

UNIVERSITY OF CALGARY

A QoS Routing Framework for MANETs

by

Oscar Salazar Gaitán

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE
STUDIES IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

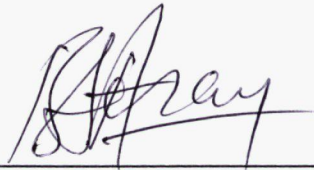
CALGARY, ALBERTA

JULY, 2003

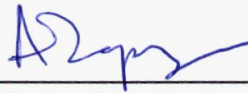
© Oscar Salazar Gaitán 2003

UNIVERSITY OF CALGARY
FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "QoS Routing Framework for MANETs" submitted by Oscar Salazar Gaitán in partial fulfillment of the requirements for the degree of Master of Science.



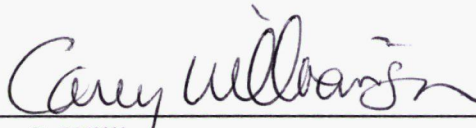
Supervisor, Dr. A. B. Sesay,
Department of Electrical and Computer Engineering



Co-Supervisor, Dr. A. O. Fajoluwo,
Department of Electrical and Computer Engineering



Dr. B. Far,
Department of Electrical and Computer Engineering



Dr. C. Williamson,
Department of Computer Science

July 28, 2003
Date

ABSTRACT

In a mobile ad-hoc network (MANET) that supports multimedia services, achieving and maintaining quality of service (QoS) pose a great challenge because of the dynamic nature of the network topology and the difficulty of finding QoS enabled routes between two communicating nodes. This thesis presents a QoS routing framework that addresses the combined issues of QoS support and routing in MANETs. The QoS routing framework for MANETs proposes a set of ideas to enable QoS provisioning regardless of the underlying routing protocol. Computer simulations are used to illustrate the improvements in important network parameters such as average route discovery time, routing overhead and power consumption, enabled through the application of the QoS routing framework.

The main advantage of the QoS routing framework is that QoS support can be enabled in routing protocols for MANETs originally designed without this functionality.

ACKNOWLEDGEMENTS

I would first like to thank my advisors, Dr. Abraham O. Fapojuwo and Dr. Abu B. Sesay, for all their guidance during my years as a graduate student. I also thank them for trusting me and recruiting me into ENEL and for helping me secure funding from TRILabs during the masters program.

I thank TRILabs for the support provided during my research and also thank all my fellow labmates for making TRILabs such a great place to work.

I would also like to thank the other members of my thesis committee, Dr. Behrouz Far and Dr. Carey Williamson for taking the time to participate on my committee and for providing additional feedback into my research.

I give special thanks to the Government of the State of Colima, especially to Fernando Moreno and Victorico Rodriguez for showing such an interest in my professional development. Lastly and most important, I thank my family for all the love, guidance and support they have always given me. Thank you for being supportive and for always believing in me.

Table of Contents

Approval page	ii
Abstract	iii
Acknowledgements	iv
Table of contents	v
List of Tables	viii
List of Figures	ix
List of Notations	xi
Abbreviations and Acronyms	xi
CHAPTER ONE: INTRODUCTION	1
1.1 Motivation	1
1.2 Infrastructure-based Wireless Networks	1
1.3 Mobile Ad-Hoc Networks	3
1.3.1 Complexity	4
1.3.2 Applicability	4
1.4 Quality of Service	5
1.4.1 Quality of Service in MANETs	6
1.5 Routing	13
1.5.1 Routing in MANETs	15
1.6 QoS Routing in MANETs	17
1.7 Thesis Contributions	18
1.8 Thesis Overview	18
CHAPTER TWO: A QoS ROUTING FRAMEWORK FOR MANETs	20
2.1 Introduction	20
2.2 QoS Routing Issues	22
2.2.1 QoS Routing Overhead Issue	22
2.2.2 QoS Route Discovery Issue	23

2.2.3	QoS Robustness Issue	23
2.2.4	QoS Route Maintenance Issue	24
2.3	The QoS Routing Framework.....	25
2.3.1	Assumptions	25
2.3.2	Routing Overhead Reduction Technique	26
2.3.3	QoS Route Discovery Approaches	28
2.3.4	QoS Robustness Enhancement Ideas	29
2.3.5	QoS Route Maintenance Techniques	30
2.4	Application of QoS Routing Framework.....	31
2.4.1	QoS Route Discovery	31
2.4.2	QoS Route Maintenance	36
2.4.3	QoS Robustness Enhancement.....	40
2.5	Summary	49

CHAPTER THREE: QoS ROUTING FRAMEWORK SIMULATION

	ENVIRONMENT	50
3.1	Introduction	50
3.2	Simulation Objective	50
3.3	Simulation Structure Using OPNET	51
3.3.1	OPNET Node Model for QoS Routing Framework	51
3.3.2	QoS Routing Process Model	54
3.3.3	OPNET Network Model	58
3.4	Inputs to the Simulation	60
3.4.1	Size of the Network	60
3.4.2	Physical Layer	60
3.4.3	Radio Propagation Loss	60
3.4.4	MAC Layer	61
3.4.5	QoS Routing Protocol	62
3.4.6	Application Layer	63
3.4.7	The Mobility Model	64
3.5	Simulation Outputs	65

3.6	Verification of Simulation Outputs	66
3.7	Validation of Simulation Outputs.....	67
3.8	Summary	70
CHAPTER FOUR: QoS ROUTING FRAMEWORK		
	PERFORMANCE	71
4.1	Introduction	71
4.2	QoS Routing Framework Performance Simulation Results	71
	4.2.1 Impact of Mobility on Performance	71
	4.2.2 Impact of Congestion on Performance	79
4.3	Summary	85
CHAPTER FIVE: CONCLUSION		87
5.1	Thesis Conclusions	87
5.2	Suggestions for Future Work	88
BIBLIOGRAPHY		89

List of Tables

1.1	Service Differentiation	7
1.2	Routing Table for Source Node # 1	14
2.1	Route Request Packet Structure	32
2.2	Routing Table Structure Illustrating Modifications Needed for QoS Support.	32
2.3	Route Request Sent Repository Structure	33
2.4	Route Reply Packet Structure.....	35
2.5	Node 2 Routing Table	35
2.6	Traffic Class Requirements	36
2.7	Route Error Packet Structure.....	42
2.8	Routing Table Additions for Multiple Routes Support	44
2.9	Multiple Routes QoS RREQ Packet Structure	45
3.1	Traffic Parameters	52
3.2	Receiver Parameters	53
3.3	Transmitter Parameters	54
3.4	OPNET General Simulation Parameters	59
3.5	MAC Layer Parameters	62
3.6	Routing Protocols Comparison	62
3.7	M/M/1/K Queueing System Parameters	69

List of Figures

1.1 Infrastructure Based Wireless Network	3
1.2 Example of Ad-Hoc Network	4
1.3 Inter-connection of Devices	5
1.4 Wireless Network Interaction	5
1.5 INSIGNIA Wireless Flow Management Model at a Mobile Node	10
1.6 Routing Scheme for Source Node # 1	14
1.7 Hierarchical Routing	16
2.1 QoS Routing Framework	19
2.2 Interaction between OSI Model Layers to achieve Full QoS Support in MANETs ..	22
2.3 Tie-breaking Metrics for Route Selection.....	28
2.4 Route Discovery	35
2.5 Admission Control	38
2.6 QoS Resource Release	39
2.7 Simple Error	41
2.8 Critical Error	42
2.9 Multiple Routes QoS RREQ Process	46
2.10 Multiple Routes QoS RREP Process	47
3.1 OPNET Node Model	51
3.2 OPNET QoS Routing Finite State Machine	55
3.3 Example Network Model Showing Initial Locations of Nodes	58
3.4 Hidden Terminal Problem	60
3.5 Random Waypoint Mobility Model	65
3.6 QoS Routing Framework Validation	67
3.7 M/M/1/K Queueing Model	68
3.8 End-to-End Delay, Analytical vs. Computer Simulation	70
4.1 Link Partitions, Mobility	72
4.2 Link Partitions due to Hidden Nodes, Mobility	72
4.3 Mobility Factor	73
4.4 Average Number of Hops, Mobility	73
4.5 Data Packets Transmitted, Mobility	74
4.6 Route Request Packets, Mobility	75
4.7 Route Reply Packets, Mobility	75
4.8 Route Repair Attempts, Mobility	75
4.9 Route Error Packets, Mobility	75
4.10 Efficiency, Mobility....	77
4.11 End-to-End Delay, Mobility	77
4.12 Average Route Discovery Time, Mobility.....	78
4.13 Cumulative Routing Overhead, Mobility	79
4.14 Average Power Consumption per Node, Mobility.....	79

4.15	Link Partitions, Congestion	79
4.16	Link Partitions due to Hidden Nodes, Congestion	79
4.17	Data Packets Transmitted, Congestion	80
4.18	Route Request Packets, Congestion	81
4.19	Route Reply Packets, Congestion	81
4.20	Route Repair Attempts, Congestion	82
4.21	Route Error Packets, Congestion	82
4.22	Efficiency, Congestion	83
4.23	End-to-End Delay, Congestion	83
4.24	Average Route Discovery Time, Congestion	84
4.25	Efficiency, Resource Release.....	85
4.26	Throughput, Resource Release	85

0.1 List of Notations

K	Buffer Size
λ	Mean packet arrival rate
μ	Mean service rate
ρ	Traffic Intensity

0.2 Abbreviations and Acronyms

AODV	Ad-Hoc On Demand Distance Vector
ATM	Asynchronous Transfer Mode
BE	Best-Effort
CEDAR	Core Extraction Distributed Ad-Hoc Routing
CSMA / CA	Carrier Sense Multiple Access / Collision Avoidance
DARPA	Defense Advanced Research Projects Agency
DCF	Distributed Coordination Function
DF	Differentiated Services
DiffServ	Differentiated Services
DIFS	DCF Inter-Frame Spacing
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
DS-SS	Direct Sequence Spread Spectrum
ECN	Explicit Congestion Notification
FQMM	Flexible QoS Model for MANETs
GPS	Global Positioning System
IEEE	Institute of Electrical and Electronic Engineers.
IntServ	Integrated Services
IP	Internet Protocol
ISM	Industrial, Scientific and Medical band
MANET	Mobile Ad Hoc Network
Mcps	Mega Chips per Second
NIST	National Institute of Standards and Technology
OSI	Open System Interconnection
PAN	Personal Area Network
PDF	Point Coordination Function
PHB	Per-Hop Behavior
PRnet	Packet Radio Network

QoS	Quality of Service
RERR	Route Error
RREP	Route Reply
RREQ	Route Request
RSVP	Reservation Protocol
RTS / CTS	Request to Send / Clear to Send
SIFS	Short Inter-Frame Spacing
SLA	Service Level Agreement
SWAN	Service Differentiation in Stateless Wireless Ad-Hoc Network
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
WLAN	Wireless Local Area Network
ZRP	Zone Routing Protocol

Chapter 1

Introduction

1.1 Motivation

Future generation wireless networks will support multimedia and real time traffic with the Quality of Service (QoS) determined by the applications. Each application needs different levels of QoS to perform adequately. For example, real-time video is delay-sensitive whereas web surfing is delay-tolerant.

Assuring quality of service in mobile ad-hoc networks (MANETs) is not an easy task due to the dynamic nature of the network. To overcome this dynamic nature and support the services aforementioned, a QoS framework is needed. In the remainder of this thesis a robust QoS routing framework is proposed in order to provide the QoS requirements the application needs to perform over the wireless channel. The objectives of this thesis are to:

- Propose and design a QoS routing framework that enables QoS support in the routing process in mobile ad-hoc networks.
- Analyze the performance of certain parameters of the proposed QoS routing framework using queueing theory techniques.
- Construct a simulation model for a MANET and use the tool to evaluate the performance of different mechanisms that can be used to build an efficient QoS routing framework.

1.2 Infrastructure-Based Wireless Networks

The IEEE 802.11 working group was founded in 1987 to begin standardization of wireless local area networks for use in the ISM (Industrial, Scientific and Medical) band. In 1997 IEEE 802.11 was finally standardized and provided interoperability standards for WLAN manufacturers using the same configuration (11 Mcps DS-SS spreading and 2

Mbps data rates). IEEE 802.11 specifies physical and data link layer as well as two configurations for wireless networks: PCF (Point Coordination Function) and DCF (Distributed Coordination Function) [1].

Point coordination function uses an access point to coordinate medium access and packet routing. The access point polls each node in the network for packet transmission. If a polled node has a packet to send, this node is granted channel access and can transmit the packet within an interval of time. This access technique is known as contention free because the nodes do not have to compete for the channel. Thus, collisions are non-existent.

