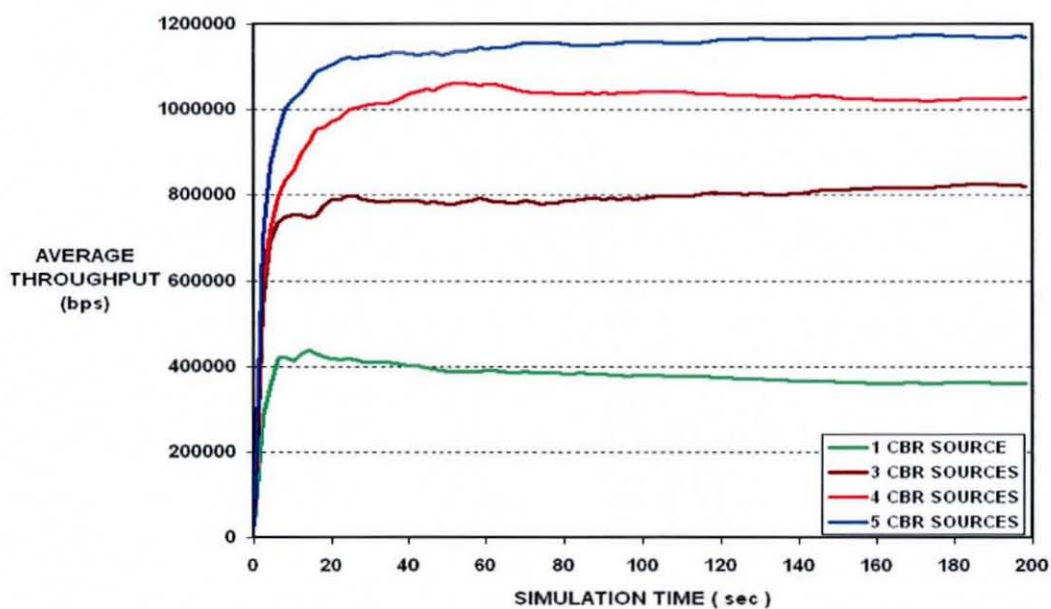Figure 6-14 Average Packet Delay as a Function of Simulation Time

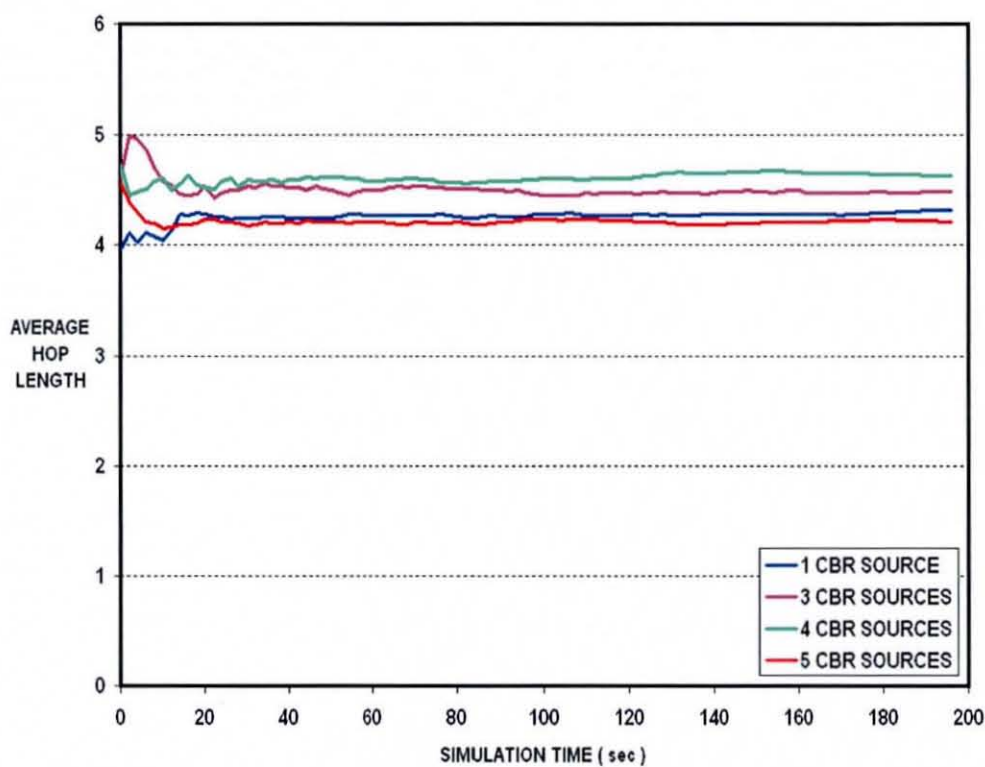Figure 6-15 Average Throughput as a Function of Simulation Time

Figure 6-16 Average Number of Hops for Each Destination

6.6.3 Effect of Maximum Velocity on Performances

Node mobility generally influences the overall performance of the network. The influenced of velocity on the APDR, AETED and ARLR was investigated. The number of CBR sources was set to five. The simulation was ran by varying uniform velocity distribution with mean outcome of V_{max} as 1, 2, 5, 10, 15, 20 and 25 m/s. For stationary nodes, the RWP parameter setting was removed altogether. The source data rates used were 40 kbps and 200 kbps. The aimed of the simulation experiment was to relate the mobility of nodes and its effect on the overall performance of QOSRGA. The results were shown in Figure 6-17 to Figure 6-20.

Figure 6.17 shows the APDR against maximum velocity for 40 kbps and 200 kbps. With the increase in velocity, the 40 kbps remained constant at approximately 82%. The performance of the 200 kbps traffic shows a decreasing trend as the velocity increased. It dropped substantially from 78% to 42% at 5 m/s, then to 19% at 20 m/s and improved a little to 22 % at 25 m/s. QOSRGA performed effectively with low source data rate throughout the range of velocities but not with high source data rate. With higher data rate and faster node movement, more packets are dropped due to short node pair connectivity time, that is, high *nci* value. Figure 6.18 shows the average delay against maximum velocity. With 40 kbps source, the average delay was much less than that with a source of 200 kbps. The 200 kbps source generated more packets and resulted in higher congestion, and hence produced a greater delay. Nevertheless, it was still below the 100 ms, which was the maximum delay allowed for most multimedia services. The reading also shows a constant trend and not an increasing trend. It was due to the delay reading, which was based on all the packets that actually arrived at the destinations, thus not considering the dropped packets.

AVERAGE PACKET DELIVERY RATIO AS A FUNCTION OF MAX VELOCITY
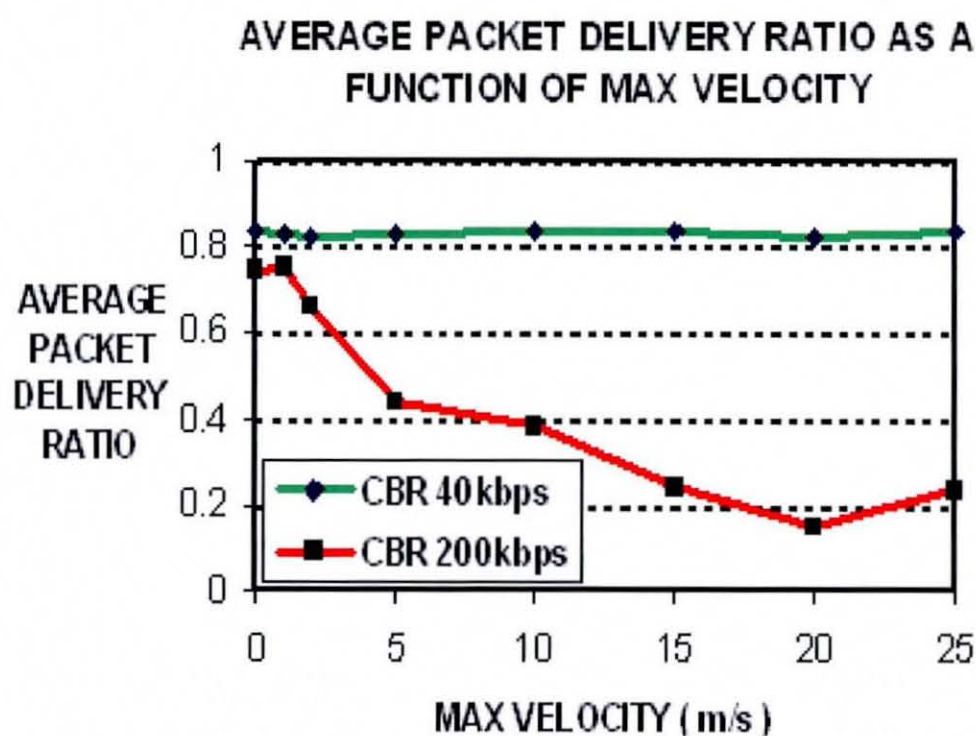
Figure 6-17 Average Packet Delivery Ratio as a Function of Max Velocity