Distributed coordination function mode is used in ad-hoc networks. That is each node is a router, which receives and re-transmits the packets to the destination. The principal characteristic of this network is that the nodes have to compete for the channel using CSMA/CA (Carrier Sense Multiple Access / Collision Avoidance) and different spacing intervals called: DIFS (DCF Inter-Frame Spacing) and SIFS (Short Inter-Frame Spacing), depending on the priority of the transmission

The physical layer specifies the transmission medium and the multiple access technique that must be used. IEEE 802.11 operates in the ISM frequency band, which comprises 902-928 MHz, 2.4-2.483 GHz, and 5.725-5.825 GHz. This standard supports Direct Sequence Spread Spectrum or Frequency Hopping Spread Spectrum as multiple access technologies. Data rates from 1 to 2 Mbps are supported in the original standard. In 1999, the 802.11 High Rate standard called IEEE 802.11b was approved, thereby providing new data rates of 11 Mbps and 5.5 Mbps in addition to the original 2 Mbps and 1 Mbps rates (in 2.4 GHz band). Data rates of 54 Mbps are achieved in IEEE 802.11a in the 5 GHz band. Depending on the configuration, DCF or PCF wireless networks use different medium access protocols. In DCF, the nodes use CSMA/CA combined with RTS (Request to Send) / CTS (Clear to Send) to overcome the hidden terminal problem. PCF uses the polling protocol to grant access to the channel.

High Performance Local Area Network (HiperLAN) is the European standard for wireless LANs. This network operates in 5.7 and 17.1 GHz providing data rates of 1 to 20 Mbps to the users. Mobility is considered and it can operate up to vehicle speeds of 35

km/hr. HiperLAN 2 has emerged as the next generation of wireless network in Europe and will provide up to 54 Mbps to a variety of networks. HiperLAN 2 supports ad-hoc network mode to meet the requirements needed in future generation wireless networks. Figure 1.1 illustrates an example of an infrastructure-based wireless network.

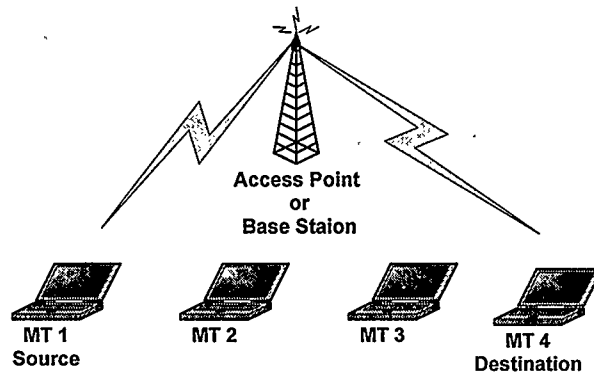


Figure 1.1: Infrastructure-Based Wireless Network

1.3 Mobile Ad-Hoc Networks

Mobility plays an important role in future generation wireless networks due to the advances in portable devices such as notebook computers, PDAs (Personal Digital Assistant), GPS (Global Positioning System), and handheld devices. Networking, another important area in telecommunications, is adopting wireless technology to provide mobility and portability to the users. Mobile ad-hoc networks, known also as infrastructure-less wireless networks, allow peer-to-peer communications among the network nodes without the need for an access point. Another important characteristic is that each node may have to act as a router to convey information from one node to another.

PRnet (Packet Radio Network) was the first ad-hoc network and it was created by DARPA for military applications [2]. In this scheme, each soldier is viewed as a node and router at the same time, providing interconnection among all the soldiers in the battlefield. Thus, base stations are no longer needed.

Future wireless ad-hoc networks will be formed by a collection of nodes creating a spontaneous network anytime and anywhere, e.g., Bluetooth [3] and PANs (Personal Area Networks). An example of an ad-hoc network is depicted in Figure 1.2.

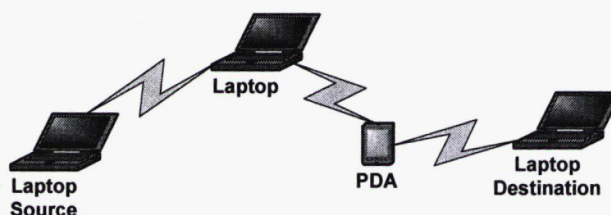


Figure 1.2: Example of Ad-Hoc Network

1.3.1 Complexity

Wireless channels are dynamic per se. Path-loss, Doppler shift, fading, and interference, are some of the problems in wireless communications [4]. Another important characteristic is the heterogeneous nature of the network. Note that an ad-hoc network can be created by different kinds of devices, each having its own capabilities such as battery power, computing power, coverage area, et cetera. Thus, due to these characteristics ad-hoc networks are a big challenge for researchers and developers. Applications, protocols, and services have to provide rapid adaptation to overcome the constant changes in the network induced by mobility.

1.3.2 Applicability

The principal application of MANETs is under hostile scenarios where base-stations or centralized infrastructure cannot be deployed, such as battlefield operations, disaster recovery, and search and rescue operations. The evolution of the information society has changed the manner in which individuals perform daily activities. Mobility also allows the utilization of multiple services from different providers, thus, interaction among different wireless networks or devices has become imperative.

MANETs allow the inter-connection of wireless devices to exchange information among individuals or organizations. Figure 1.3 depicts a wireless ad-hoc network formed by devices with different characteristics.

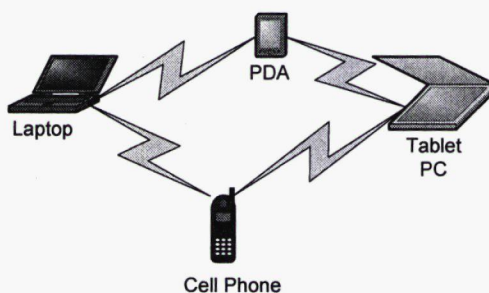


Figure 1.3: Inter-connection of Devices

Currently, lots of wireless networks such as IEEE 802.11 and cellular networks co-exist, providing several benefits to the users. Examples of such benefits include Internet access from coffee-shops, restaurants, airports, and access to multiple services such as weather forecasts, stock information, news, et cetera from cellular phone companies. Despite the fact that the architecture of these networks is completely different, they have something in common. They rely on the frequency spectrum and electro-magnetic waves to communicate. Thus, wireless networks can inter-operate to improve capacity, performance, and services. Figure 1.4 depicts wireless network interaction.

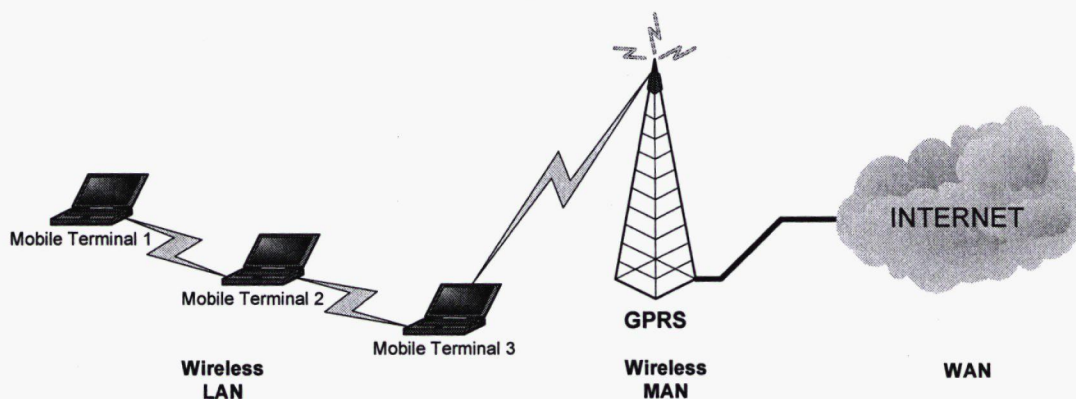


Figure 1.4: Wireless Network Interaction

1.4 Quality of Service

QoS defines a set of service requirements for a traffic flow (such as bandwidth, delay, loss, and jitter) to be met by the network [5]. The importance of QoS is determined by the

application. Applications need different levels of quality (e.g., real-time video is delay-sensitive, web surfing is delay-tolerant), hence the provisioning of each application is completely different. In order to support QoS provisioning over wireless channels, it is necessary to implement a complete QoS architecture constituted by the following mechanisms: admission control, resource management and load (congestion) control. Once the architecture is completed, we can provide acceptable levels of QoS in communication links.

1.4.1 Quality of Service in MANETs

QoS provisioning in MANETs is the network's ability to provide resources to support traffic flow's service requirements throughout the duration of the flow.

The dynamic nature of MANETs makes it challenging to provide QoS efficiently. Thus, flexible QoS provisioning schemes are needed. Consequently, several solutions have been proposed to address QoS support in MANETs [8] [9] [10]. Notwithstanding, most of these previous solutions are only suitable under specific conditions of mobility, node density, data traffic, et cetera. This section summarizes the different approaches utilized for QoS provisioning in MANETs.

A. DiffServ

Differentiated Services (DiffServ) provides a limited number of aggregate classes to differentiate the data traffic in the network [6]. This approach relies on the field TOS (Type of Service) in the IP header, also called DS (Differentiated Services) field, and on a base set of packet forwarding rules, called Per-Hop-Behavior (PHB). In this approach every router performs classification, marking, policing, and shaping once traffic differentiation is required. Thus, once a data packet with the DS field enabled is detected, each router forwards that packet based on the rules already established for that specific type of service. Table 1.1 depicts how services are differentiated based on QoS requirements.

Table 1.1: Service Differentiation

Type of Service	Packet Loss	Delay	Jitter	End-to-end bandwidth
Premium Service	Low	Low	Low	Assured
Assured Service	Flexible	Flexible	Flexible	Guaranteed or at least expected
Olympic Service	Decreasing Quality	Decreasing Quality	Decreasing Quality	Decreasing Quality

As indicated in Table 1.1, Premium Service requires strictly low levels of loss, delay, and jitter, and also assured bandwidth. Hence, supporting Premium Service in MANETs is almost impossible. Assured Service is suitable for MANETs due to the fact that the QoS requirements are flexible for all the metrics. Hence, this characteristic is good for applications that require better reliability than Best Effort Service, which does not take into account any QoS requirements. The goal of assured service is to provide guaranteed or at least the expected throughput for applications. Furthermore, the flexibility in the QoS for assured service allows it to operate under dynamic environments (e.g. the wireless channel). Olympic Service offers three tiers of services: Gold, Silver and Bronze with decreasing quality based on the requirements of each tier.

DiffServ may be a possible solution for QoS provisioning in MANET because it is not a complex model per se. In addition, it provides Assured Service without the need of signaling protocol which minimizes the cost of signaling within the network. However, DiffServ was originally created for fixed networks and modifications must be performed to adapt it to MANETs. DiffServ utilizes two types of routers: boundary (edge) and interior (core). In MANETs there is no way to predefine these routers hence every node should act as boundary and interior depending on the situation. The other challenge is the concept of SLA (Service Level Agreement) which is a contract between the customer and the Internet Service Provider that specifies the type of services the user will receive. In other words, in the Internet if a node does not have an SLA, it cannot receive

differentiated services. In MANETs there is no centralized infrastructure that negotiates the traffic rules hence DiffServ is not recommended. Nevertheless, hybrid approaches combining DiffServ and IntServ are currently used for QoS provisioning in MANETs [8].

B. IntServ

The idea of Integrated Service (IntServ) is that *state information* about specific flows is stored in every IntServ-enabled router [7]. A flow is a session between two end users. The states kept in routers include bandwidth requirements, delay bound, and cost of the flow. This approach also proposes three different classes of service: Best-Effort, Guaranteed Service which is provided to applications requiring fixed delay bound, and Controlled Load Service for applications requiring reliable and enhanced Best-Effort Service (suitable for MANET, due to the flexibility).

The implementation of IntServ relies on four components: the signaling protocol which is the Reservation Protocol (RSVP), admission control routine, the classifier, and the packet scheduler. It also relies on a routing protocol generally provided by the router and a management agent, also provided by the router in advance. Note that the routing protocol and the management agent can be changed if required.

IntServ was originally designed for fixed networks and it is not suitable for MANETs due to the following constraints:

1. Keeping flow state in each node may imply large storage and processing overheads depending on the number and the duration of the flows.
2. The utilization of RSVP signaling packets consumes bandwidth in MANETs. Signaling overhead also increases as the network becomes more dynamic.
3. Every node must do admission control, classification, scheduling, and routing. This can place heavy demands on the resource-limited nodes in MANETs.