AVERAGE DELAY AS A FUNCTION OF MAX VELOCITY
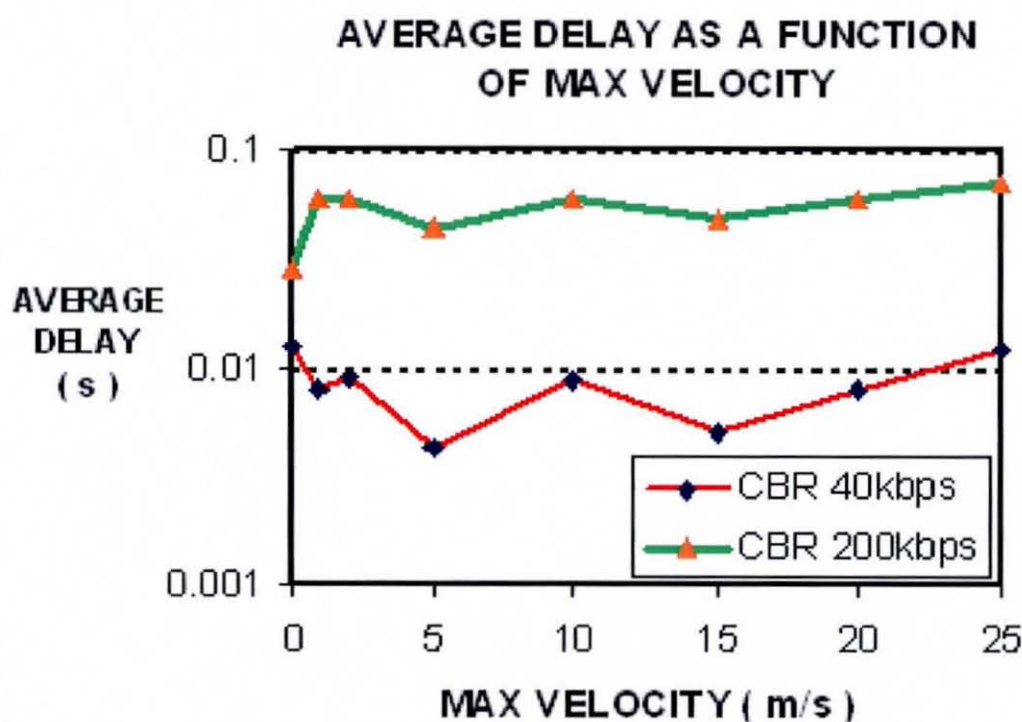

Figure 6-18 Average Packet Delay as a Function of Max Velocity

162

AVERAGE ROUTING LOAD RATIO AS A FUNCTION OF MAX VELOCITY
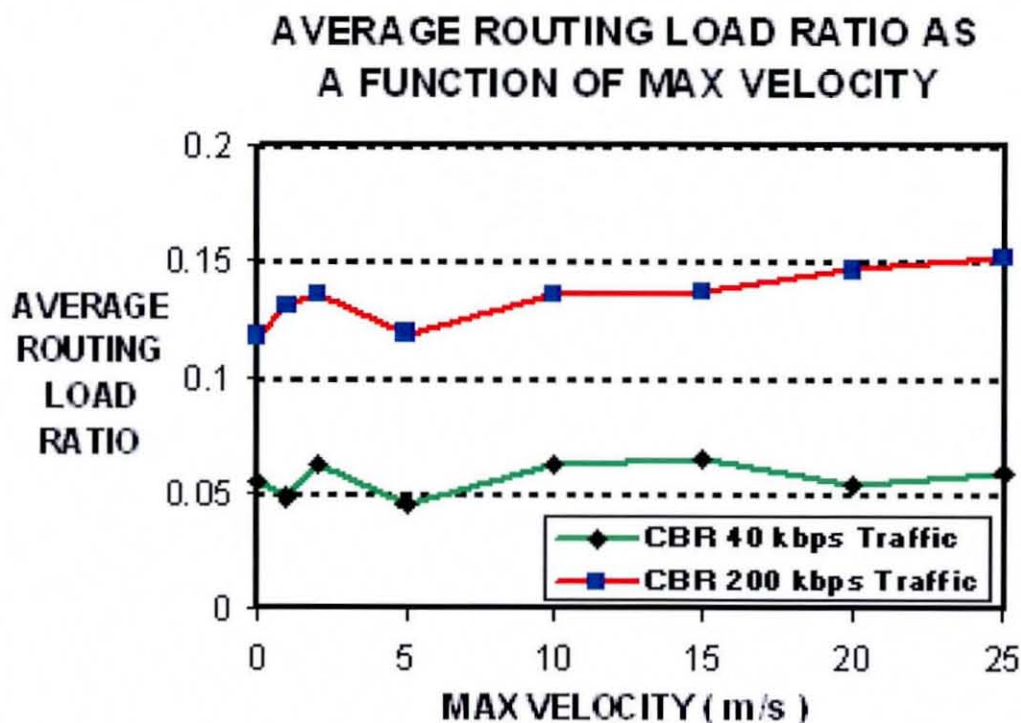

Figure 6-19 Average Routing Load as a Function of Max Velocity

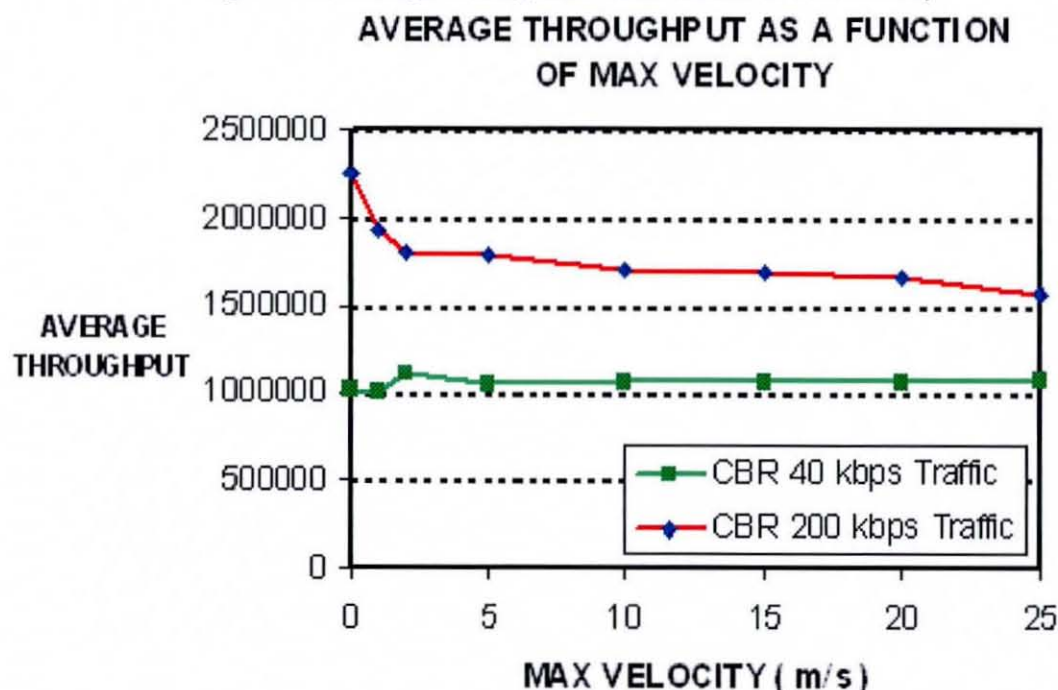# AVERAGE THROUGHPUT AS A FUNCTION OF MAX VELOCITY

Figure 6-20 Average Throughput as a Function of Max Velocity

Figure 6.19 shows the variation of average routing load ratio (ARLR) against maximum velocity. The routing traffic for 200kbps is 100% more than that of 40kbps. Also, as the velocity increases, the ARLR for the 200kbps source shows an incremental trend. The 40kbps source on the other hand remains constant at approximately 0.06%. As the velocity increased, the higher source needs more overhead packet for every successful packet reception at the destination. Figure 6-20 shows the variations of throughput. Once again the CBR 40 kbps source produced a constant throughput for all velocities. The throughput for CBR 200 kbps showed an immediate decline, after which the decline was gradual. At higher velocity, the node pair connectivity time was less, the packet dropped tend to increase, and hence the bit rate transfer to the destination was reduced considerably.

6.7 Summary

A scheme has been presented for QoS route selection in a MANET based on Genetic Algorithm. In the proposed scheme of QoS routing, selection of a route was based on (1) bandwidth and delay requirements, and (2) the longest node pair connectivity time indicated by node connectivity index (*nci*). The route selection algorithm was outlined and implemented. The variable length chromosomes represented the routes and genes represented the nodes. The algorithmic process was initialised by introducing a limited population. Limited population was accumulated during the route discovery by the NDMRD protocol. The fitness calculation was done using the weighted sum approached, combining the entire objective functions into a single objective. The weight of each objective was deduced by inspection. Next, a one-point crossover was implemented, as were a restoration function and a route mutation. To execute crossover and mutation process properly, it was shown that the value of P_c and P_m chosen satisfy the transmission efficiency test. Performance of QOSRGA was done to determine its behaviour due to congestion and increasing velocity. From this simulation experiments, it was learnt that QOSRGA performs better for a 40 kbps CBR traffic rate compared to a 200 kbps CBR traffic rate. Further work could be done to ascertain the range of source traffic which can be applied. In the next

chapter further evaluation of the performance of QOSRGA was done by comparing it with other existing routing protocols.

CHAPTER 7
PERFORMANCE EVALUATION OF QOSRGA

7.1 Introduction

Link quality changes rapidly and with unpredictable fashion in MANETs where each node moves randomly. Consequently, route lifetime is reduced considerably, and a QoS routing mechanism should be able to find new routes to maintain connectivity and route reliability while preserving QoS requirements. The theme of this chapter is about testing the performances of QOSRGA such that it can (1) provide for effective operations over a range of networking contexts, and (2) react effectively to topological changes and traffic demands while maintaining effective routing in mobile networking contexts. In evaluating the merit of a routing protocol, one needs both qualitative and quantitative metrics with which to measure its suitability and performance. In this chapter, first the qualitative properties are described and then quantitative characteristics based on the simulation results are obtained using Opnet Modeler. The aim of the simulation experiments was to use the various performance metrics such as packet delivery ratio, end to end packet transmission delay, throughput and normalised routing load in the comparative evaluation of the proposed QOS routing algorithms. QOSRGA was compared with the most common MANET protocols which are BE-

DSR and BE-AODV. BE-DSR and BE-AODV were described in Section 2.2.3.1 and in Section 2.2.3.2 respectively.

The remainder of this chapter is organised as follows. Section 7.2 describes the qualitative properties of the QOSRGA protocol. Section 7.3 describes in detail the performance metrics, the simulation model and the scenario used in the simulation experiments. Section 7.4 presents the impact of source traffic rate variations on the performance metrics. Section 7.5 investigates the impact of node mobility on the performance metrics. Section 7.6 presents the influence of node density on the performance of the QOSRGA protocol. Section 7.7 elaborates on the effect of congestion level on the QoS routing protocol. Section 7.8 summarises the chapter.

7.2 Qualitative Properties of QOSRGA

This section outlines the qualitative properties of QOSRGA which are compatible within the context of mobile networking environment. The following list describes the qualitative properties that can be attributed to QOSRGA:

(i)   *Distributed operation.* The overall protocol is run on a per node basis. Its operation is triggered by the arrival of packets from the next hop neighbours. No central unit exists to organise the network. End nodes function as source or sink. The intermediate nodes function as routers, forwarding packets to the next hop neighbours.

(ii)   *Loop-freedom.* The protocol inherently avoids looping in order to increase the overall performance. One of the processes in the QOSRGA QoS routing operation is to check for loop occurrences. If loops occur then a restoration process to remove the redundant nodes is initiated.

(iii)   *Demand based operation.* The protocol is designed to operate on demand. Demand is initiated at the Application Layer by the packet generation process which embeds bandwidth and delay requirements. When demand is not available the periodic connectivity packet transmission still continues to operate in which case it maintains the information regarding neighbour nodes connectivity.

(iv)   *Interaction with the standard IP routing protocol.* QOSRGA protocol is designed to be either a source or a sink within MANET. This is done by building the QOSRGA code as a Process Model which is a child process of the manet_manager. The manet_manger itself is a child process of the IP Process Model. The routing protocol successfully interacts with the standard IP layers by adhering to the proper IP addressing and convention. The IPv4 convention was used throughout.

(v)   *QOSRGA is built on three cooperative protocols.* QOSRGA consists of three cooperative protocols which include the Non-disjoint Multiple Routes Discovery protocol, the Node State Monitoring protocol and QoS Route selection using Genetic Algorithm.

7.3 Simulation Environment Model, Performance Metrics and Scenarios

OPNET 10.5 Modeler [29] was used in creating a simulation environment to develop and analyse the proposed QOSRGA and compare its performance with the already existing on-demand BE-AODV and BE-DSR routing protocols.

7.3.1 Physical Environment

The field configuration assumed was a flat square shape, with the dimensions 1000m by 1000m [52][53] typical for a campus area environment. The maximum number of nodes in the network was set at 40 nodes [52][53]. The physical layer parameters were as follows: maximum data rate of 2 Mbps, operating frequency of 2.4 GHz, transmit power of 5mW using DSSS modulation. Free space propagation with log-normal shadowing model was assumed. In addition, the wireless transmission range was set at 250 m. Each node used the basic configuration IEEE 802.11 [28], CSMA/CA operating in the DCF mode. All transmitters and receivers were configured such that they were included when performing the computation. A node was considered to be within another node's transmission range if it satisfied the distance and path loss calculation. The initial setup of the simulation is shown in Figure 7-1.

Figure 7-1 The Initial Setup of Simulation Environment with 40 Nodes

7.3.2 Mobility and Traffic Models

Mobility and traffic models similar to those previously reported [52][54] were used. The random waypoint model [57] was used to model the random movement of nodes. Each node started its journey from a random location to a random destination point with a specific speed. Once the destination was reached, stopped for a duration of time, then calculate another random destination point . Initial angle of motion for every node is set at 0 degree. In this version of Opnet, the random waypoint mobility model was included with the Wireless Module of the Opnet Modeler. It was then configured into the model through the Mobility Profile Definition. It was specified that all the nodes moved within the boundary of the field configuration. The pause time was kept constant at 1 second [52][53] for all the simulation experiments. This gives consistency in the nodes' movement for all the scenarios. The start time was set to zero until the end of simulation. The only variation within the RWP model was the speed. The speed was configured into a uniform distribution between zero and V_{max}, where V_{max} can be set accordingly.

169

Traffic sources with 512 bytes data packets [52] [53] [55][56] were CBR in nature. The source-destination pairs were spread randomly over the network and the number of sources was varied to change the offered load into the network. During the lifetime of a flow, a source node continuously generated data packets at the rate determined by the inter-arrival rate. The sending rate was varied according to Table 7.1, from between 24 pps to 292 pps. Nodes in all the three protocols maintain an infinite send buffer which contain queued packets. Each node buffered all data packets while waiting for a route. All packets (both data and routing) sent by the routing layer were queued at the buffer until the MAC layer was able to transmit them. Routing packets were given higher priority than data packets in the buffer.

Table 7.1 Traffic Sending Rates

Inter Arrival Rate(sec)	Traffic Rate (pkts/sec)	Traffic Rate (kbps)
0.2048	4	20
0.1024	9	40
0.0683	14	60
0.0512	20	80
0.04096	24	100
0.02048	49	200
0.01365	74	300
0.01024	98	400
0.008192	122	500
0.006827	146	600
0.005851	170	700
0.00512	195	800
0.004551	219	900
0.004096	244	1000
0.003724	268	1100
0.003413	292	1200

Simulations were run for 200 simulated seconds [52][53][56]. Each data point represented an average of 10 runs with identical traffic models, but different randomly-generated mobility scenarios by using different seeds to the random

number generator. Another interesting aspect of the protocol design was to understand the protocol performance with various congestion levels.

7.3.3 Performance Metrics

The following metrics [57] were used in varying scenarios to evaluate the three different protocols:

Average packet delivery ratio: Since the study is essentially based on bandwidth measurement, a metric was proposed which expressed the efficiency of bandwidth, as an average packet delivery ratio. The average packet delivery ratio (APDR) was defined as the ratio between the total packets generated by every node to the total received packets at the upper layer within the nodes in the system. It was expressed it in terms of a percentage.

$$APDR = \frac{\sum_{1}^{TOTAL\ NUM\ NODES} Average\ Number\ of\ Packet\ Arrived\ at\ Nodes\ Upper\ Layer}{\sum_{1}^{TOTAL\ NUM\ NODES} Average\ Number\ of\ Packet\ Generated\ in\ The\ Nodes\ Upper\ Layer}$$

Average total end to end delay of data packets: This includes all possible delays from the moment the packet is generated to the moment it is received by the destination node. The statistics of average delay of all the packets received during the simulation time were taken and then divided by the average total number of packets arrived at every receiving node. This gaves the average delay of a packet.

$$ATETED = \frac{\sum_{\forall\ DELIVERED\ PACKETS} (Time\ Of\ Packet\ Arrival_{DESTINATION} - Time\ Of\ Packet\ Sent_{SOURCE})}{\sum_{1}^{TOTAL\ NUM\ NODE} Number\ of\ Packets\ Delivered}$$

Total Average Throughput: In this context the throughput is defined as the total number of bits (in bits/sec) forwarded from the WLAN layers to higher layers in all WLAN nodes of the network. To find the average throughput of a single node, one has to divide by the number of nodes in the system.

$$TAT = \sum_{1}^{\text{TOTAL NUM NODE}} \left(\frac{\text{Total Number of Packets Delivered to This Node}}{\text{Time Taken To Deliver These Packets}} \right)$$

Normalised Routing Load: In order to measure the cost of the QOSRGA protocol as compared to other protocols, a metric was included to evaluate the overhead necessary for successful data packet transmission. The overhead was usually measured as the number of control packets transmitted to establish and maintain the paths in the network. This is rather misleading, since the amount of resources wasted due to imprecise routing information is not consider. Hence in this work a Normalised Routing Load was used, which considered all the routing packet that were dropped in other nodes in the network. The Normalised Routing Load was defined as follows:

$$NRL = \frac{\sum_{1}^{\text{TOTAL NUMBER OF NODES}} \left(\text{Average Number Of Routing Packets Re ceived by Destination Node} \right)}{\sum_{1}^{\text{TOTAL NUMBER OF NODES}} \left(\text{Average Number Of Data Packets Re ceived by Destination Node} \right)}$$

The Normalised Routing Load is an important metric to compare the performance of different protocols, since it can give a measure of the efficiency of protocols, especially in a low bandwidth and congested wireless environment. Protocols that transmit a large number of routing packets can also increase the probability of packet collisions and waiting time of data packets in transmission buffer queues.

7.4 Impact of Source Traffic Rate Variation on Performance

The simulation experiments were carried out by keeping the maximum node velocity constant at 2 m/s, with 40 nodes. This was to study the effect of varying the source traffic rate from 20kbps to 1200 kbps. In the simulation environment, 10 nodes were set as the sources to random destination nodes. The traffic rate of the sources was varied by configuring the source node with an exponentially-distributed inter-arrival rate.

7.4.1 Average Packet Delivery Ratio (APDR)

The traffic that was considered originated from the *manet_mgr* process model and also sinks at that level in the destination node. It took into account the data transmission rate and the control packet transmission. The control packet transmission must be considered, since these packets also load the network. In order to compare the APDR, the average of total traffic sent and average of the total traffic received were recorded for each traffic rate of 20 kbps to 1200 kbps. For each traffic rate of the load, the simulation was repeated for 10 runs and the average readings were recorded. Each run used a seed with different values, so as to diversify the simulation output. Hence each point in the graph is a result of 10 runs. The ratios of average total packets received to average total packets sent were taken for each traffic rate. A plot of APDR against source traffic rates is shown in Figure 7-2. At low source traffic rate, all protocols showed similar results. The plots dropped rapidly until about 40% where each produced a different rate. BE-DSR goes further until 20%, where it stays constant. QOSRGA performed a little better than BE-AODV. When searching for the routes, QOSRGA readily acquired network information that included the bandwidth availability and the connectivity index. It had chosen the route which had less probability of being lost in the near future. It chose a route which was more reliable than the other two BE protocols. When the source traffic was more than the 400 kbps, congestion caused the ratio to stabilise at approximately 40%.

7.4.2 Average End to End delay of data packets

Figure 7-3 depicts the variation of the average end-to-end delay as a function of the traffic rate. It can be seen that the QOSRGA protocol has a lower average delay than BE-DSR and BE-AODV under all source traffic rates. The primary reason is that the number of route discoveries is reduced in QOSRGA.
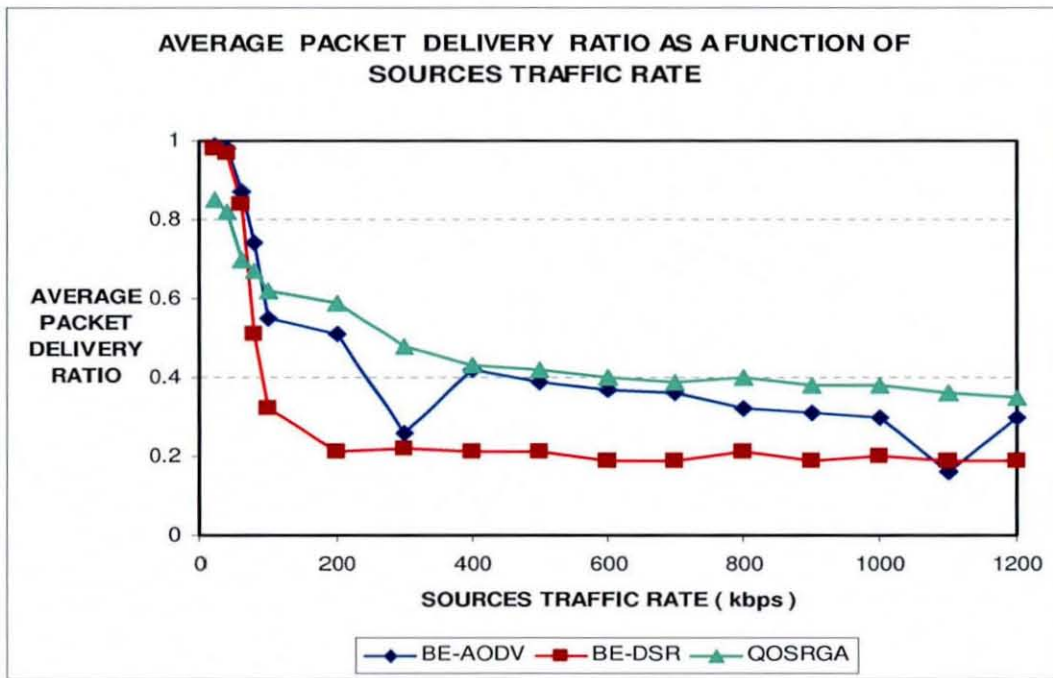
AVERAGE PACKET DELIVERY RATIO AS A FUNCTION OF SOURCES TRAFFIC RATE

Figure 7.2 APDR against the Source Traffic Rates

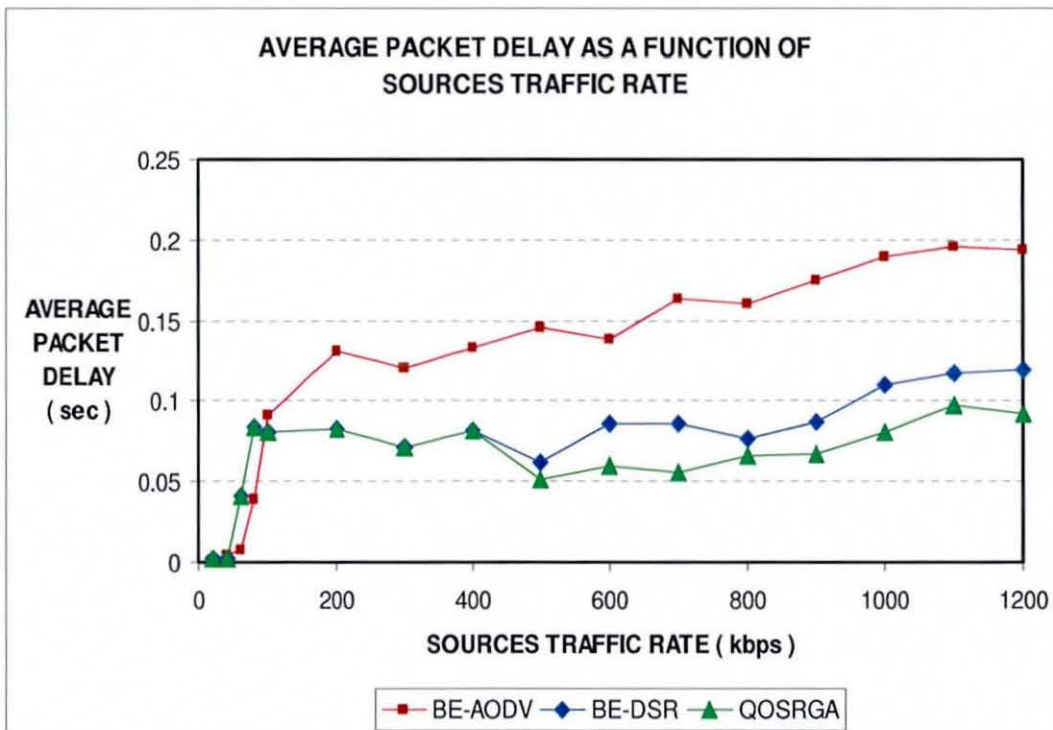AVERAGE PACKET DELAY AS A FUNCTION OF SOURCES TRAFFIC RATE

Figure 7-3 Average End To End Delay Against Source Traffic Rate

Although QOSRGA has a low number of route discoveries, its delay also decreases gradually with increase in the traffic rate. The reason is that an increase in the traffic rate leads to higher network load traffic in the ad hoc networks. But because QOSRGA was set to choose enough bandwidth first, then, due to the GA process, the route selection always tries to locate the route with enough bandwidth without dropping packets, as happens for BE-DSR and BE-AODV. Beyond 800 kbps, the congestion was so high that the average delay for the QOSRGA protocol increased. From 20 kbps until 1200 kbps, the average packet delay for QOSRGA was less than 100 ms, which is within the stated QoS requirements as in Table 5.1.

7.4.3 Total Average Throughput

Figure 7-4 shows the total average throughput against the source traffic rate. Below 100 kbps, the performance of QOSRGA is similar to BE-DSR. The rate of increase in throughput for QOSRGA and BE-AODV is steeper between 100 kbps until 600 kbps compared to BE-DSR. Beyond 600 kbps, the throughput increase rate starts to reduce. One observation is that QOSRGA and BE-AODV delivered a similar performance, while QOSRGA is approximately 5% better than BE-AODV and almost 100% better than BE-DSR. When the source node generates packets at higher bandwidths, the GA process works better by choosing a path which can accommodate the given bandwidth. DSR on the other hand, dropped all packets that cannot be accommodated, due to limited bandwidth.

7.5 Impact of Node Mobility on the Performance

The second set of simulation experiments varied the velocity for a 40 nodes network and 10 CBR sources. The mobility was varied to see how it affected the different metrics that were measured. The packet sending rate was fixed at two rates: 98 pkts/sec (400 kbps) and 4 pkts/sec (20kbps). The simulations were run with uniform velocity, where the maximum velocities were 0.5, 1, 1.5, 2, 5, 10, 15, 20, and 25 m/s. In all the simulations, the velocity was limited to 25 m/s, since the analysis in **Chapter 4** suggested that the upper bound on the velocity is limited up to 25 m/s which is equivalent to 90 km/h, in order to maintain node connectivity successfully. In most literature, pause time is used instead of velocity.

Each data point was obtained after 10 runs with different seed values for the random number generator.