In spite of all these constraints, the main idea of the IntServ approach can be taken and modified to be supported in MANETs. Thus, the routing process can operate with a QoS

routine to provide QoS routing without the need of a separate signaling protocol such as RSVP.

C. FQMM

A Flexible QoS Model for MANETs (FQMM) considers the characteristics of MANETs and proposes a hybrid approach based on IntServ (per-flow service granularity) and DiffServ (service differentiation) [8].

FQMM defines three types of nodes: ingress, interior, and egress. An ingress node is the source node that sends data through the network. Interior nodes forward data for other nodes, and the egress node is the destination node. In this approach, the role of each node is defined by its position and the network traffic in the network.

QoS provisioning in FQMM is performed on a per-flow basis for a small portion of the traffic in MANET, given that a large amount of traffic belongs to per class granularity. Ingress nodes also act as conditioner that is responsible for re-marking traffic streams, admission control according to the traffic profile, et cetera.

FQMM was the first approach proposed for QoS support in MANET. However there are some problems that must be solved. Scalability poses a big challenge for this approach due to the fact that there is no control over the number of services with per-flow granularity. Interior nodes rely on PHB (Per Hop Behavior) coded in the DS field, and coding that information in highly dynamic networks is not easy to achieve. Finally, FQMM per se does not support dynamically negotiated traffic profile.

D. INSIGNIA

INSIGNIA is an in-band signaling system for QoS provisioning in MANETs [9]. This can be considered the first signaling protocol designed specifically for MANETs. The signaling control information is conveyed within the data packets, to be specific, in the IP option field of every IP data packet. INSIGNIA operates on a *per-flow* management fashion. The state information for each flow is modified in response to any topology change or end-to-end quality of service condition.

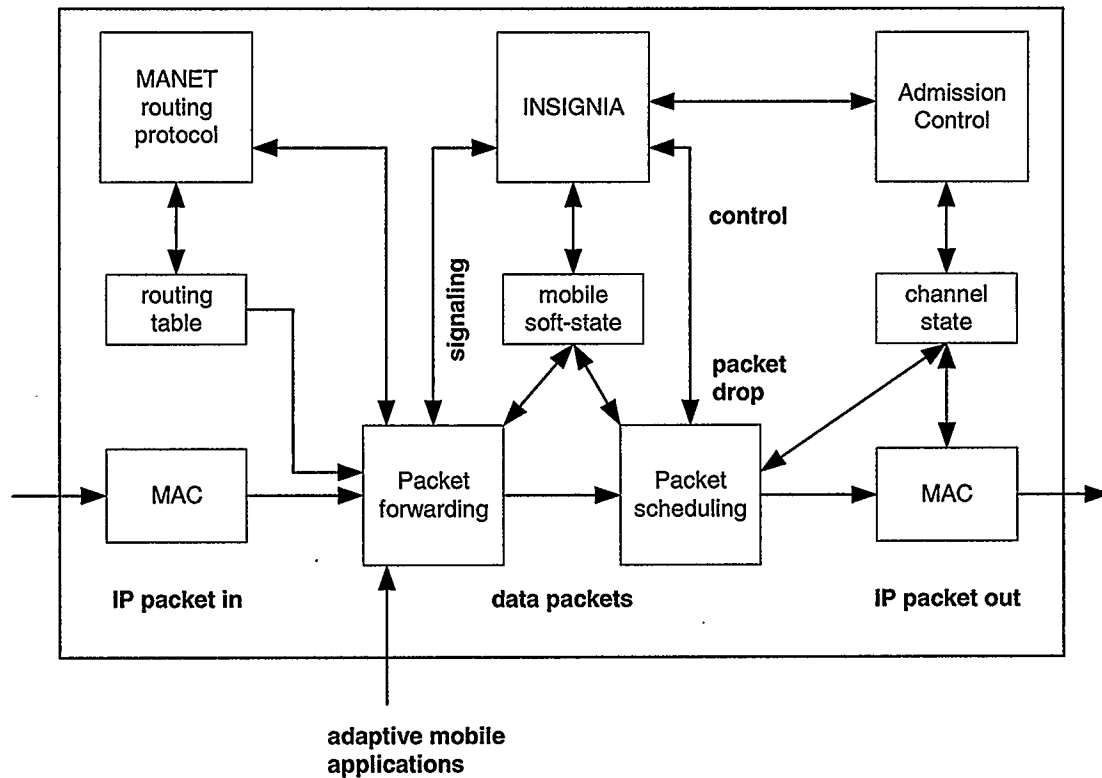


Figure 1.5: INSIGNIA Wireless Flow Management Model at a Mobile Node [9]

As illustrated in Figure 1.5, INSIGNIA relies on a routing protocol and MAC mechanisms for QoS provisioning. Thus, the packet forwarding module classifies the incoming IP packets and forwards them to the appropriate modules (INSIGNIA, routing, local applications, and packet scheduling modules). If the data packet includes the INSIGNIA option, the control information is forwarded to the INSIGNIA module for processing. If the node is the destination the packet is processed by a local application. If the node is just an intermediate, the next hop is determined by the routing protocol. All the packets must be scheduled before transmitting them to the MAC layer.

The INSIGNIA module is responsible for establishing, restoring, adapting, and releasing real-time flows. INSIGNIA allocates resources to the flow only if the Admission Control module allows it. To maintain low signaling overhead within the network, INSIGNIA does not send error messages if the resources cannot be allocated.

As a whole, INSIGNIA is an effective signaling protocol for QoS provisioning in MANETs. The main drawback is that it relies on a routing protocol to operate. Hence, poorly designed routing protocols can affect the overall performance of MANETs even though INSIGNIA shows good performance.

E. CEDAR

Core-Extraction Distributed Ad-hoc Routing (CEDAR) was the first routing protocol designed for QoS support in MANETs [10]. This protocol includes three components: core extraction, link state propagation, and route computation.

1. Core Extraction

The objective of core extraction is to elect a set of nodes to form a core of the network. The core of the network is built by an approximation of a minimum dominating set [10] of the network using only local computation and local state. Every node not in the dominating set selects one of its neighbors in the dominating set as its *dominator*. Every node in the dominating set is called a core host. Therefore, the dominator of a core host is itself. Two core hosts are called nearby core hosts if the distance between them is no more than 3 hops. The path joining two core hosts is called a *virtual link*. The graph resulting from core nodes and virtual links connecting nearby core hosts is called a *core graph*. A *core path* is a path in the core graph.

CEDAR relies on a distributed algorithm to elect core nodes. Once a node loses connectivity with its dominator due to mobility, it elects a new core neighbor as its dominator. If there is no core neighbor nearby it nominates a non-core-neighbor to join the core or itself to join the core. Further details of the core extraction protocol are given in reference [10].

Flooding the network with redundant broadcasts consumes substantial percentage of bandwidth in the network. CEDAR utilizes core broadcasts (broadcast performed only among core nodes) to minimize routing overhead within the network and adapts efficiently to topology changes.

2. *Link state propagation*

In order to provide feasible QoS routes in CEDAR, each core maintains its local topology as well as the link-state information of high-bandwidth links further away. Core nodes do not keep information about unstable or low-bandwidth links due to the fact that these links are not suitable for QoS provisioning. To maintain reliable routes, CEDAR utilizes an approach called increase/decrease waves. Once a host detects an unstable link it communicates with its dominator. The dominator releases a core broadcast for a decrease wave, which indicates the unstable link. On the other hand, when a node detects a stable link, its dominator releases a core broadcast for an increase wave to announce the stable link. The difference between these waves is the speed at which they travel within the network. A decrease wave is propagated faster than an increase wave; this causes the fast-moving wave to over take the slow-moving wave. At the end of this process, the increase wave propagates the stable high-bandwidth link state information through the cores. CEDAR also provides mechanisms to avoid the propagation of the decrease wave through the whole network. Thus, the unstable low-bandwidth link states are kept locally.

3. *Route computation*

QoS route computation in CEDAR includes three steps: (a) route discovery and establishment of the core path, (b) searching for a stable QoS route within the core path already established, and (c) error recovery (dynamic QoS route re-computation) upon link failures or topology changes.

Route discovery is performed among dominators within the core, and QoS route selection is performed based on the local link state information kept in the dominators. CEDAR deals with link failures by two mechanisms: dynamic re-computation of a feasible route at the point of failure (local repair), and notification back to the source to re-compute the route from the source. These mechanisms are used to respond to topology changes.

It has been shown by simulation that CEDAR can compute feasible routes with high probability and adapt efficiently with low routing overhead to highly dynamic MANETs [10].

F. SWAN

Service Differentiation in Stateless Wireless Ad-Hoc Networks (SWAN) uses distributed control algorithms to deliver service traffic differentiation in MANETs. SWAN uses rate control for UDP (User Datagram Protocol) and TCP (Transmission Control Protocol) best-effort traffic, and source-based admission control for UDP real-time traffic [11]. Explicit congestion notification (ECN) regulates dynamically admitted real-time traffic to overcome the dynamic nature of MANETs (i.e., topology changes, and congestion levels). SWAN is also designed to support real-time services over best-effort MACs without the need to install and maintain QoS states at MANET nodes. This makes this approach simple, scalable, and robust for QoS provisioning.

SWAN uses feedback information for the network to detect changes in the topology or QoS levels instead of depending on state information. This minimizes the overhead produced by QoS information exchange. Admission control is a key element for QoS provisioning. Thus, SWAN places an admission controller at every node to estimate efficiently local bandwidth availability. Once congestion is detected through lower layer mechanisms, ECN analyses the problem and proposes an efficient solution based on the information obtained from the network.

1.5 Routing

A network is a collection of nodes interconnected by links for the purpose of sharing information or devices on the network. The transmission of packets is performed peer to peer when the nodes are interconnected. Interconnection can be achieved through different topologies: e.g., star, bus or ring. Hence, routing is required when a node tries to transmit a packet to another that is located more than one hop away. In this case the node has to know a path to the destination, and keep this path up to date. The aforementioned tasks can only be performed with the aid of routing protocols. Routing protocols for fixed networks are not suitable for wireless ad-hoc networks because of node mobility and the dynamic wireless channel. Constant changes in the wireless channel and node mobility modify the network topology. Convergence, an important characteristic in routing

protocols, is the time required for the routing table to stabilize (i.e., find an alternate route to the destination) following changes in the network topology. Fast convergence is important in MANETs. For this reason, routing protocols specifically for MANETs have to be designed. Figure 1.6 and Table 1.2 illustrate how routing information is processed and stored as a routing table.

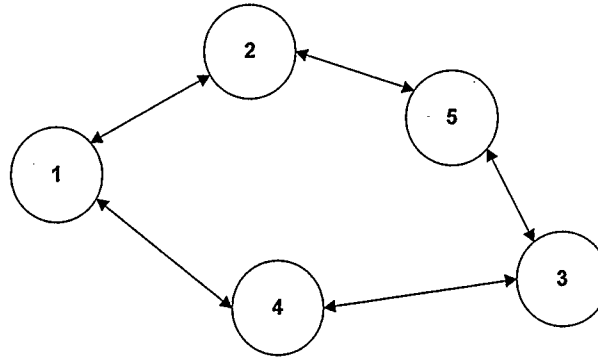


Figure 1.6: Routing Scheme for Source Node # 1

Table 1.2: Routing Table for Source Node # 1

Destination	Next Hop	Cost
2	2	1
3	4	2
4	4	1
5	2	2

Node 1 maintains a table with routing information regarding all the nodes within the network (obtained through continuous routing updates or through route request packets). The routes in the routing table are stored based on a routing metric. The routing metric is a selection criterion that ensures the routes stored are the best. In this example, the field Cost specifies the number of hops (i.e., routers) a data packet must traverse to reach the destination. Thus, according to the routing metric, Table 1.2 only stores the shortest path to the destination. If the source, in this case node 1, wants to send information to the destination node 3, it tries to find a route to the destination in its routing table. Once the route is found, node 1 initiates data transmission through the node specified in the Next Hop field. Next Hop field stores the node (i.e., router) used to reach the destination.

1.5.1 Routing in MANETs

Routing can be performed through the use of routing tables. These tables help the routing protocol choose the best route to a destination based on a metric defined a priori. The way the routing protocol creates, propagates, and maintains routing tables classifies the routing protocol [12]. In MANETs there are three types of routing protocols: proactive, reactive, and hybrid. Proactive routing protocols create and maintain the routing tables before they are needed. On the other hand, reactive protocols obtain routing information only when the node needs it. Hybrid protocols combine characteristics of both approaches.