7.5.1 Average Packet Delivery Ratio

The graph of Average Packet Delivery Ratio against node maximum velocity is shown in Figure 7-5. Two set of results were obtained, one for CBR sources at 4 packets/sec and the other for 98 packets/sec. For 4 packets/sec sources BE-AODV performed better than BE-DSR and QOSRGA for the whole maximum velocity range. By comparing QOSRGA and BE-DSR, QOSRGA produced a slightly better APDR. When the mobility is less than 12 m/s, QOSRGA gives a similar reading. When node mobility is more than 12 m/s QOSRGA performed better, in fact 5% better than BE-DSR. For high bandwidth sources of 98 packets/sec, clearly QOSRGA consistently performed better than BE-DSR and BE-AODV for all the mobility ranges. Generally, it is 5% to 15% better than BE-AODV and 5% to 30% better than BE-DSR. In QOSRGA, multiple routes were found with the corresponding QOS metrics information B_{AVA}, D_{ETE}, D_{MAC} and *nci*. The selection of the routes was based on the probable length of time each node pair stay connected, which is indicated by *nci*. The degradation of BE-DSR occurred as the mobility rate increases. In high mobility scenarios, many route reconstruction processes are invoked. When a source floods a new RREQ packet to recover a broken route, many intermediate nodes send RREP packets back to the source, because of the route caching mechanism of BE-DSR. However routes overlap the existing routes, hence resulting in severe congestion, and it cannot deliver packets along the route. Moreover the stale or outdated routes produce a reply to source with invalid routes. Ultimately, many packets are dropped, resulting in poor BE-DSR performance. In QOSRGA, an aging mechanism is used, hence the stale routes will be replaced.
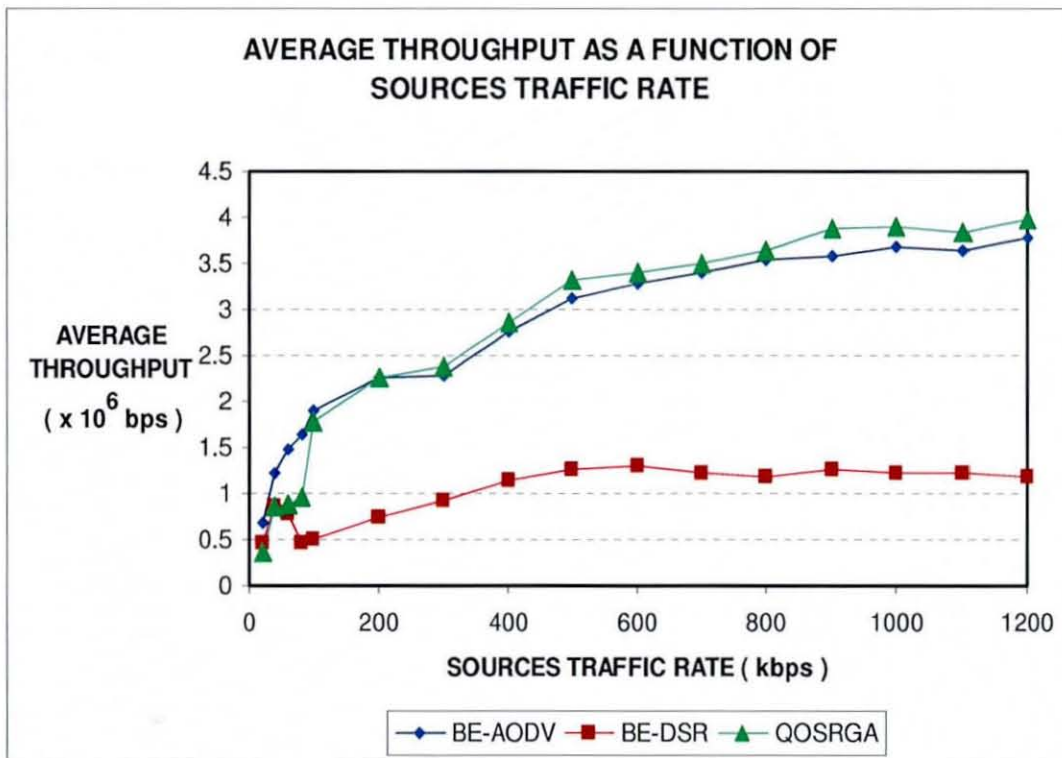
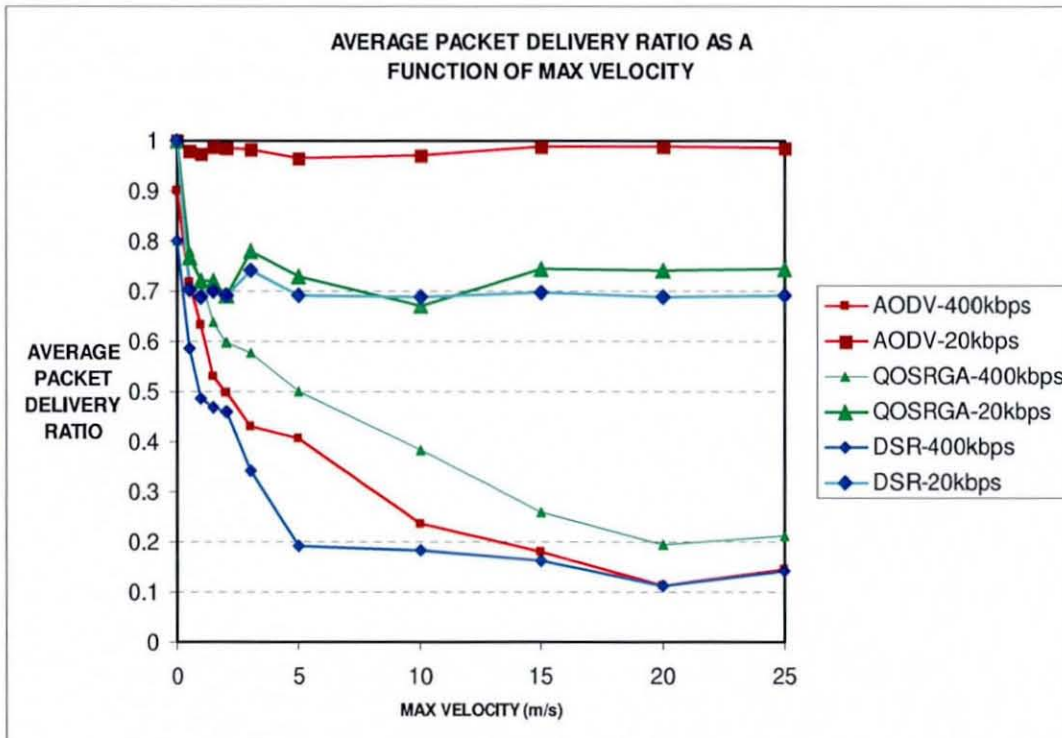Figure 7-4 Total Average Throughput against the Source Traffic Rate

Figure 7-5 The Average Packet Delivery Ratio as a Function of Maximum Velocity

7.5.2 Average End to End Delay of Data Packets

The average end-to-end delay includes all possible delays from the moment the packet is generated to the moment it is received by the destination nodes. Generally, there are three factors affecting end-to-end delay of a packet: (1) Route discovery time, which causes packets to wait in the queue before a route is found; (2) Buffering waiting time, which causes packets to wait in the queue before they can be transmitted; and (3) The length of the routing path. More hops means a longer time to reach the destination node. Figure 7-6 depicts the variation of the average end-to-end delay as a function of the velocity of nodes.

It can be seen that the general trend of all the curves is an increase in delay with the increase of velocity of nodes. The reason is mainly that high mobility of nodes results in an increased probability of link failure that causes an increase in the number of routing rediscovery processes. This means data packets have to wait longer in the queue until a new routing path is found. The delay of BE-DSR is lower than QOSRGA and BE-AODV for 98 packets/sec source data. When the source sends 4 packet/sec packets, BE-AODV and BE-DSR is better than QOSRGA. When the velocity is more than 5 m/s, the delay for all protocols is maintained at almost the same level. QOSRGA performed the worst. This is clear, since QOSRGA was designed to collect as much information about the network as much as possible, so that the process of route selection using the GA is based on all possible information. However, all the delays incurred by QOSRGA are still less than 0.1 second. This is because availability of cached routes in QOSRGA eliminates route rediscovery latency that contributes to the delay when an active route fails. In addition, when a congestion state occurs in a routing path, the source node can distribute incoming data packets to the other non-disjoint routing paths to avoid congestion. This reduces the waiting time of data packets in the queue.

7.5.3 Average Total Throughput

Figure 7.7 shows the total average throughput of the QOSRGA compared to the other protocols. Throughput is the total number of bits delivered to the destination hosts. For QOSRGA, the ability of transferring the data dropped from 2.5 Mbps to 1.5 Mbps as the mobility increases from 2 m/s to 25 m/s. The

throughput is less than that of BE-AODV, but, when compared to BE-DSR, QOSRGA offered an improvement of 25% to 80%. Nodes with high velocity will produce small numbers of low value *nci* among the node pairs. The number of routes of longer lifetime will be less, and hence the rate of data transfer to the destination nodes will be less.
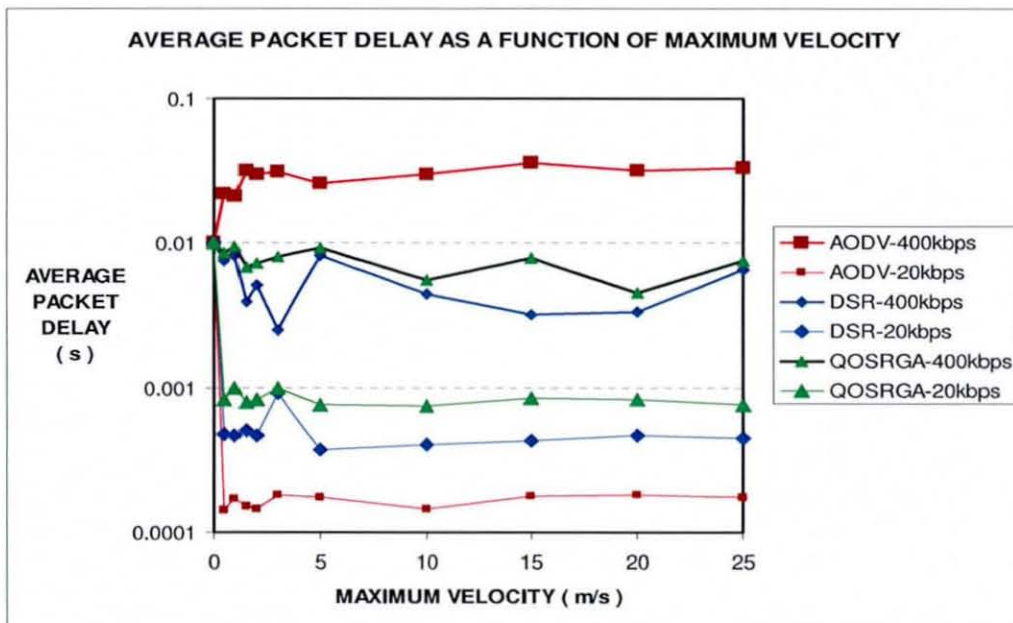

Figure 7.6 Average End to End Delay As a Function of Maximum Velocity
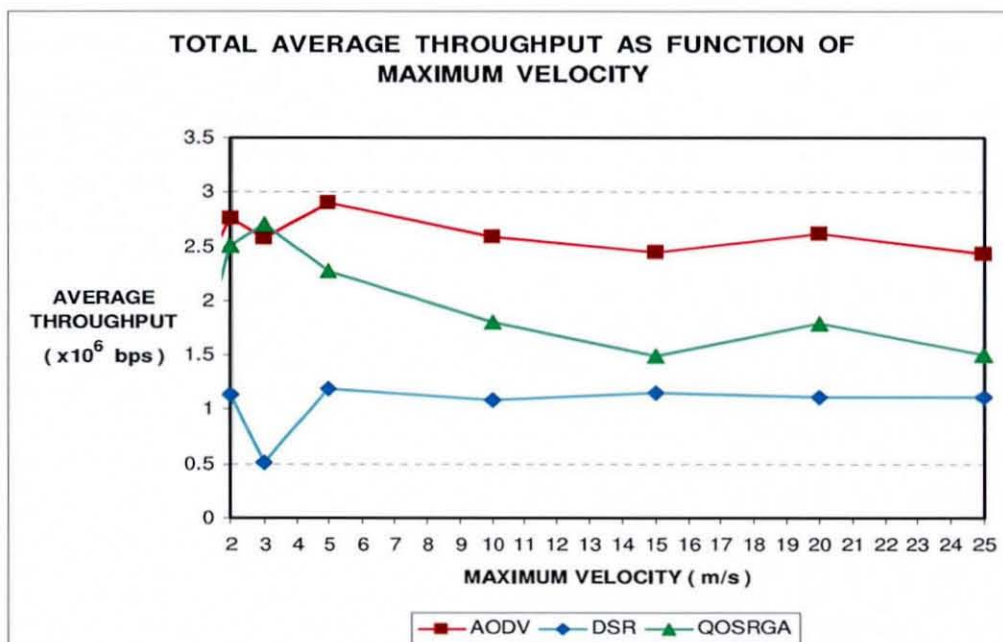

Figure 7.7 Average Total Throughput against Mobility Level

179

7.6 Influence of Node Density on Performance

The ability of different MANET protocol schemes to handle node density was analysed in this set of simulations. It inherently assessed the scalability of QOSRGA, and compared its performance to BE-DSR and BE-AODV with different node densities. In this case, the area, source traffic rate and maximum velocity are kept constant at 1000 x 1000 m^2, 100 kbps and 2 m/s respectively. The number of CBR source nodes was set to 10, generating CBR towards random destinations. Each point was obtained after 10 runs with different seed numbers. The node density is defined as the ratio of the total number of nodes in the network to the area of the field configuration. The number of nodes were varied from 10 to 50, with the same size field configuration. The simulation experiments are based on the densities of $1x10^{-5}$, $2x10^{-5}$, $3x10^{-5}$, $4x10^{-5}$ and $5x10^{-6}$ nodes per square metre. The metrics measured were: (i) average packet delivery ratio, (ii) average end to end packet delay, (iii) average normalised routing load, and (iv) average throughput.

7.6.1 Average Packet Delivery Ratio

Figure 7.8 illustrates the Average Packet Delivery Ratio for QOSRGA, BE-AODV and BE-DSR as a function of node density. Overall the patterns of the QOSRGA graph and BE-DSR graph are normally quite similar. BE-DSR fall more rapidly from its maximum value down to a density of $2.0x10^{-6}$, and then stabilised at $4.0x10^{-6}$ and $5.0x10^{-6}$. After $5.0x10^{-6}$ onwards, QOSRGA produced better results. It is 10% better than the BE-DSR at high node density. The operation of QOSRGA requires fast accumulation of multiple routes. As the node density increases, a reasonable number of routes as an initial population can be obtained. A good number of multiple routes ensure better selection process by the GA algorithm. A route with very low *nci* could then be produced and selected.
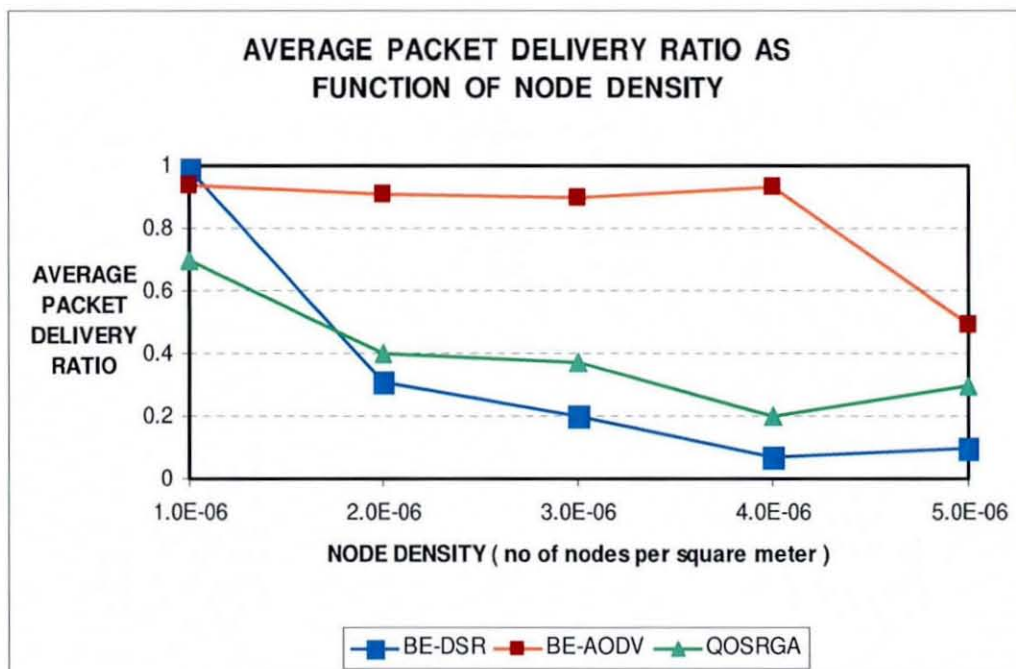
180

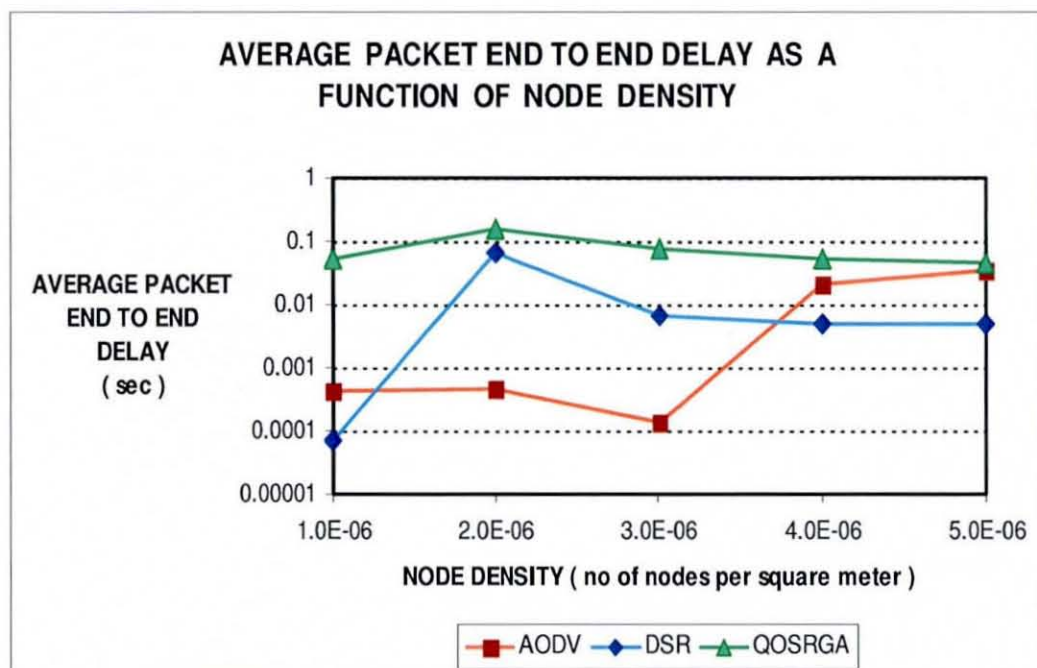Figure 7-8 Average Packet Delivery Ratio against Node Density

Figure 7-9 Average end to end delay against node density

7.6.2 Average End to End Delay of Packets

Figure 7-9 illustrates the Average End to End Packet Delay as a function of node density. Generally QOSRGA performed the worst among the other protocols. It varied from approximately 0.05 to 0.1. This can be attributed to the fact that for QOSRGA, in every intermediate node, the processing time is significant. The node had to do the monitoring of packet arrival, set and reset all the node cache, perform the bandwidth calculation, *nci* calculation and, most importantly, run the route selection routine using the GA. Hence, the packet delay observed from Figure 7-9 can be attributed to these node state manifestations. However, for node densities of more than 3.0×10^{-6}, the delay is less than 0.1 second, which is the maximum delay allowed.

7.6.3 Average Normalised Routing Load

Figure 7-10 shows the Average Normalised Routing Load (ANRL) as a function of node density. Since QOSRGA first performs the NDMRD protocol route discovery algorithm, the number of routing packets generated contributed significantly to the number of packets in transition within the wireless framework. The second phase of QOSRGA is to perform the route selection algorithm among the valid routes discovered by NDMRD protocol. These two phases require a significant overhead routing load. For densities less than 3×10^{-6}, the value for ANRL is less than that for BE-DSR. Beyond that BE-DSR is better. A high density configuration produces more nodal interactions for QOSRGA, resulting in very high routing packets per data packet. For low densities (10 nodes and 20 nodes) the value of routing packet per data packet is approximately equal to BE-DSR.

7.6.4 Average Total Throughput

Figure 7.11 shows the average total throughput of the QOSRGA compared to other protocols. The average throughput increased as the node density is increased. In this context, QOSRGA performed on a par with the BE-DSR protocol. When a node density is high, then the probability of longer route lifetime can be realised.

182

7.7 Effect of Congestion level on the Performance

To study the impact of congestion level on QOSRGA performance, the number of active flows in the simulation was varied from 2, 5 and until 30 in steps of five. Each traffic flow carried traffic of 100kbps or 24 packets/sec. The mobility level was set to a normal distribution with maximum velocity 2 m/s. The number of nodes was fixed at 40. The metrics measured were: (i) average packet delivery ratio, (ii) average end to end packet delay, (iii) average normalised routing load, and (iv) average throughput.
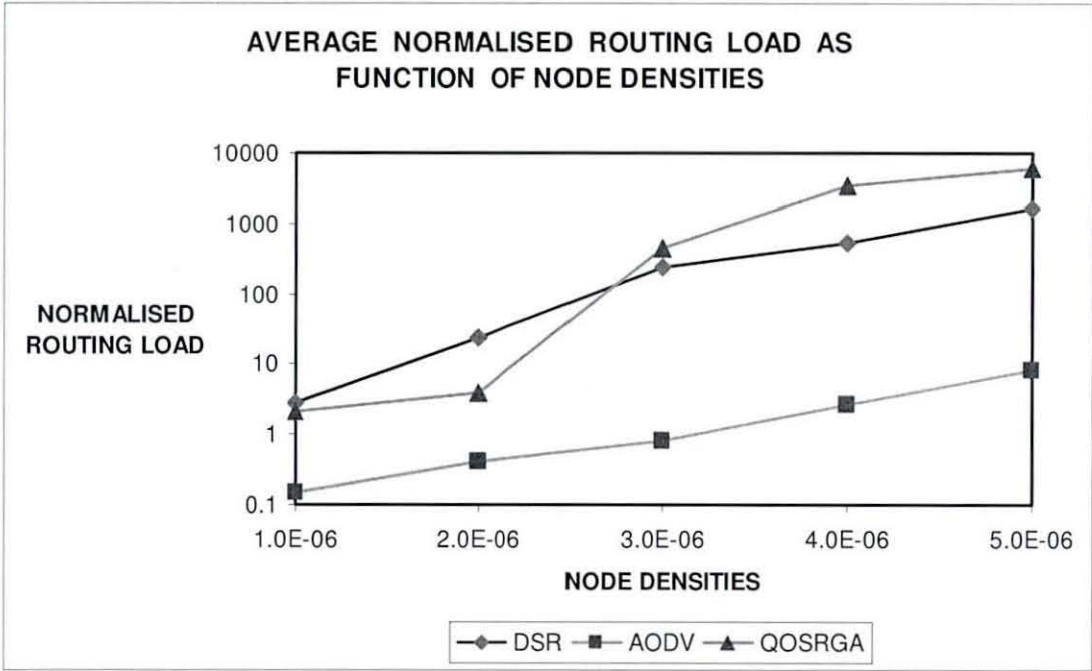
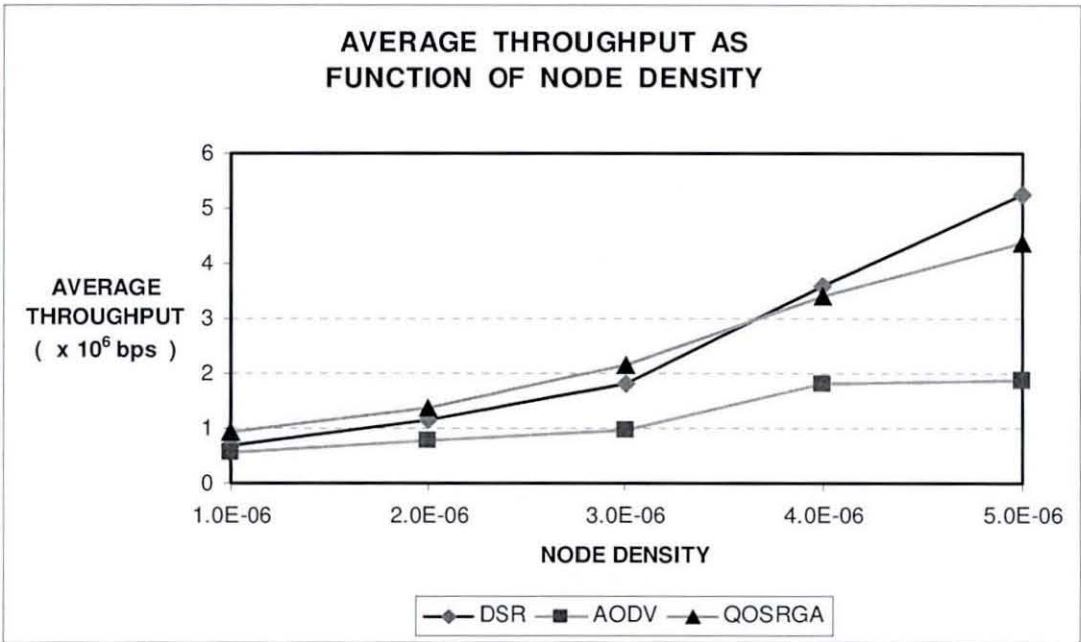Figure 7.10 Average Normalised Routing Load against Node Density

Figure 7.11 Average Total Throughput against Node Density

7.7.1 Average Packet Delivery Ratio

Figure 7.12 shows the graph of average packet delivery ratio as a function of network traffic load. Generally all protocols, are sensitive to an increase in traffic load. Node mobility is the same for all protocols and hence the packet drops are caused by buffer overflow, collision and congestion. BE-DSR is most sensitive to traffic congestion. QOSRGA performs better than BE-DSR, and at some points shows an improvement of more than 40%. The reason is that there are more collisions in the air and congestion in node buffers, when the number of sources increases. In QOSRGA, more load means more RREQ packets being sent. More route replies were generated within the Route Accumulation Latency. The population size of the routes will increase, giving a better chance of selection of the fittest solution that is the longest connectivity pair within the routes. Furthermore, less route breakage will occur in the whole session compared to BE-DSR.