A. Proactive Routing Protocols

Proactive routing protocols always maintain routing information regarding all the nodes in the network. Hence, continuous routing updates must be performed to avoid stale routes. These protocols produce more routing overhead than reactive protocols, but the advantage is the fast routing convergence in the presence of topology changes. DSDV [24] is an example of a proactive routing protocol for MANET.

B. Reactive or On-demand Routing Protocols

Obtaining routing information on demand saves memory in each node and reduces the overhead produced by continuous routing updates. Therefore, global knowledge of the network is not required.

In source routing protocols such as DSR [13] the routing information is conveyed within the routing packets. Route request packets append the IP address of each hop along the path, thus when the route request packet reaches the destination or an intermediate node with a valid route to the destination, the node receiving the route request packet (i.e., destination node or intermediate node) automatically reverse the route (to the source) and initiates the route reply process. Once the route reply packet (conveying the entire route) reaches the source, the source is ready to initiate data transmission. Protocols based on source routing show better performance in terms of

convergence time over reactive protocols [14]. The trade-off is the routing overhead (in bytes) caused by conveying the entire route (multiple IP addresses) in routing packets [14].

C. Hybrid Routing

Hybrid routing protocols combine proactive, reactive, and/or hierarchical approaches. Zone Routing Protocol [15] is an example of a hybrid routing protocol. Routing protocols based on hybrid routing are not very popular due to their complexity.

Routing can be performed in different ways according to the distribution of the nodes in the network. Flat routing is performed within a network where all the nodes are in the same domain. Hierarchical routing defines three types of nodes: gateway, cluster-head, and common nodes [16]. The network is viewed as a cluster containing one cluster-head, one or more gateways, and multiple common nodes. The cluster-head is the controller of the network. It has the routing tables and each packet has to pass through it. This approach produces two kinds of routing: intra-domain and inter-domain. The former is performed within the nodes in a cluster. The latter is performed only among cluster-heads. The gateway is a node within the range of multiple wireless networks. It is in charge of the network interconnection. Common nodes are just terminals transmitting and receiving packets. Figure 1.7 illustrates an example of a network utilizing hierarchical routing.

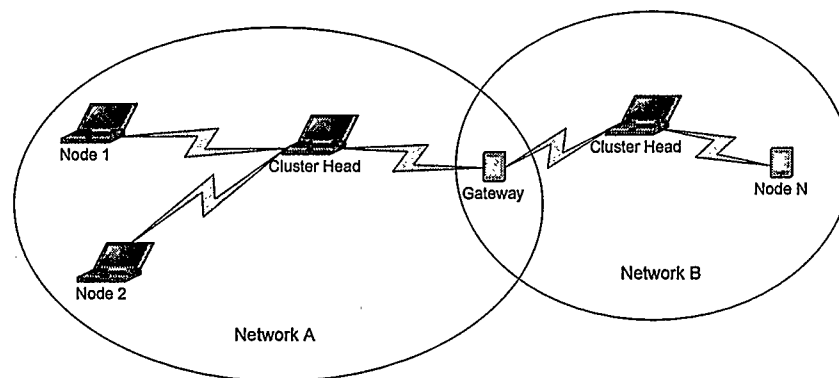


Figure 1.7: Hierarchical Routing

1.6 QoS Routing in MANETs

The routing process encompasses discovering and maintaining one or multiple paths (routes) for transporting a data flow from a source to the destination(s). In the traditional routing protocols for MANETs, the most common routing metric used for route selection is shortest path (i.e., minimum number of hops). On the other hand, for QoS routing in MANETs, the routing metric must be defined by the QoS requirements of the flow. As such, QoS routing in MANETs is defined as the problem of selecting a path based on the flow's QoS requirements and some knowledge of the available resources (i.e., bandwidth). This means that each route found must meet at least the minimum QoS levels required by applications. If multiple routes are obtained, a single route is then selected based on the predefined QoS routing metric (e.g., bandwidth, delay, jitter, loss, et cetera). Hence, unlike the traditional routing in MANETs, in QoS routing sometimes the shortest path is not the best route to the destination.

The QoS routing process encompasses the following tasks. The first task is to obtain a route that meets the QoS requirements specified by the application. Examples of route-finding techniques include proactive and reactive routing approaches, as described in section 1.5.1. The second task involves selection of a route in case multiple routes are discovered. The final task addresses reservation of resources along the path selected if required. This task is required in order to guarantee the QoS constraints indicated by the applications. In MANETs, an efficient QoS routing is necessary to provide reliable QoS support in the presence of continuous topology changes. The challenge is to achieve a good balance between performance (e.g., routing efficiency, QoS guarantee) and network overheads (e.g., routing overhead, power consumption). Fast topology changes can result in network instability: the propagation of topology update information is not completed before the next topology change. Consequently, QoS routing in MANETs is challenging due to the difficulty of maintaining up-to-date routing and QoS information. Solution to the QoS routing problem in MANETs is therefore of active research interest in the literature. Several QoS routing algorithms have been proposed, see for example, [17] – [20]. The algorithms are characterized by the assumed QoS metric for path computation, mechanism for propagating and maintaining network state, routing strategy, route

computation strategy, routing architecture and whether or not multiple paths are maintained. A comparison of the characteristics of the existing QoS routing algorithms for MANET is presented in [21].

1.7 Thesis Contributions

This thesis contributes to QoS support in campus area MANETs by proposing a QoS routing framework that provides QoS functionality to the routing process regardless of the underlying routing protocol. The QoS routing framework proposed in this thesis applies only to bandwidth.

Following the specifications of the QoS routing framework, the performance of the routing process in terms of average route discovery time, routing overhead, and power consumption is improved. The main difference between the proposed QoS routing framework and related work on QoS for MANET is that the framework is an architecture that enables QoS support in routing protocols that do not support it originally and does not rely on signaling mechanisms to provide QoS.

An OPNET-based computer simulation tool is developed to evaluate the proposed QoS routing framework. Numerical results are presented for the original NIST (National Institute of Standards and Technology) / AODV routing protocol [33] and the modified version for QoS support using the proposed framework. The original and the modified routing protocol (based on QoS routing the framework) were evaluated under the same conditions of mobility and congestion to demonstrate the improvements achieved.

1.8 Thesis Overview

The remainder of the thesis is organized as follows. Chapter 2 presents a QoS routing framework for MANETs and justifies the importance of such a framework in QoS provisioning in MANETs. The QoS routing framework proposed in Chapter 2 must be evaluated to determine its suitability. Hence, Chapter 3 introduces the simulation environment utilized to analyze the performance of the QoS routing framework under different scenarios of mobility and traffic loads. The results obtained through

mathematical analysis and computer simulation are presented in Chapter 4. Finally, Chapter 5 presents the conclusions of the thesis and suggestions for future work.

Chapter 2

A QoS Routing Framework For MANETs

2.1 Introduction

The aim of this chapter is to explain the functionality of a QoS routing framework and to explain briefly the underlying approaches that constitute it. Basically, the QoS routing framework relies on four mechanisms: routing overhead reduction, QoS route discovery, QoS robustness enhancement, and QoS route maintenance. Routing overhead reduction minimizes the impact of routing and QoS information exchange in the MANET. QoS route discovery is in charge of discovering routes that meet at least the minimum QoS requirements to support applications efficiently. QoS robustness enhancement tries to maintain the QoS level supported for the duration of the data flow, regardless of changes in the network topology. QoS route maintenance addresses congestion control, QoS resource release, and maintaining valid QoS routes within the network. Cooperation between processes is necessary to achieve better efficiency in terms of routing and QoS assurance. Figure 2.1 illustrates this relationship.

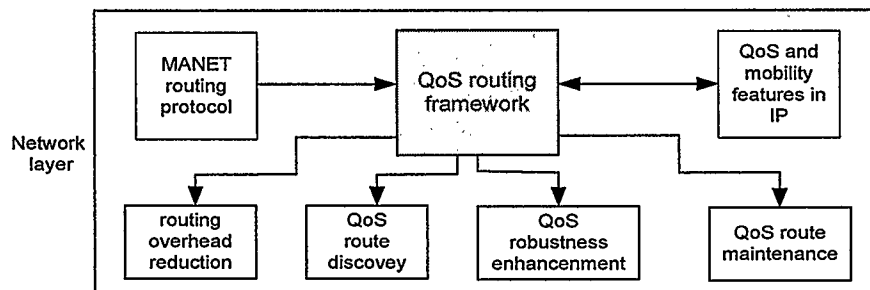


Figure 2.1: QoS Routing Framework

QoS routing addresses QoS support from the network layer of the OSI model (Figure 2.1). However, inter-layer cooperation is necessary to provide full and robust QoS support in dynamic networks such as MANETs [22]. Figure 2.2 illustrates the cooperation among processes within the application, transport, network and medium access control protocol layers to address QoS support in MANETs. Routing protocols for MANET rely on the QoS routing framework to enable QoS support. At the same time the framework relies on different mechanisms such as routing overhead reduction, QoS route discovery, QoS robustness enhancement, and efficient route maintenance mechanisms. Efficient QoS support in MANETs cannot be achieved without inter-layer cooperation. Feedback from upper and lower layers facilitates information exchange about applications types, QoS levels, accurate detection of broken links, et cetera. Thus, applications, protocols and processes can adapt to changes in network topology. Flexibility and adaptability are important to support multimedia and real-time applications over highly dynamic environments, such as MANETs. The focus of this thesis is, however, on the QoS routing framework defined at the network layer.

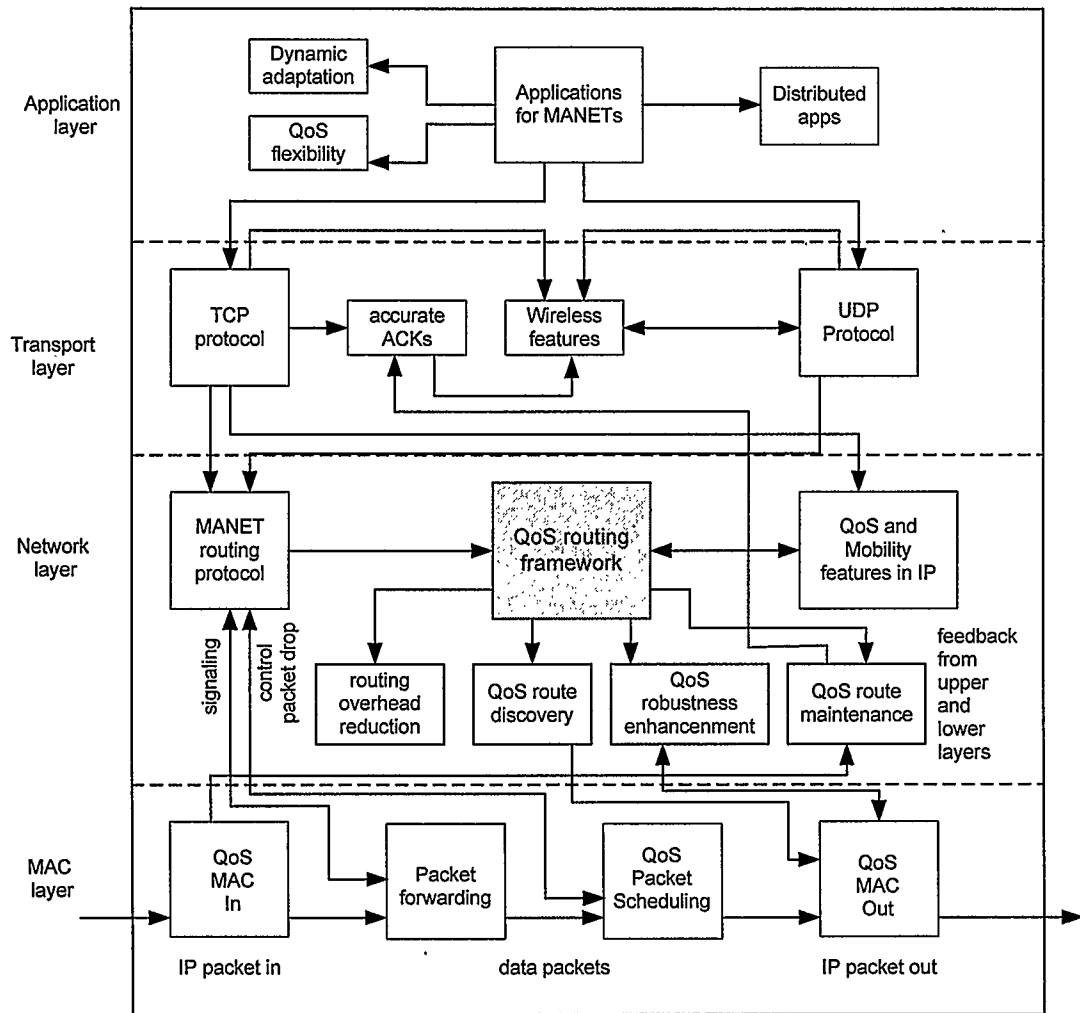


Figure 2.2: Interaction between OSI Model Layers to achieve Full QoS support in MANETs

2.2 QoS Routing Issues

The dynamic nature of MANETs poses many issues that make the QoS routing process quite challenging. This section provides a review of the most important issues associated with QoS routing in MANETs.