7.7.2 Average End to End Delay of Packets

Figure 7.13 shows the effect of congestion level on Average End to End Packet Delay for QOSRGA, compared to BE-AODV and BE-DSR. QOSRGA had a larger delay compared to BE-DSR, due to the high routing overhead as the network got more congested. QOSRGA and also BE-DSR had to perform route rediscovery if the existing route is broken. This may add to the delay. At the highest congestion of 30 flows, the delay for QOSRGA was equal to delay for BE-AODV. When the number of sources was high, the overall delay for QOSRGA packet was increased. BE-AODV used a scheme where the node upstream of the disconnected link initiated an immediate route reconstruction. Since the route rediscovery is done locally, less time is needed to search for and obtain a new route. Due to this local recovery schemes delays., BE-AODV yielded shorter
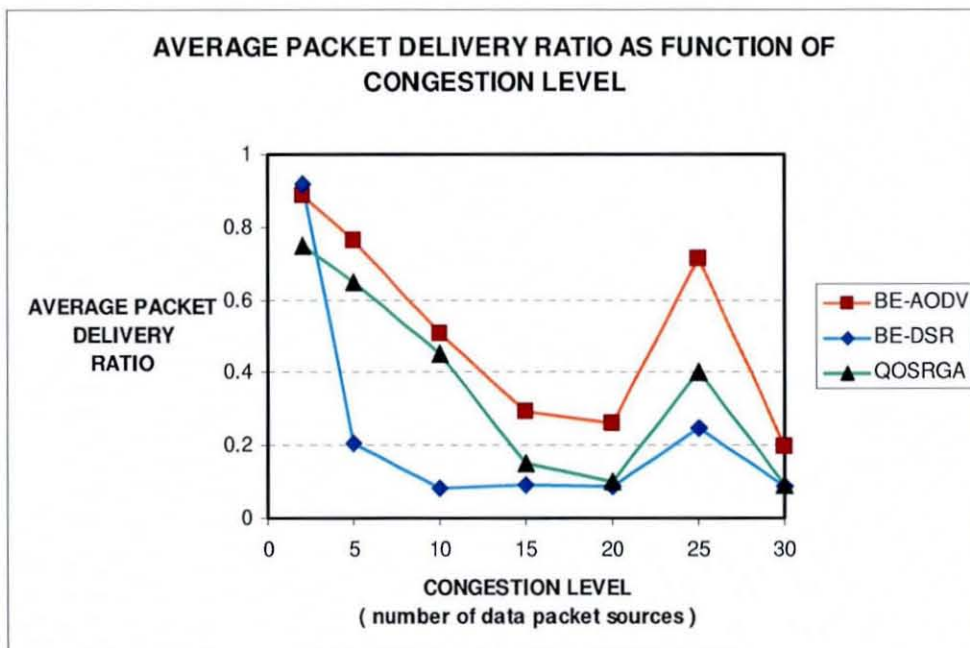
delays.

Figure 7.12 Average Packet Delivery Ratio as a function of Congestion Level
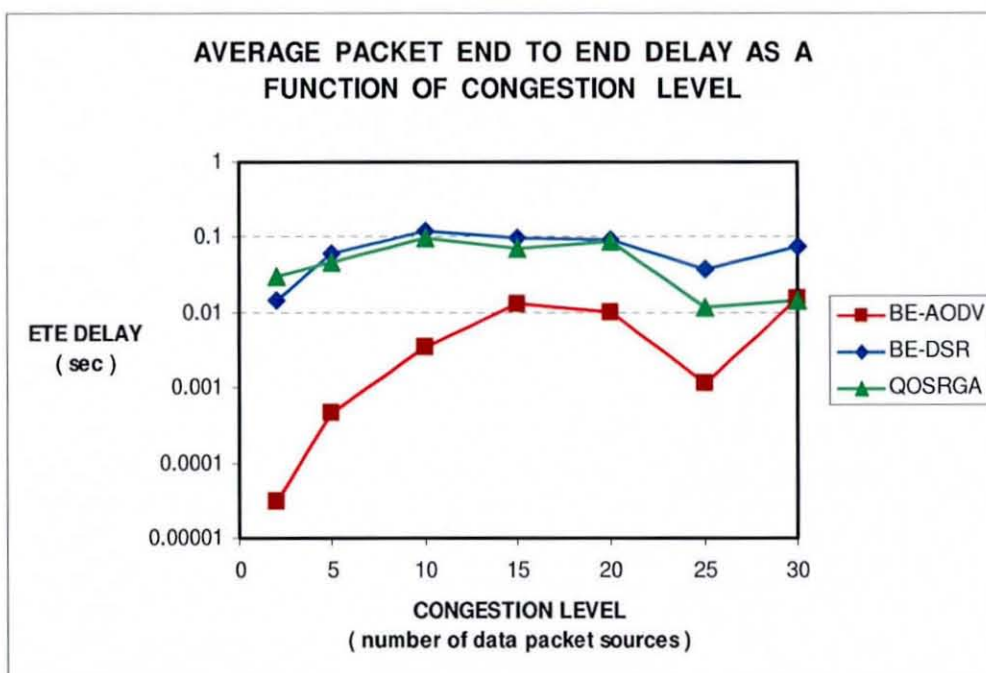

Figure 7.13 Average End to End Delay Against Congestion Level

7.7.3 Average Total Throughput

Figure 7.14 shows the average total throughput of QOSRGA compared to other protocols. QOSRGA performed better than BE-DSR by about 5%. As the number of sources was increased, the throughput of QOSRGA decreased. The reason is that there are more collisions in the air and congestion in node buffers, when the number of sources increases.

7.7.4 Average Normalised Routing Load

Figure 7.15 depicts the Average Normalised Routing Load as a function of congestion level. With an increase of the number of sources, the probability of packet collision and packet congestion increases. This leads to the decrease of normalised routing load. It is seen that QOSRGA has similar ANRL to both BE-AODV and BE-DSR for all possible numbers of sources. The ANRL in BE-AODV and BE-DSR decreases more quickly than that in QOSRGA with an increase of the number of sources. The reason is that QOSRGA has multiple non-disjoint routes. When an active routing path encounters packet congestion due to high network load traffic, the source node will terminate the current route and calculate the next route. To avoid the loss of data packets, the source node can at once select another valid non-disjoint routing path from its route table to send data packets towards the destination.

7.8 Summary

In this chapter, a novel and practical QOSRGA protocol was analysed for mobile ad hoc networks. The protocol can reduce routing control overhead dramatically and achieves multiple non-disjoint routing paths. The simulation model of QOSRGA in the OPNET Modeler was also implemented. Performance results for unipath routing protocols BE-AODV, BE-DSR and QOSRGA were compared for different scenarios. Simulation results show that performance of QOSRGA is better than BE-DSR. When compared with BE-AODV, it is better than BE-AODV in many scenarios.
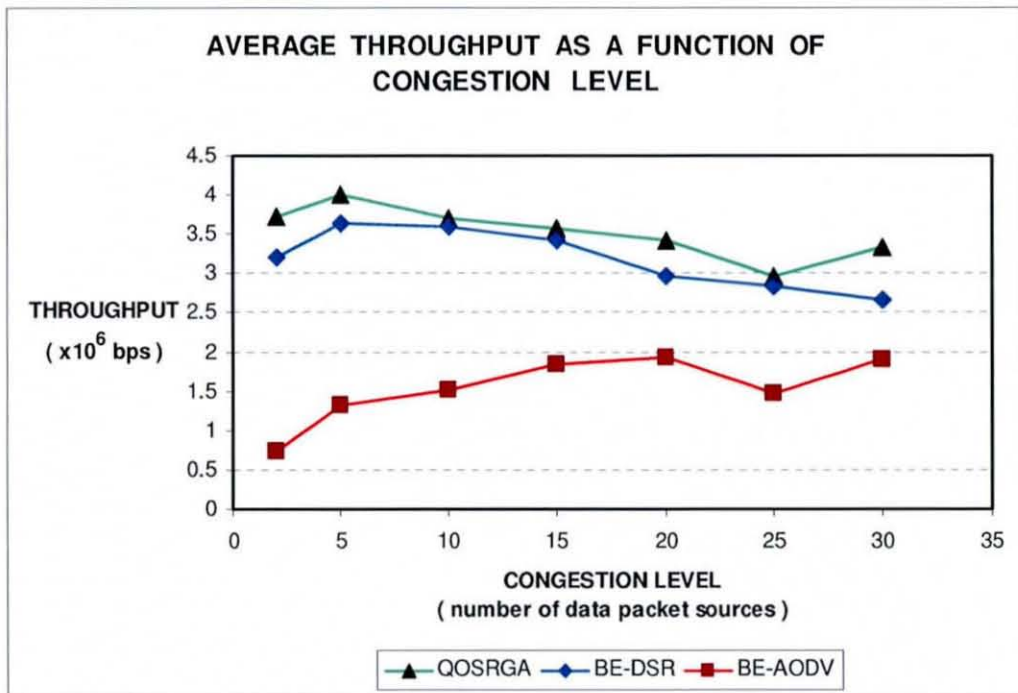
Figure 7.14 Average Total Throughput against the Congestion Level
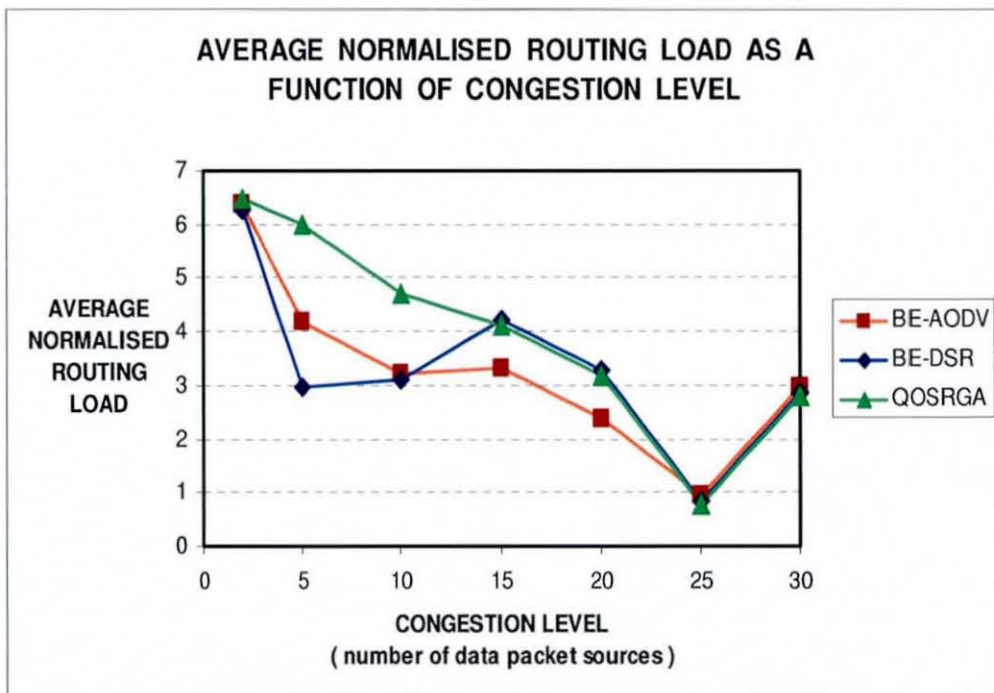
Figure 7.15 Average Normalised Routing Load as a Function of Congestion Level

188

CHAPTER 8
SUMMARY AND FUTURE WORK

8.1 Introduction

The goal of this thesis has been to use a Genetic Algorithm to develop an efficient QoS route selection algorithm for mobile ad hoc networks. The focus was to explore the functionality of GAs in searching for optimal solutions with several constraints, such as bandwidth availability, end-to-end delay requirements, and the *nci* metrics. The selection was done simultaneously using the GA mechanism. A GA based QoS routing protocol was successfully designed as a collection of three cooperative protocols. These protocols are: the Non-Disjoint Multiple Routes Discovery protocol, the Node State Monitoring protocol, and the GA-based QoS Route Selection protocol. This was achieved through the development of a QoS routing framework that provided a means for testing the operation of the protocol. The test was organised around various scenarios such as when the number of nodes varied, the maximum velocity varied, the node density varied and also when the source traffic rate varies.