2.2.1 QoS Routing Overhead Issue

Routing overhead is the routing packets (i.e., route request, route reply, and route error packets) required to compute routes, calculate network and QoS state information in the network nodes, and maintain the routes when the network topology changes. Routing

overhead becomes an issue in MANETs due to redundant route computation (i.e. stale routes or duplicate routes) and excessive routing packets. The manner in which routing protocols obtain and update routing information is directly related to the routing overhead in the network. The challenge then is to devise QoS routing protocols that perform routing updates with minimum routing overhead and accurate routing decisions.

2.2.2 QoS Route Discovery Issue

The route discovery process establishes a logical connection between the source and destination nodes. This process can be performed by the existing routing protocols for MANETs such as dynamic source routing (DSR) [13], ad-hoc on-demand distance vector (AODV) [23], destination-sequenced distance vector (DSDV) [24], zone routing protocol (ZRP) [16], and others. However, as stated earlier, QoS route discovery implies that a feasible route is found only when the QoS requirements for a traffic flow are satisfied. The need to support multimedia applications with different and conflicting requirements makes the QoS route discovery process difficult to achieve. The first issue pertains to determining the QoS metric (bandwidth, delay, jitter, packet loss rate, et cetera). However, with multimedia applications, QoS route discovery becomes more complicated when multiple QoS metrics are required. In fact, QoS route discovery based on multiple QoS constraints has been found to be a non-deterministic polynomial (NP) complete problem [25]. In order to support QoS efficiently, resources such as bandwidth have to be reserved to guarantee the QoS for the applications. There is the issue of selecting the right reservation protocol that is scalable and with minimal overhead for reserving the resources in a MANET with dynamic topology. Another issue is that of determining the order of performing the route discovery and QoS reservation for efficient QoS routing. Some researchers have proposed that the route discovery process be performed first, followed by resource reservation. This is the underlying approach for INSIGNIA [9].

2.2.3 QoS Robustness Issue

QoS robustness is in charge of guaranteeing the allocated resources (i.e., bandwidth) regardless of MANET topology changes [26] if possible. For MANETs with multimedia

support, QoS robustness is an important issue because of the need to guarantee the QoS for real-time applications (e.g., delay and delay jitter for video and voice) during any topology updating. Also, if the frequency of topology change is too high due to fast mobility and high frequency of link failures, the time between topology changes may become less than the time to find the new route and propagate topology update information to all the nodes in the network, resulting in network instability. Another issue pertains to the difficulty of simultaneous support of the QoS requirements of different application types. Making the network QoS-robust for a particular application can degrade the performance of the other traffic types in the network. It is therefore paramount to consider QoS robustness important for assessing the suitability of a QoS routing protocol for multimedia support in MANETs.

2.2.4 QoS Route Maintenance Issue

The route discovery process is in charge of finding routes that meet at least the minimum QoS constraint specified by an application. Mechanisms are therefore required to maintain these routes throughout the duration of a connection. Such mechanisms include congestion control, resource release mechanisms, and multiple route support. The process of monitoring and releasing resources raises a number of issues. A first issue is the bandwidth consumed by the signaling to perform resource monitoring, release and information update at the nodes in the network. A second issue is a consequence of the distributed nature of MANETs: since it takes a finite amount of time for information to travel from one point to another, the information about resource availability at the different nodes in the network will not be up-to-date.

Congestion control assures the resources are not over-utilized. Poorly designed admission control algorithms can lead to over-utilization of network resources such as bandwidth. The dynamic nature of MANETs and the different characteristics of the nodes forming the network (i.e., data rate) make congestion control more complex. Hence, each node must be able to control its own resources and perform allocations if the resources are available. Otherwise, the resource allocation must be denied to avoid traffic congestions in the network.

Multiple route support allows the utilization of back-up routes once an error in the network is detected (i.e., link partitions due to mobility or hidden nodes). The back-up route selection process is an issue due to the changing topology of MANETs. It is necessary to maintain accurate information about all the routes in the routing table, including of course, the back-up routes. A poorly designed back-up route selection process can increase the convergence time of the routing protocol and the routing overhead within the network.

2.3 The QoS Routing Framework

2.3.1 Assumptions

As a prelude to presenting the elements for the proposed QoS routing framework, we state the key assumptions made in this thesis:

1. The routing protocol evaluated in this thesis is AODV (Ad-Hoc On demand Distance Vector).

Justification: Assumption 1 is made due to the routing efficiency exhibited by AODV [14] and the continuous updates to the routing protocol.

2. QoS routing is performed within a campus area network formed by 30 nodes with symmetric links.

Justification: Symmetric links facilitate the routing process and QoS provisioning. Campus area network and the number of nodes are assumed for simplicity.

3. The nodes in the network are highly active in terms of routing information exchange due to continuous link partitions (i.e., mobility and/or hidden nodes).

Justification: Assumption 3 is made due to the need of new routes and the need to overcome link partitions.

4. All the nodes have the same characteristics in terms of memory, computing resources and transmission power.

Justification: This is to ensure homogeneity of the network nodes to simplify the computer simulation.

5. The QoS metric utilized to evaluate the proposed QoS routing framework in this thesis is bandwidth. Other QoS metrics besides bandwidth are not supported in this thesis.

Justification: Bandwidth was selected as a routing metric to simplify the simulation process. Multiple QoS metrics are not supported to simplify the route discovery process.

6. The source node cannot change its QoS demand during the lifetime of the call. The lifetime of the call is assumed to be equal to the simulation time (1800 seconds).

Justification: Assumption 6 is made to simplify the simulation process.

7. Each node in the network supports a maximum of two back-up routes (3 routes per node including the primary).

Justification: Assumption 7 is made to reduce the back-up route computation time.

The proposed framework is described in terms of solutions to address the QoS routing issues taking into account the assumptions of Section 2.3.1

2.3.2 Routing Overhead Reduction Techniques

As noted earlier, QoS support introduces signaling overhead (i.e., control packets) that are associated with establishment and maintenance of QoS-based routes. The challenge is to keep the overhead as low as possible during the QoS routing process so as to optimize bandwidth utilization. In the following, we present some techniques for reducing the QoS routing overhead.

1. Embedding QoS Information within the Routing Packets

For efficient bandwidth utilization, the routing process must ensure that QoS information exchange is performed by reducing the amount of QoS signaling packets generated in the network. We propose to obtain QoS routing information by embedding the QoS constraint as part of the routing packets generated at every call setup. Under this condition, it is no longer necessary to transmit separate QoS signaling packets. This approach provides savings in extra routing overhead.

2. Reactive Routing Approaches

Proactive routing protocols such as DSDV [24] propagate keep-alive messages periodically to maintain the routes, resulting in considerable routing overhead that affect the QoS support in MANETs [27]. On the other hand, continuous routing updates are not required for the reactive approaches (e.g., DSR [13], AODV [23]). Hence, reactive protocols are better than the proactive protocols in terms of routing overhead reduction. However, a trade-off exists in the sense that convergence time is longer for reactive than for proactive routing approaches.

3. Hop-by-hop Routing

DSR and AODV routing protocols belong to the reactive approach. In principle, both AODV and DSR protocols use the same route discovery approach. However, AODV incurs lower routing overhead (in bytes) than DSR [14]. The lower routing overhead of AODV is due to its use of hop-by-hop routing while creating a path instead of conveying the entire route within the routing packet like DSR. Hence, the use of hop-by-hop routing mechanism reduces the routing overhead (in bytes).

4. Expanding Ring

Expanding ring is a two-step process that minimizes the routing overhead produced by controlling repetitive broadcasts of routing packets [23]. Before transmitting a routing packet, the source node first sets the number of hops the packet can traverse to *zero*. Intermediate nodes receiving the routing packet (whose hop count is set to zero) will not

propagate it further. If an intermediate node with a valid route to the destination is found, then the node sends a route reply packet to the source. Otherwise, the source must re-transmit the routing packet but this time allowing the intermediate nodes to propagate the packet until the destination is reached or until the number of hops traversed reach the net diameter specified by the routing protocol (e.g., NIST / AODV defines 15 hops as maximum net diameter). The high likelihood of finding an intermediate node with a valid route to the destination therefore reduces the routing overhead.

2.3.3 QoS Route Discovery Approaches

Efficient QoS route selection algorithms are required to minimize the routing overhead, and enhance the routing efficiency in terms of reduced average route discovery time. A scheme that satisfies all the aforementioned requirements may be difficult to find. Furthermore, in a case when multiple routes with the same QoS level are found, this must be addressed using tie-breaking metrics. An example is illustrated in Figure 2.3 where it is required to select a route that satisfies the QoS objective of 2 Mbps. It is seen from Fig. 2.3 that the three routes 1, 2 and 3 meet the 2 Mbps objective. Next, the shortest-path constraint (used here as the tie-breaking metric) is then imposed which elects route 1. The above process can be generalized: first, use the QoS metric (i.e., bandwidth) to determine multiple feasible routes. Next apply each of the remaining tie-breaking metric(s) one at a time until a route that meets all the QoS constraints is found, this route is then selected.

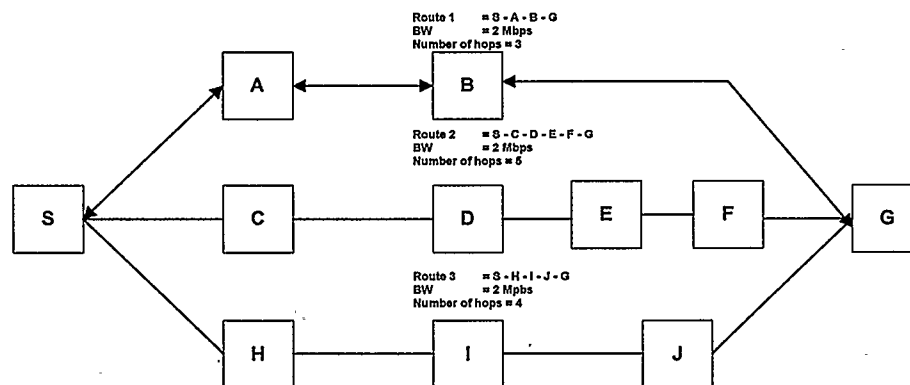


Figure 2.3: Tie-breaking metrics for route selection

2.3.4 QoS Robustness Enhancement Ideas

We propose to enhance QoS robustness using two mechanisms: dynamic adaptation, and specification of QoS requirements as a range instead of a single value.

1. Dynamic Adaptation

The best way to deal with dynamic systems such as MANETs is through dynamic protocols and algorithms, and adaptive applications. Furthermore, instead of the traditional network protocol layering design practice that assumes independence of the networking protocol layers, efficient QoS support can be achieved and maintained in future MANET protocol design by taking advantage of the interaction among the protocol layers [22]. As an illustration, Figure 2.2 shows the QoS mechanisms operating at the data link, network and application layers. First, at the data link layer, medium access control (MAC) mechanisms must regulate access to the channel. Clearly, knowledge of the channel condition and QoS requirements of the applications are vital for design of efficient scheduling mechanisms to gain channel access. Second, at the network layer, QoS routing must find and maintain routes that meet the QoS desired by applications; call admission control (CAC) and resource management protocols must satisfy the QoS requirements of applications and also be able to map application level QoS to MAC-level QoS. Finally, at the application layer, applications that can adapt their QoS requirements based on the current conditions of the network and then perform the necessary QoS negotiation are required. For example, a Web server sends information with graphics (images or animations) and multimedia if the channel conditions allow it. Under poor channel conditions when such services can no longer be supported, the Web server can send the information in plain text, satisfying the need of the user for information despite lack of images or multimedia. Other applications, such as real-time video must change the compression or renegotiate the quality of the video transmitted to meet the QoS resources supported by the wireless channel. Thus, the applications must be flexible in terms of QoS requirements and are able to re-negotiate the QoS parameters [28].

2. Specification of QoS Requirement (Bandwidth) as a Range

Another idea for ensuring QoS robustness is by expressing numeric QoS requirements (i.e., bandwidth) as a range instead of a single value [9]. This approach offers a certain level of flexibility in case the exact bandwidth requested cannot be met. Two QoS limits are specified: minimum bandwidth (MIN) and maximum bandwidth (MAX) such that every value within the $(\text{MIN} \leq \text{QoS allocated} \leq \text{MAX})$ range is acceptable. Of course, it is better if the maximum value is met. In this thesis, bandwidth is the only QoS requirement evaluated, from the key assumptions listed in Section 2.3.1).

2.3.5 QoS Route Maintenance Techniques

This section presents approaches for maintaining QoS routes in a MANET.

1. Cooperation among Network Nodes

Cooperation among the nodes in a MANET is facilitated by information exchange to improve protocol convergence. Due to the broadcast nature of the wireless channel, certain routing information (e.g., active routes, QoS levels, errors, control messages, etc.) can be shared every time a change in network characteristics is detected.

2. Multiple routes support

The route discovery process can provide information to the source about more than one route supporting the QoS requested. Hence, the QoS routes stored in the routing table can be classified as primary and back-up routes. The primary QoS route is selected based on how fast it was acquired. The back-up routes are defined as the routes arriving to the source after the primary route was obtained (Note that all the routes received by the source meet at least the minimum bandwidth specified by the source). The maximum number of back-up routes a node can store is two based on the specifications of the QoS routing framework. A backup route is selected if the primary route fails (i.e., link partition). Through the backup routes approach, the average route discovery time (average time the nodes spend in the route discovery process) is improved considerably

[29]. The application of back-up routes based on the QoS routing framework is explained in the next section.

2.4 Application of QoS Routing Framework

This section illustrates the application of the QoS routing framework on NIST / AODV routing protocol for MANETs.

2.4.1 QoS Route Discovery

In reactive routing protocols, route discovery is performed when a node needs a route to the destination, and this route cannot be found in the routing table. Nodes deal with route requests in three different manners depending on their position in the network. Thus, the nodes are classified as:

- *Source or originator* is the node that needs the route. Hence, it is in charge of originating the route request process.
- *Intermediate* is a node that receives the route request but is not the destination (hop).
- *Destination* is the node that terminates the route request process, once the route request arrives to it.

The nodes described above perform different tasks based on the role they are playing in the network (e.g. source, intermediate, or destination). The main goal of the QoS routing framework is to enable efficient QoS functionality whether or not the underlying routing protocol supports it originally (AODV in this case). From Section 2.3.1, QoS route discovery assumes symmetric links both in forward and reverse paths.

A. Route Request Packet Structure

The route request packet is used to obtain a valid route to the destination that meets the QoS value defined by the application. These packets travel through the network gathering information about different routes to the destination. Once the information is gathered,

the source determines the route to the destination. It is important to state that the source cannot change the value of the QoS reserved during the call.

Table 2.1 depicts how QoS support can be enabled in the route request packet. To enable QoS support, we propose three extra fields must be added: *QoS RSV*, *QoS_min*, and *QoS_max*. *QoS RSV* specifies the maximum QoS value (i.e., bandwidth) supported by *all* the nodes along the route. The content of *QoS RSV* field (bandwidth requested) is modified if and only if the available bandwidth is less than its current value. We impose the constraint that all the nodes forming the route must support the same bandwidth in order to avoid packet losses caused by bottlenecks in the network. *QoS_max* and *QoS_min* specify the bandwidth requirements as a range.

Table 2.1: Route Request Packet Structure

Option Type	Flags	Hop Count
Src IP address	Dest IP address	
Src Seq Num	Dest Seq Num	
RREQ ID		
QoS RSV	QoS_max	QoS_min

B. Routing Table

For table-driven routing protocols, routing tables are important elements in the routing process. Table 2.2 depicts a generic routing table which contains the minimum information required for routing protocols.

Table 2.2: Routing table structure illustrating modifications needed for QoS support

Dest	Next hop	Seq. num	Num. of hops	Flags	QoS	Stability

The fields in Table 2.2 store different route information about specific nodes in the network. Maintaining up-to-date information in the routing table improves the overall efficiency of the routing process and avoids the formation of *loops* produced by stale routing information. *Dest* stores the IP address of the destination node. *Next Hop* indicates the identity of a neighboring node that knows a route to the destination. Hence, data packets must be forwarded to that node. *Seq. num.* (*Sequence number*) avoids stale routing information and continuous re-broadcasts of the same RREQ packet. This also operates as the serial number of each RREQ packet. *Num. of Hops* describes how far the destination is with respect to a specific node. This field is useful when the primary routing metric is *number of hops*. *Flags* field indicates the status of the route: whether it is available, broken and waiting for repair, or if it has to be deleted from the routing table. In addition, two extra-fields must be added to the routing table to enable QoS support. These fields are *QoS*, which specifies the amount of bandwidth allocated (supported) for a route and *Stability* which serves as a tie-breaking metric based on how active (in terms of routing information exchange) a node is with respect to another.

C. Route request sent repository

Route request sent repository is a table that stores a copy of the route request packets that have not been yet answered by route reply (RREP) packets (for re-transmission purposes). Table 2.3 depicts the structure of a route request sent repository.

Table 2.3: Route Request Sent Repository Structure

Dest	Attempts	QoS

Originally, this structure contains two fields: *Dest* field which stores the IP address of the destination node for which the route is requested. The *Attempts* field is a counter of how many times the node has re-transmitted the RREQ packet. NIST / AODV specifies a maximum of 2 route request re-transmission attempts.

Additionally, a QoS field is added to the original table to record the maximum QoS supported for that specific destination and waiting to be allocated. This route request sent repository is useful to minimize the routing overhead produced by several broadcasts of the same RREQ packet and to avoid the utilization of bandwidth that is waiting to be allocated (RREP process).

D. Route Reply Process

The route reply process is performed when a route request packet arrives at the destination. The principal goal of this process is to provide the source with a route(s) to the destination that meets the QoS value required (The source node then selects the best route in terms of the QoS metric chosen). Once the destination node receives the route request packet, it transmits back to the source a route reply packet with a valid route. The reason for allowing only the destination to initiate the route reply is twofold. First, the route reply process provides to the source a route to the destination and second it allocates the QoS value (i.e., bandwidth) required by the application. QoS allocation is performed by the route reply process due to the fact that once the route request packet arrives at the destination it carries the information about what is the maximum QoS value supported along the entire route (i.e., the contents of QoS RSV field in the RREQ packet). Based on this, only the destination is allowed to initiate the route reply process.

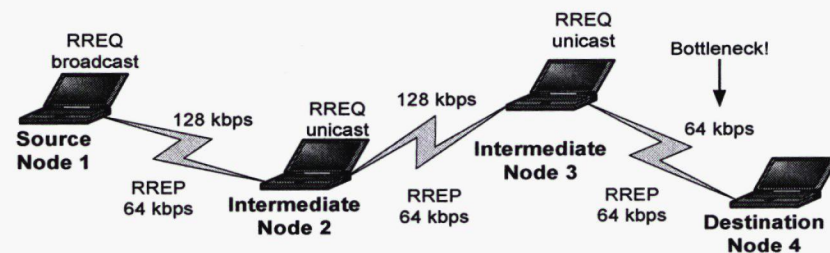
E. RREP Packet Structure

Route request packets are used to obtain route(s) to the destination and to specify the QoS level required. RREP packets are used to respond to those requirements. The RREP packet structure varies depending on the routing protocol. Table 2.4 depicts the RREP packet. To enable QoS support in the RREP packet, just one extra field is needed as illustrated in Table 2.4. This field is called *QoS* and it stores the amount of QoS (i.e., bandwidth) that must be allocated by the nodes along the route.

Table 2.4: Route Reply Packet Structure

Reserved	Type	Hop Count
Dest Seq Num		Dest IP address
Src IP address		Lifetime
QoS		

Figure 2.4 depicts the route discovery process (RREQ and RREP) based on the QoS routing framework. Table 2.5 illustrates a simplified routing table of node 2. Node 2 stores in the routing table information regarding valid routes in the network. Destination IP address, number of hops, and QoS allocated are part of the content of the routing table. The values in these fields are obtained through the route discovery process as the route request packet traverses each node within the network (e.g., node 1, node 2, node 3 until it reaches node 4). The information contained in the QoS field is obtained through the route reply packet generated by the destination node (maximum QoS supported in the network).

**Figure 2.4: Route Discovery****Table 2.5: Node 2 Routing Table**

Dest	N.H.	No. hops	QoS	
4	3	2	64	Successor (Route to destination)
1	1	1	64	Predecessor (Reverse route)

Source (node 1) broadcasts a RREQ packet asking for a route to the destination (node 4) with a minimum bandwidth of 64 kbps and a maximum bandwidth of 128 kbps. Once an intermediate node (node 2) receives the RREQ packet, it checks to see if it has a route that satisfies the QoS value specified in the RREQ packet. Based on its routing table,

node 2 knows that the destination can be reached through node 3. Hence, node 2 does not generate a RREP packet but instead unicasts (node 2 already knows the next hop to the destination hence there is no need for broadcasts) a RREQ packet through the next hop stored in its routing table to reach the destination and trigger the QoS allocation process (RREP packet). If the next hop does not support the QoS requested anymore the route discovery process is resumed by node 2 in the traditional manner (broadcasting RREQ packets). Once the destination node (i.e., node 4) receives a RREQ packet, it compares the *Destination IP address* in the packet with its own address. If the destination address matches its own IP address, this means that the destination has been reached. For the reactive routing protocol assumed in this thesis, arrival of a RREQ packet at the destination implies that a reverse route has been created in advance. Hence, node 4 utilizes a predecessor from its routing table to transmit the RREP packet to the source. The RREP packet travels along the reverse path towards the source, allocating the QoS indicated in the QoS field (i.e., 64 kbps in Figure 2.4 due to the bottleneck link between nodes 3 and 4) as it reaches each hop along the route. Once the source receives the RREP packet, it is ready to transmit data packets.

2.4.2 QoS Route Maintenance

QoS route maintenance is the process in charge of maintaining acceptable levels of QoS in the network for the duration of the data flow. Once a QoS route is established, it must be kept until the end of the data flow. In the presence of link partitions, QoS route maintenance must provide an alternate route to avoid interruptions in the data transmission (to make the error invisible to the user).

Congestion control is important to perform an efficient utilization of resources (e.g., bandwidth). The bandwidth available in the network decreases as the number of QoS reservations increases. Hence, admission control (AC) mechanism is needed to assure that the channel is still meeting the requirements that the applications require.

The first step is to classify the traffic in the network. Ad-hoc networks will support multimedia and real-time applications and they will support best-effort traffic as well.

Following the traffic classification in asynchronous transfer mode (ATM) [30] networks, the traffic in an ad-hoc network can be classified as follows:

- CBR (Continuous Bit Rate).
- VBR (Variable Bit Rate).
- UBR (Unspecified Bit Rate).
- Best-Effort (BE).

Table 2.6 depicts the requirements for different traffic classes.

Table 2.6: Traffic Class Requirements [30]

Traffic class	Application	Bandwidth	Jitter sensitive	Packet-loss sensitive	Delay sensitive
CBR	Videoconferencing	384 kbps	Yes	Yes	Yes
CBR	Voice	64 kbps	Yes	Yes	Yes
VBR	Streaming	64 kbps	Not much	Yes	Not much (buffering)
UBR	Banking	32 kbps	No	Yes	No
BE	Web Browsing	Available	No	No	No

A. Admission Control

Admission control ensures that incoming QoS route requests do not consume QoS resources already allocated, accepting route requests only when the available bandwidth is equal to or greater than the bandwidth requested. This technique enhances QoS route maintenance by assuring that the nodes reserve only resources that can be supported. Thus, the over-utilization of resources in the MANET is avoided.

The QoS routing process must perform the admission control every time a solicitor requests a route to the destination. Every node within the route must check if there are resources available and, if so, the reservation can be performed and the node must re-transmit the RREQ packet to the next hop, repeating the same process until the destination is reached. If a node along the route cannot meet the minimum QoS requirements the RREQ packet is dropped. This process must be really simple to reduce the computation complexity, delay, and CPU utilization. An example of the admission control performed during the route discovery process is depicted in Figure 2.5.