8.2 Research Contributions

The overall implementation of the QoS routing algorithm for MANET is presented in the thesis. QOSRGA is a collection of cooperative protocols that have to function in tandem to each other. These cooperative protocols include: the Non-Disjoint Multiple Routes Discovery (NDMRD) protocol, the Node State Monitoring protocol, and a GA-based QoS route selection protocol. The main contributions of this thesis are categorised as follows.

8.2.1 Non-Disjoint Multiple Routes Discovery Protocol

The NDMRD is a QoS-Aware protocol since the contents of its Route Reply packet consists of QoS information. It initiates the propagation of Route Request packets towards the destination. The salient feature is that in each node it allows Route Request packet duplication, so that a good number of non-disjoint routes are obtained. Non-disjoint routes are necessary in this work, in order to increase the chances of getting a better solution after the process of crossover and mutation. Route Reply packets then extract QoS information from nodes that make up the route, and are carried to the source node. The protocol caused the accumulation of routes within the Route Accumulation Latency period. The number of Route Request duplicates, and the value of Route Accumulation Latency, are predetermined. These values are chosen in such a way to ensure high throughput performance. Since the consideration of delay and possible congestion, the limit imposed on the two variables results in less QoS information being collected at the source. Hence, the NDMRD protocol produced imprecise information, which is in reality a trade-off with a low throughput performance. The imprecise information, here means that many more Route Reply packets with the QoS information are dropped due to time limitation. Besides this, a *converging storm* phenomena might occur at the destination node, which may reduce the QoS information packets further. The *converging storm* means that the destination receives a great number of Route Request packets from the same source, resulting in an buffer overflow.

190

8.2.2 Node State Monitoring Protocol

The importance of Node State was examined by systematically defining its function, elaborating its advantages and suggesting an initial list of Node States. The list of Node States can be expanded further. A Node State is a viable means to produce a specific characterisation of the network. A suitable system model and monitoring scheme was developed whereby a Node State Monitoring protocol was implemented for MANET. The scheme involved monitoring packet arrivals of various types, extracting the Node State, distributing QoS parameters, and regularly updating the Node State to avoid stale values. The most important aspect of the Node State in this work is that it can infer node connectivity and variation in topology.

8.2.3 Node Bandwidth Measurement Using NAV

When dealing with QoS routing, it is imperative that the protocol gets accurate information on the consumed bandwidth and available bandwidth. A method has been proposed of node bandwidth measurement. In measuring the instantaneous node bandwidth, the NAV duration was used at the MAC layer. By using NAV, not only the instantaneous bandwidth of a node was measured, but also all the neighbouring nodes that were within the contention range. An algorithm for determining the bandwidth in two stages, was designed, one for calculating channel busy time, and another performed the sampling of the busy time. The method was validated by performing a simulation experiment to compare the true bandwidth setting with the measurement reading. The reading fell reasonably close to the true value. Hence, it is believed Hence, it is believed that the measure is appropriate and can be used in the QoS Routing protocol. Incidentally, the measurements took place at the MAC layer in a cycle of 20 ms, the bandwidth value was output, and was read by the Node State Monitoring protocol at the Network Layer. Hence the QoS Routing protocol is a cross layer protocol.

8.2.4 Development of a *nci* as a new Mobility Metric

A novel mobility metric, *nci,* was developed which could indicate the length of time a node is in connection with its single-hop neighbour. It depends on the relative velocity of the node-pair and the power as a result of the packet arrival.

191

The indication of time is not absolute time, rather a value for the purpose of comparison amongst the node-pair. Initially, the contraction and expansion mobility model was developed where *npem* and *npcm* were defined. The model used two consecutive packet arrivals to resolve the power due to packet arrival and time associated with the packet transmission and reception. It was shown that the *nci* can be used to select the node which will result in longer connectivity time, which will ultimately contribute to a longer route lifetime.

8.2.5 Designed a Method of QoS Route Selection Using a GA with Low Population Size

A method of QoS route selection was developed using a Genetic Algorithm. Initially the representation of a chromosome is chosen, and the format depends largely on how the data structure is to be manipulated. For QOSRGA, variable length chromosomes was used with the maximum length, depending on the number of nodes in the system. A chromosome represents routes which are a collection of nodes from source to destination, whereas a gene represents a node. Encoding processes convert the route list as a result of NDMRD protocol into a set of chromosomes which represent the initial population. One salient feature of the GA method is the used of a small population instead of a typically large population. This technique is rationalised by referring to other literature. The initial route list that results from the NDMRD protocol is already a set of solutions. The GA is used merely to determine the most reliable route. Another feature is that at least one node must exist in a route that is shared by other routes. The existence of common nodes increases the convergence rate towards the final solution. This is because the crossover and mutation operations are effective if there are common genes. These are the crossover points and mutating points.

Fitness computation is the most important aspect of the GA. It is here that the overall quality parameter of the route is measured simultaneously. The fitness function was designed successfully incorporating bandwidth, end-to-end delay, node delay and *nci* into one single objective function using the weighted sum approached. The determination of the weighting coefficient, however, depends on various factors. The most important of these is the QoS parameters that needs to

be dominant. For QOSRGA, equal importance was placed on all the QoS metrics: bandwidth, delay and *nci*.

Other important GA operators are the crossover and mutation operator. Since the chromosome length is variable, the position of crossing site is arbitrary, but must be at a common node of the two parents. The algorithm also takes care of the possibility of generating infeasible chromosomes. A restoration function is built into the proposed QOSRGA protocol to eliminate any lethal gene, without dropping any of the chromosomes. The mutation operator generates another chromosome from a chromosome. It functions to generate the genetic diversity of the population.

QoS route selection presents a serious challenge. In the literature there are few techniques that practitioners normally use. Some methods are better than others for specific applications. In this case, it was necessary to use the method which only required the least number of generations to converge. Hence some offline tests were performed in order to identify the most appropriate selection methods. It was concluded that tournament selection is the most appropriate. Mutation and crossover probability were chosen systematically. Both were chosen by looking at the final output. The value of the throughput at each setting was used to indicate which scenario performed better.

The performance of QOSRGA was obtained by running the simulation experiment for a medium size network and illustrating the characteristics of the protocol. The influence of congestion and mobility on QOSRGA performance were compared which includes APDR, delay and throughput for two different source traffic types.

8.3   Future Work
In this section, a set of topics that extends the proposed framework is presented. A set of topics that can be extended are presented as follows.

8.3.1 Improving the GA Technique for QoS Route Selection
The current GA technique requires a large amount of memory for it to operate successfully. This is due to less importance being placed on the memory

optimisation and also on the programming techniques. Extension to the current work might involve a radical way of formulating the GA process such as the population size, the number of generations and, most importantly, the way QoS parameters are organised into data structures. A good data structure formulation is necessary in order to increase the efficiency in running the GA. The efficiency of the GA relies on the information gathered during the course of Non-Disjoint Multiple Routes Discovery. By restricting the amount of information collected and processes to be completed, the potential of the GA as a premier methodology for QoS routing could be increased considerably. In doing the Multiple Route Discovery for GA route calculations, limitations on the number of maximum hops to each destination could be imposed. Additionally a cluster mechanism could be applied for a very large number of nodes. Hence, a radically new QoS Routing solution could be evolved by considering the clustering mechanism.

8.3.2 QOSRGA Coupled with Bandwidth Reservation

The work in this thesis can be improved further by including a bandwidth-reservation mechanism. Bandwidth estimation has already been done. The bandwidth can actually be reserved for the whole session, or for 95% of the route lifetime after it has been selected using the GA technique. For the extra 5% of the time before route breakage occurs, the protocol can be modified in such a way that it starts to search new routes by recalculating the fitness function.

8.3.3 QOSRGA with added Compression Algorithm

One of the biggest limitations of the **QOSRGA** protocol is that it uses source routing to route the data packets. The data packet consists of a list of all nodes that need to be traversed. Similarly, the **RREP** packet consists of a list of nodes to reach the source node. The results of this are that the bandwidth overhead is significant, when the source and destination are far from each other. The **RREP** carries extra information which includes the QoS parameters towards the source node after the monitoring process. The overhead could be reduced by designing a scheme whereby the list of the information collected and attached to the packet is compressed. Opportunities exist using compression algorithms, such as the Bloom filter [58] . It was reported that Bloom filters reduce the per-packet control

overhead of the DSR protocol. It increases the network capacity by a factor of up to 15. Hence, by coupling the QOSRGA with a Bloom filter, it is anticipated that the protocol could be applied potentially to a very large mobile ad-hoc network [59].

8.3.4 Modify QOSRGA Into QOSRGA(BW-biased), QOSRGA(DL-biased) and QOSRGA(BW-DL)

The current version of QOSRGA provides a balanced way of testing the operation of the QoS Route selection algorithm by setting the weightings in Eqn. 6.7 such that it can work with either bandwidth biasing, delay biasing or a combination of bandwidth-delay biasing. The operation of the selection algorithm can then be modified adaptively such that it is possible to choose the QOS requirement according to the current applications either voice, video or pure text and image data.

8.3.5 Developing a fully-fledge working algorithm

QOSRGA was designed and implemented in a simulation mode. Although it is compiled and runs on the Opnet Modeler framework, it only works within the Opnet Modeler simulation platform. In reality, to really make it useful, the simulation code needs to be transformed into working software which can be applied to real terminals and interfaces. The fully-fledged working algorithm could be developed using C++ , and installed into 802.11 enabled terminals.

8.3.6 TCP performance in Ad Hoc Networks using the QOSRGA protocol

Performance of TCP in wireless MANETs was studied in [60] . It performed very poorly due to high packet losses, which resulted in a high number of TCP retransmission time-outs. First, a node drops a packet if it cannot forward the packet to the next hop of the route on which the packet is to be relayed, as the next hop node has moved out of the transmission range. The second reason for the packet loss is the congestion in the shared medium. In the second case, a node cannot reach the next hop, because there are too many nodes trying to access the channel at the same time, considering the medium access protocol of IEEE802.11 is used. Node mobility has the effect of severely degrading the performance of TCP, even at very light loads.

It is believed that the QOSRGA protocol could stem the degradation of TCP performance due to mobility. In the design of QOSRGA, it has been shown that the node connectivity index can be used to identify a route which has a longer lifetime compared to other discovered routes. The protocol can be modified such that the time of departure (TOD) can be estimated for nodes within the route. The TOD is the time when the node is nearly out of range from the other node. Then the route can be produce proactively from a list of routes that were ranked earlier. In this, the GA will operate on a population of routes, that not only produce a single best route but a number of the most reliable routes. To enhance the speed of route calculation, another interesting technique which can be investigated further is performing the GA calculation in the background.

In order to cope with failures that are due to congestion, a mechanism could be proposed by which the MAC layer, based on its estimate of whether a neighbour is still within the transmission range, persists in its attempt to reach that neighbour for a longer period of time. In this case the fitness function of the QOSRGA, can be adjusted such that it gives more importance to the connectivity rather than bandwidth and delay. That is to improve on the biasing techniques in the fitness function. By doing so, the GA always tries to locate the route which would stay connected for a longer period of time.

8.4 Conclusions

In this thesis, a set of cooperative protocols have been designed, developed, and performance tested that provide support for QoS routing in Mobile Ad Hoc Networks. It includes the Non-Disjoint Multiple Routes Discovery (NDMRD) protocol, the Node State Monitoring protocol and the online implementation of QoS route selection using a Genetic Algorithm. It has been shown how the QOSRGA performs in various simulation experiments. Comparative performance evaluations were also done and have shown that QOSRGA could potentially be offered as an alternative protocol for QoS routing.

Appendix A

The following lists the author's publications in chronological order.

1. J. Abdullah and D.J. Parish, *"QoS Routing For Mobile Ad Hoc Networks"* Proceedings, The 5th Annual Postgraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting, Liverpool John Moores University, UK, 28-29 June 2004, PGNET2004

2. J. Abdullah and D.J.Parish, *"Impact of QoS Routing Metrics for MANETs in the Pervasive Computing Environment."* Proceedings, IEEEE 2007 Fourth International Conference on Wireless and Optical Communications Networks, (IEEE WOCN2007), Singapore, July 2-4, 2007.