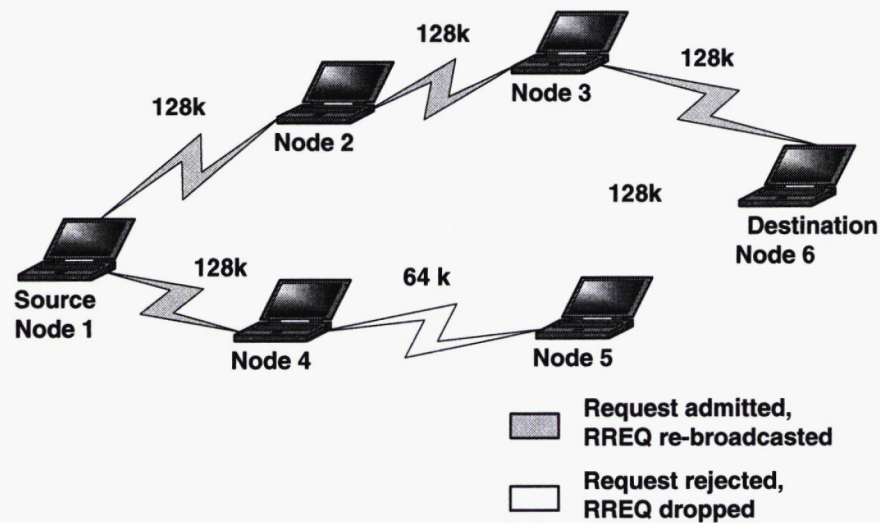


Figure 2.5: Admission Control

Node 1 requests a route with a minimum bandwidth of 128 kbps and a maximum of 256 kbps. The admission control algorithm in each node detects if the requested QoS is supported. If the node has at least 128 kbps available, the request is admitted and the RREQ packet is re-broadcasted. In this example, node 4 does not support the bandwidth required hence, the RREQ is rejected and the packet is dropped (the route request is not transmitted to node 5). Thus, nodes that do not support QoS do not contribute with unnecessary routing overhead (RREQ). Admission control enhances QoS route maintenance by assuring that the nodes in the network only allocate resources that they can provide. Thus, bandwidth is efficiently used.

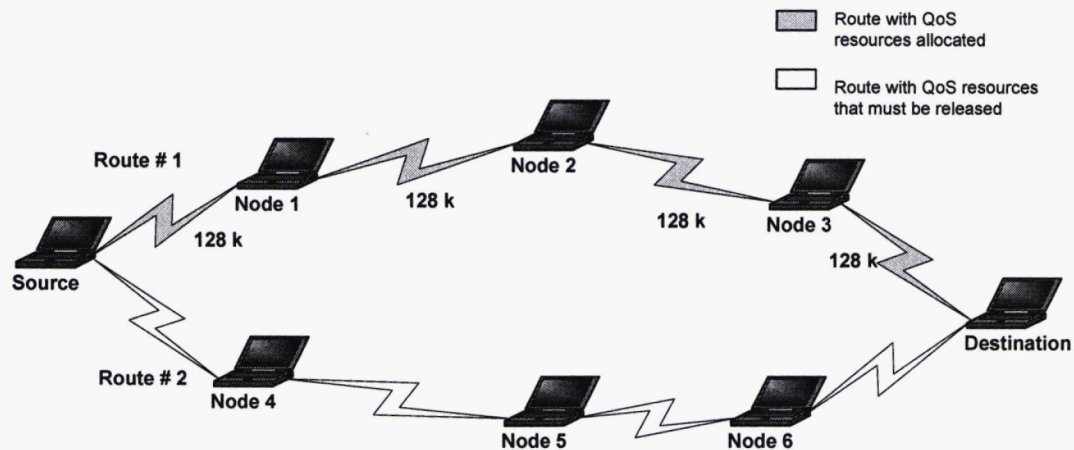


Figure 2.6: QoS Resource Release

B. QoS Resource Release

Route discovery and resource allocation are needed for QoS routing. Once the route is obtained and the resources allocated, the source can start transmitting the data flow along the selected route. If there is no change in the network topology or in the wireless channel, data transmission is performed efficiently. There is, however, another important process that must be considered to improve overall efficiency of QoS routing. This process is QoS resource release.

Once the user finishes data transmission and the route is no longer needed, the source must free the resources to allow resource re-use. QoS resource release mechanisms avoid stale reservations and idle resources. The resource release process is performed through the *expiration timeout* parameter. Signaling packets are unicast packets that are transmitted over the network releasing QoS resources in all the nodes along the route. The main drawback is the overhead these packets cause in the network. On the other hand, expiration timeout releases QoS resources when the nodes do not receive any data from a specific node within a certain period of time. Expiration timeout is fixed by the protocol and can be the same value used in the routing table to avoid stale routing information (i.e., the expiration timeout of a route in NIST / AODV is 6 seconds). If the network is highly dynamic, the expiration timeout can be decreased to achieve better performance. This approach is overhead free and does not require QoS information

exchange among the nodes in MANETs. In Figure 2.6 the source obtains two valid routes to the destination, both with the QoS already allocated. The source selects the primary route based on a pre-defined QoS routing metric (i.e., bandwidth). If both routes provide the same characteristics, a tie-breaking routing metric such as stability must be applied. Once the route is selected, the node starts data transmission. The resources allocated in the other route are released once the timer expires.

2.4.3 QoS Robustness Enhancement

QoS robustness enhancement provides fast response to link partitions to maintain the data flow. Hence, efficient repair mechanisms must be used. Local repair and multiple route support are two techniques used to provide fast response to topology changes.

A. Local Repair

Local repair is a mechanism to overcome link partitions by obtaining an alternate route to the destination. Local repair is performed by any node in the network regardless of whether or not the node detecting the link partition is the source. Route discovery is not re-initiated by the source until the error is classified as critical. Errors are defined as simple or critical based on their characteristics.

1. Simple Error

Simple errors are link partitions that can be overcome by obtaining an alternate route through one of the neighbors. Once a node detects an error in the network, it has to initiate the error correction to maintain the stability of the network. The error correction process has to be performed by minimizing the cost in terms of routing overhead in the network. Thus, once an error is detected, the node must initiate the error correction process which comprises a route discovery process among its neighbors to see whether a node still has a valid route to the destination. This process is performed in an expanding ring fashion [24] to reduce the routing overhead. The entire process is depicted by Figure 2.7.

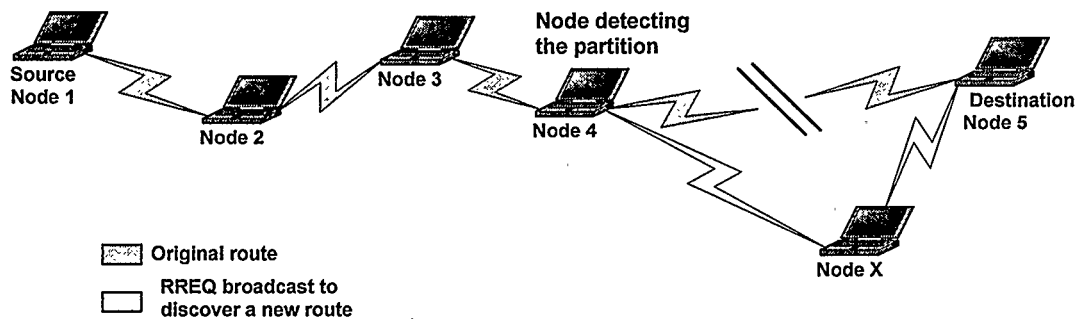


Figure 2.7: Simple Error

Node 4 detects a link partition and then it reinitiates the route discovery process to obtain a valid route to the destination (through one of its neighbors, say node X). The route discovery process initiated by node 4 ensures that node X supports the QoS required (QoS allocated in node 4). If the route discovery process is successful the QoS route (route through node X) is then used and the error is overcome. Otherwise, the error becomes critical.

It is important to state that in the case depicted by Figure 2.7 the error correction process was performed efficiently because there was a node (Node X) that provided an alternate route with the QoS required to the destination.

2. Critical Error

Critical errors are produced when a link partition occurs and the local repair process cannot obtain any alternate route from the neighbors. Therefore, the error cannot be fixed at all and the destination can no longer be reached. Hence, the node detecting the error must broadcast a route error packet through the network to inform the nodes in the network that the destination can no longer be reached through that route. This is explained in detail in the next sub-section.

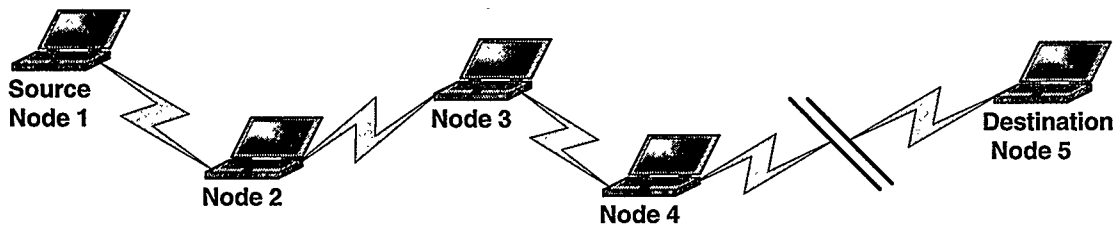


Figure 2.8: Critical Error

B. Error Packets

Route error packets (RERR) are used when repair of a critical error is unsuccessful, as illustrated in Figure 2.8. However, error packets are also used when data packets arrive to a node and it can no longer provide a route to the destination (route expired). These informative route error packets are transmitted to the source generating the data packets to indicate that the destination is unreachable. In Figure 2.8, error packets are broadcast through the network to indicate that node 4 has lost connectivity with the destination (node 5). Nodes receiving the RERR packet must update their routing table (i.e., release resources in case of allocations and delete the node from the routing table) if they rely on node 4 to reach their respective destinations. Once the source receives the RERR packet, it must re-initiate the route discovery process. The route error packet structure is illustrated in Table 2.7.

Table 2.7: Route Error Packet Structure

Reserved	Type	Dest Count
Error Dest IP address		
Error Dest Seq Num		
Cause of Error		

The *Cause of Error* field in RERR packet contains an error code that indicates the cause of error. This can be a link partition or a decreasing QoS level.

Route error packets cooperate in enhancing QoS robustness by informing the nodes within the network that a change in the network has occurred and the route previously established can no longer be used.

C. Multiple route support

Some routing protocols just maintain a single route to the destination but some others store multiple routes to the same destination [29] to be utilized as back-up routes once a link partition is detected.

Once a node receives multiple route reply packets conveying routing information, it must select the primary and the back-up routes. Once the node has selected the primary route, it must organize the back-up route(s) based on pre-defined tie-breaking metrics such as stability and number of hops. In this thesis, a maximum of two back-up routes per destination is allowed.

Backup-routes can speed the convergence time of the routing protocol. Once a link partition occurs a back-up route is used rather than initiating the error correction process. Consequently, the routing overhead produced by repair attempts (error correction process) is reduced considerably, in this manner saving important resources in the network such as bandwidth.

1. Enabling Multiple Route Support

Enabling multiple route support in routing protocols does not imply major modifications in the routing process. There are some modifications (extra fields in the routing table and routing packets, route request list, et cetera) that must be performed in the routing elements such as routing packets, routing table, extra table to record route request ids, et cetera. In this thesis, the NIST / AODV OPNET [33] model was modified based on the proposed QoS routing framework to support QoS metrics, call admission control, resource reservation and multiple routes.

Table 2.8: Routing Table Additions for Multiple Routes Support

Destination	Destination
seq. number	seq. number
hop count	advertised_hopcount
next hop	next hop
expiration_timeout	backup_routes_list [Next_hop_1, num. of hops, tie-breaking metric Next_hop_2, num. of hops, tie-breaking metric]
	expiration_timeout
	QoS_RSV

(a) Classic routing table entry

(b) QoS Multi-path routing table entry

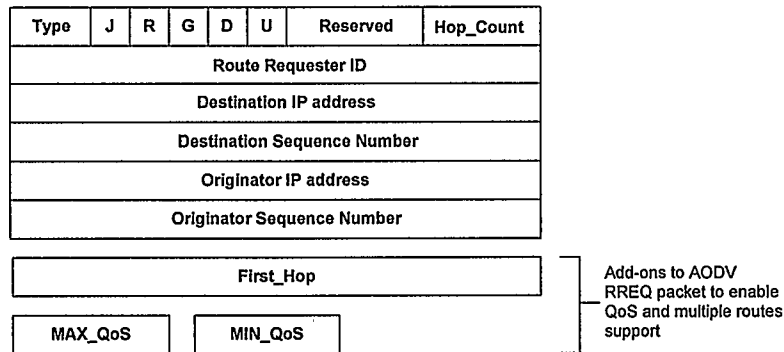
Table 2.8a depicts a generic routing table entry showing only the most important fields. Table 2.8b illustrates the extra field additions proposed by the QoS routing framework to enable QoS and multiple route support [29]. The field *advertised_hopcount* stores the highest number of hops of the back-up routes. These routes are only updated if the new route's hop count is less than the value in the advertised count field. The field *backup_routes_list* stores the IP address of the next hop to the destination. In this case it can store more than one IP address, each one corresponding to a back-up route. This list also stores the back-up route selection metric (e.g., number of hops) and the tie-breaking metric (e.g., stability). In Table 2.8a hop count is given as an example of a back-up routing metric. QoS metric is not used as a back-up route selection metric due to the fact that all the routes in the back-up routes list satisfy the minimum QoS level required but the QoS resources are not allocated. Multiple route support enhances QoS by decreasing the average discovery time and consequently the routing overhead produced by continuous repair attempts.