3. J.Abdullah and D.J.Parish, *"Node Connectivity Index as Mobility Metric for GA based QoS Routing in MANET"*, Accepted by ACM/IEEE International Conference on Mobile Technology, Application and Systems, 10-12 September, Singapore.

4. J.Abdullah and D.J.Parish, *"Effect of Mobility on the Performance of GA-based QoS Routing in Mobile Ad Hoc Networks"*, for the ICIAS 2007, IEEE International Conference on Intelligence & Advanced Systems, 25-28 Nov 2007, Kuala Lumpur.

References

[1] Imrich Chlamtac, Marco Conti, Jennifer J.-N. Liu, "Mobile ad hoc networking: imperatives and challenges," *Adhoc,* vol. 1, pp. 13-64, 2003.

[2] Manet Working Group Charter, "Internet Engineering Task Force," 2005. Available: http://www.ietf.org/html.charters/manet-charter.html; Last accessed: 30 May 2007.

[3] S. B. Lee, A. Gahng-Seop, X. Zhang and A. T. Campbell, "INSIGNIA: An IP-based quality of service framework for mobile ad hoc networks," *Journal PADC,* vol. 60, pp. 374-406, 2000.

[4] G. S. Ahn, A. T. Campbell, A. Veres and L. H. Sun, "Supporting Service Differentiation for Real-Time and Best-Effort Traffic in Stateless Wireless Ad Hoc Networks (SWAN)," *IEEE TMC,* vol. 1, pp. 192-207, July-September 2002.

[5] P. Mohapatra, J.Li and C.Gui, "QoS In Mobile Ad Hoc Networks," *IEEE Wireless Communications,* vol. 20, pp. 44-52, June 2003.

[6] J.L. Sobrinho and A.S. Krishnakumar, *"Quality-of-Service in Ad Hoc Carrier Sense Multiple Access Wireless Networks"*, JSAC, Vol 17, N8, August 1999, pp 1353-1368.

[7] C. R. Lin and M. Gerla, "Asynchronous multimedia multihop wireless networks," in *Proc. IEEE INFOCOM,* 1997, pp. 118-125.

[8] D. B. Johnson, D. A. Maltz and Y. C. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)" *IETF MANET Working Group, INTERNET-DRAFT,* 2007. Available: http://www.ietf.org/rfc/rfc4728.txt Last accessed: 30 May 2007.

[9] D. B. Johnson, D. A. Maltz and Y. C. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)" *IETF MANET Working Group, INTERNET-DRAFT,* 2007. Available: http://www.ietf.org/rfc/rfc4728.txt Last accessed: 30 May 2007.

[10] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," *Proc IEEE Mobile Computer Systems and Applications,* pp. 90-100, 1999.

[11] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *Computer Communications Review,* vol. October 1994, pp. 234-244, 1994.

[12] R. Sivakumar, P. Sinha and P. Bharghavan, "CEDAR: A Core-Extraction distributed Ad Hoc Routing Algorithm " *IEEE JSAC,* vol. 17, 1999.

[13] S. Chen and K. Nahrstedt, "Distributed Quality Of Service Routing In Ad Hoc Networks," *IEEE JSAC,* vol. 17, pp. 1488-1505, 1999.

[14] J. C. Z.Wang, "Quality of Service Routing For Supporting Multimedia Applications," *IEEE JSAC,* vol. 14, pp. 1228-1234, 1996.

[15] Q. Ma and P. Steenkiste, "Quality-of-service routing for traffic with performance guarantees," in *Proceedings of IFIP Fifth International Workshop on Quality of Service,* 1997.

[16] B. Li and K. Nahrstedt, "A control theoretical model for quality of service adaptations " in *Proceedings of Sixth IEEE International Workshop on Quality of Service,* 1998, pp. 145-153.

[17] D. A. Coley, *"An Introduction to Genetic Algorithms for Scientist and Engineers",* World Scientific Publishing, Singapore, 1999.

[18] M. Gen and R. Cheng, *"Genetic Algorithms and Engineering Optimization."* Wiley-Intersciences Publication, Canada, 2000.

[19] R. Elbaum and M. Sidi, "Topological Design of Local Area Networks Using GA," *IEEE/ACM Transactions on Networking,* vol. 4, pp. 766-778, Oct 1996.

[20] S. Mao, Y. T. Hou, X. Cheng, H. D. Sherali and S. F. Midkiff, "Multipath routing for multiple description video in wireless ad hoc network." *IEEE INFOCOM2005,* 2005.

[21] R. H. Hwang, W. Y. Do and S. C. Yang, "Multicast Routing Based on GA," *Journal of Information Science and Engineering, Institute of Information Science, Taiwan,* vol. 16, pp. 885-901, 2000.

[22] Y. Leung, G. Li and Z. B. Xu, "A genetic algorithm for the multiple destination routing problems. " *IEEE Transactions on Evolutionary Computation,* vol. 2, pp. 150-161, 1998.

[23] X. Xianwei and Changjia, C. and Gang, Z., "A genetic algorithm for multicasting routing problem," in *Proc. Int. Conf. Communication Technology (WCC-ICCT 2000),* 2000.

[24] S. H. Wong and J. Wassell, "Dynamic channel allocation using a genetic algorithm for a TDD broadband fixed wireless access network," in *Proc. IASTED International Conference in Wireless and Optical Communications, July 17-19, 2002, Banff, Alberta, Canada, Pp. 521-526.* 2002, pp. 521-526.

[25] N. Shimamoto, A. Hiramatus and K. and Yamasaki, "A dynamic routing control based on a genetic algorithm." in *Proceedings of the IEEE International Conference on Neural Networks,* pp. 1123-1128.

[26] M. Munetomo, Y. Takai and Y. Sato, "A migration scheme for the genetic adaptive routing algorithm. " in *Proceeding of the IEEE International Conference on Systems, Man, and Cybernetics* 1998, pp. 2774-2779.

[27] C. Y. Lee and E. K. Antonsson, "Variable length genomes for evolution strategies", in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2000)* 2000.

[28] J. Inagaki, M. Haseyama and H. and Kigajima, "A genetic algorithm for determining multiple routes and its applications," in *Proceedings of the IEEE International Symposium on Circuits and Systems,* 1999, pp. 137-140.

[29] Opnet Modeler. Available: www.opnet.com. Last Accessed: 30 May 2007.

[30] W.H. Liao, Y.C. Tseng, J.P. Sheu, and S.L. Wang. *"A Multi-Path QoS Routing Protocol in a Wireless Mobile Ad Hoc Network."* In Proc. of IEEE ICN'01: International Conference on Networking, Part II, pages 158–167, July, 2001.

[31] A. Nasipuri, R. Castaneda, and S.R. Das, *"Performance of multipath routing for on-demand protocols in mobile ad hoc networks,"* in ACM/Baltzer Mobile Networks and Applications Journal, Vol 6, 2001, pp. 339-349.

[32] C. K. Toh, "*Associativity based routing for ad hoc mobile networks,*" *WPCJ*, vol. 4, pp. 103-139, 1997.

[33] R. Dube, C. D. Rais, K. Y. Wang and S. K. Tripathi, "*Signal Stability Adaptive Routing(SSA) for Ad Hoc Mobile Networks,*" *IEEE PC,* 1997.

[34] S. Agarwal, A. Ahuja, J. Singh and R. Shorey, "*Route-Lifetime Assessment Based Routing (RABR) Protocol for Mobile Ad-Hoc Networks,*" *IEEE ICC,* pp. 1697-1701, 2000.

[35] T. Goff, N. B. Abu-Ghazaleh, D. S. Phatak and R. Kahvecioglu, "*Preemptive Routing in Ad Hoc Networks,*" *ACM/IEEE MobiCom,* pp. 43-52, 2001.

[36] W. Su, S. J. Lee and M. Gerla, "*Mobility Prediction and Routing in Ad Hoc Wireless Networks* " *Intl Journal of Network Management, Wiley & Sons,* 2000.

[37] D. Hong and S. S. Rappaport, "*Traffic Model and Performance Analysis For Cellular Mobile Radio Telephone Systems with Prioritized and NonPrioritized Handoff Procedures.*" *IEEE Trans. OVT,* vol. 35, pp. 77-92, 1986.

[38] T. S. Rappaport, *Wireless Communications Principles & Practice* Upper Saddle River, New Jersey: Prentice-Hall, 1996

[39] Chenxi Zhu and M. Scott Corson, "QoS routing for mobile ad hoc networks," in *Proc IEEE INFOCOM 2002,* 2002, pp. 958-967.

[40] T. W. Chen, J. T. Tsai and M. Gerla, "QoS routing performance in multihop, multimedia, wireless networks," *IEEE UPC,* vol. 2, pp. 557-561, 1997.

[41] Yihai Zhang, Gulliver, T.A. , "Quality of Service for Ad Hoc On-Demand Distance Vector Routing " in *Proc of IEEE Intl Conf on Wireless And Mobile Computing, Networking And Communications,* pp 192-196 Vol 3, 2005..

[42] IEEE LAN MAN Standards Committee, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, IEEE Std 802.11a" Tech Rep IEEE Std. 802.11a, 1999

[43] S.H. Shah, K. Nahrstedt, " Predictive location-based qos routing in mobile ad hoc networks, in *Proceedings of IEEE ICC 2002*, vol. 2, May 2002, pp. 1022–1027.

[44] M. S. Gast, *802.11 Wireless Networks, the Definitive Guide.* US: O'Reilly & Associates, Inc., 2002.

[45] J. Jun, P. Peddabachagari and M. Sichitiu, "Theoretical maximum throughput of IEEE 802.11 and its applications," in *Proc. IEEE International Symposium on Network Computing and Applications,* 2003.

[46] Allen B. Downey, "*Clink: a tool for estimating Internet link characteristics*", SIGCOMM99, 1999.

[47] Kevin Lai, Mary Baker. "*Measuring Link Bandwidth Using A Deterministic Model of Packet Delay*", Proc. SIGCOMM 2000, Stokholm..

[48] A. J. Chipperfield and P. J. Fleming, *The MATLAB genetic algorithm toolbox.* Available: http://www.shef.ac.uk/acse/research/ecrg/getgat.html. Last accessed: 30 May 2007.

[49] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms.* Canada: Wiley-Interscience, 1998.

[50] J.D. Schaffer; R.A. Caruana;L.J. Eshelman and R.Das, "*A study of control parameters affecting online performance of genetic algorithms for function optimization,Proceedings of the third international conference on Genetic algorithms*", 1989, pp51-60,George Mason University, Morgan Kaufmann Publishers Inc.

[51] Haupt, R.L. , *Optimum population size and mutation rate for a simple real genetic lgorithm that optimizes array factors*, Antennas and Propagation Society International Symposium, 2000. IEEE, Vol.2, Iss., 2000, pp:1034-1037 vol.2.

[52] S.J.Lee and M.Gerla, "*Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks*", In *Proceedings of the IEEE ICC*, pages 3201-3205, 2001.

[53] B. Zhou, A. Marshall, Tsung-Han Lee, "*A Cross-Layer Architecture for DiffServ in Mobile Ad-hoc Networks*", in *Proceedings of International Conference on Wireless Networks, Communications and Mobile Computing*, Maui, Hawaii, USA, June 13-16, 2005.

[54] C. E. Perkins, E. M.Royer and S. R.Das, *"Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks"*, IEEE Personal Communications, Feb 2001, pp16 - 28.

[55] Mahesh K. Marina, and Samir R. Das, *"On-demand Multipath Distance Vector Routing in Ad Hoc Networks"*. In *Proceedings of the 9th IEEE International Conference on Network Protocols (ICNP)*, Page(s):14 – 23, November 2001.

[56] Charles E. Perkins, Elizabeth M.Royer and Samir R.Das, *"Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks"*, *IEEE Personal Communications*, Feb 2001, pp16 - 28.

[57] S. Corson and J. Macker, *"Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations,"* IETF RFC 2501, January 1999. Available: http://www.ietf.org/rfc/rfc2501.txt 46. Last accessed: 30 May 2007.

[58] C.Castelluccia and P.Mutaf, *"Hash-Based Dynamic Source Routing"*, LNCS Volume 3042/2004, pg 1012-1023, Springer Berlin, 2004.

[59] F. Sailhan and V. Issarny, *"Scalable Service Discovery for MANET"*, Proc. 3rd IEEE Percom, 2005.

[60] T. Saadawi and S. Xu, *"Performance Evaluation of TCP Algorithms in Multi-hop Wireless Packet Networks"*, Journal of Wireless Communications and Mobile Computing, 2002.