2. Route Discovery Process with Multiple Routes Support

In the route discovery process the source broadcasts a route request packet in order to find route(s) to the destination. The route request packet is fundamental to perform route

discovery process efficiently. Thus, a slight modification in the route request packet enables multiple route support. Table 2.9 illustrates the new additions to the RREQ packet.

Table 2.9: Multiple Routes QoS RREQ Packet Structure



First_Hop field stores the IP address of the *first* node(s) receiving the RREQ packet (source's neighbors). Once the first node(s) rebroadcasts the packet, the information in this field (first node IP address) remains intact.

The source initializes the *First_Hop* field (zero value is placed), then it broadcasts the RREQ packet. Once the neighbors receive the RREQ packet, *First_Hop* field is checked. If the value stored in *First_Hop* is zero, then the nodes receiving the packet (neighbors) utilize this field to place their own IP address. Otherwise, the field remains intact and the value in the *First_Hop* field is stored in a route request list. Intermediate node(s) discard route request packets if the route request packet has been seen previously (i.e., with identical sequence number) *and* the first hop value is already in the route request list (this avoids the formation of loops). This reduces the routing overhead produced by redundant route request packets (from the same node). The next steps are the same as in single route discovery process.

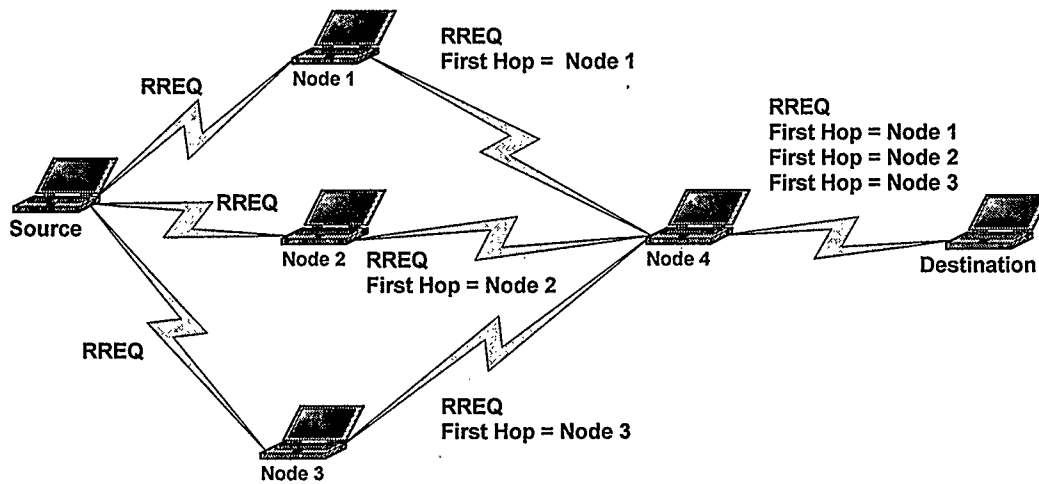


Figure 2.9: Multiple Routes QoS RREQ Process

Figure 2.9 illustrates the multiple routes QoS RREQ process. The source node broadcasts a RREQ packet to its neighbors (node 1, node 2, and node 3). The neighbors receiving the RREQ packet place their IP address in the *First_Hop* field and then they re-broadcast the packet. Node 4 receives the first RREQ packet from node 1, places the IP address of node 1 in its route request list and then re-broadcasts the RREQ packet to obtain a route to the destination. Node 4 does the same for other RREQ packets (transmitted by nodes 2 and 3) until the maximum number of back-up routes (2 is assumed in this thesis) is reached or the IP in the *First_Hop* is found in the route request list.

3. Route Reply Process with Multiple Route Support

Route reply process is performed in the same manner as in single route approach. The principal difference is that nodes allow the propagation of multiple route replies to the same source along the reverse route to the source. RREP packets are unicast along the nodes forming the routes from destination to the source. Thus, the source will receive multiple RREP messages from the nodes within the network. Nevertheless, the source only stores the number of RREP needed to maintain one primary and two back-up routes. Figure 2.10 illustrates the route reply process with multiple route support. Once the RREQ packet(s) arrive at the destination, it transmits the RREP packets along the reverse route(s) generated in the route discovery process. The destination is only allowed to

transmit three RREP packets to the same source due to the maximum number of back-up routes constraint. Thus, the RREP packet(s) goes/go via the route(s) stored in the destination's routing table. Intermediate nodes cannot generate RREP packets due to the fact that the maximum QoS supported along the route is not established until the RREQ packet arrives at the destination.

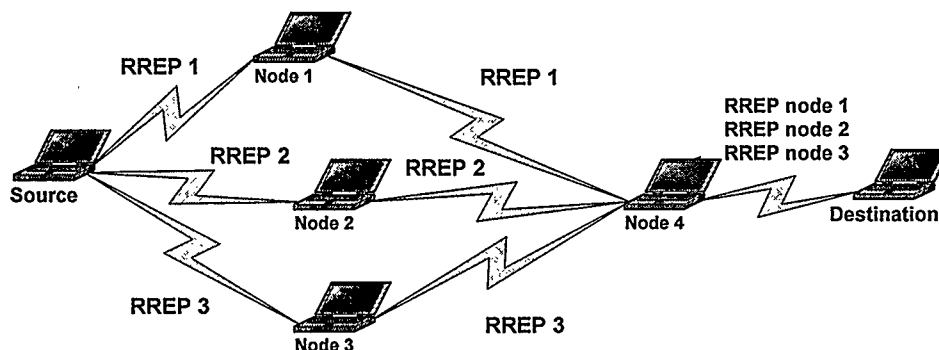


Figure 2.10: Multiple Routes QoS RREP Process

4. Primary and Back-up routes selection

As illustrated in Figure 2.10, once the source receives the first RREP packet, it selects this route as primary due to the fact that the route meets the minimum QoS required and it arrives faster than the others. The other RREP packets arriving to the source are selected as back-up routes. The IP addresses of the back-up routes are stored in the back-up routes list. Number of hops is also stored for route selection purposes.

Back-up routes are utilized once a link breakage is detected. This is to avoid the interruption of data flows. The drawback of this approach is that sometimes the routes determined as back-up no longer exist. If a back-up route is selected but is no longer valid, the node has to detect this anomaly and re-initiate route discovery. This consumes more time than if the node just discovers the route again rather than select a back-up route. QoS implies resource allocation *a priori*. Back-up routes are only alternate paths; they do not maintain the resources allocated. Thus, once a back-up route is selected after a link breakage the data transmitted must travel as best effort [26] data. This can at first sound as a drawback but it is not, because the user wants the “errors” in the network to be invisible. Thus, as soon as the data is transmitted as best effort, the node detecting the

error initiates the resource allocation along the back-up route utilizing the RREQ packet (the RREQ packet is unicast due to the fact that the back-up route is already known). If the QoS required cannot be allocated in the back-up route, then the route discovery process is re-initiated.

2. Tie-breaking metric based on stability

If two back-up routes have the same value in the back-up route selection metric (e.g., number of hops) then a tie-breaking metric must be applied. This thesis introduces a new tie-breaking metric called stability which is defined as the level of routing activity of a certain node with respect to another. The main assumption for the stability metric is that the nodes in the network are highly active in terms of routing information exchange (assumption 3 of Section 2.3.1).

A wireless channel is a broadcast medium. Every time a node transmits a packet to the network all the nodes within its wireless range receive the packet. The capability of overhearing the data flowing in the network is used for our purpose. Note that every time a node transmits a packet no matter if it is a control packet for medium access, routing packet, or data packet, this node is advertising itself among its neighbors. Hence, each node receiving the packet can check its routing table to look for the owner of the packet. If a node is the owner, the stability metric is increased by one. Of course we have to be careful about which packets can affect the stability metric. If we allow all the traffic in the network to update the stability metric the nodes will spend more time updating stability metrics than doing more important functions. In this thesis, only the routing packets are used for updating the stability metric. Thus, every time a node receives a routing packet from one of its neighbors, the node verifies whether or not the IP address of sender is in the routing table. If so the update is performed, otherwise the node does nothing. Multiple route support enhances QoS robustness by improving the route discovery process in terms of average discovery time and minimizing the routing overhead produced by repair attempts.

2.5 Summary

This chapter has reviewed the issues that must be resolved to improve the QoS routing process in MANETs. We propose a QoS routing framework for resolving the QoS routing issues. The framework comprises overhead reduction techniques, QoS route discovery, QoS robustness enhancement, and QoS route maintenance techniques.

Routing overhead reduction is in charge of maintaining the routing overhead as low as possible in the network. QoS route discovery provides mechanisms to enable QoS and multiple routes support regardless of the underlying routing protocol. QoS robustness enhancement tries to assure certain level of QoS within the network along the duration of the data flow and QoS route maintenance techniques provide efficient utilization of bandwidth in the network. Thus, the combination of these mechanisms by applying the QoS routing framework addresses the QoS routing issues described in this chapter to assure and maintain adequate QoS levels in MANETs.

Chapter 3

QoS Routing Framework Simulation Environment

3.1 Introduction

The proposed QoS routing framework provides a set of rules and additions to the reactive routing protocols for MANETs to enable QoS support. There are several ways to evaluate the proposed QoS routing framework: analytical (mathematical analysis), computer simulation, and experimentation on a test-bed. An example of a test-bed is the implementation of AODV-UU (Uppsala University) [31]. Thus, AODV-UU can be modified based on the QoS routing framework. However, the recreation of different levels of congestion or mobility is quite challenging. This implies that several mobile devices such as laptops or PDAs must be moving around at different speeds, which is not easy to coordinate. On the other hand, evaluation of congestion levels within the network implies several devices transmitting packets constantly. Computer simulation allows the recreation of these scenarios by only varying parameters in the simulation. Hence, the evaluation of the QoS routing framework in this thesis is performed through computer simulation.

3.2 Simulation Objective

The principal objective of the simulation is to evaluate the performance of the QoS routing framework under different scenarios (i.e., mobility, congestion). The simulator platform used for this analysis is the Optimum NETWORK Performance simulator (OPNET) [32], a commercially available network-level simulator.

The QoS routing framework reduces the cost of the routing process in terms of routing overhead, power consumption, and average route discovery time in mobile ad-hoc networks, providing at the same time, routes with acceptable levels of QoS (i.e., bandwidth, in the context of this thesis).

3.3 Simulation Structure Using OPNET

OPNET simulator relies on multiple levels to model the behavior of individual objects. Based on this, OPNET defines three models: process model, node model, and network model. Process model utilizes finite state machine modeling to address specific issues such as mobility, radio transmission, medium access, routing, traffic generation, et cetera. Node model integrates a set of processes to create individual objects such as mobile nodes. Network model allows the interaction between individual objects with the same or different characteristics. In this thesis the network model defines the network area where the mobile nodes are placed.

3.3.1 OPNET Node Model for QoS Routing Framework

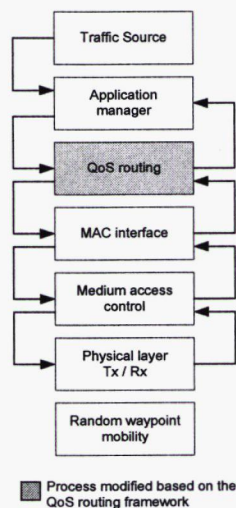


Figure 3.1: OPNET Node Model

For simulation purposes the most important layers of OSI model are integrated in the node model with the aim of providing functionalities such as traffic generation,

application management, routing, medium access control and physical layer definition.

Figure 3.1 illustrates the OPNET node model for each mobile node in a MANET.

Traffic source process is in charge of generating data packets to simulate different traffic conditions. Traffic generated in this stage can be modified through the parameters in Table 3.1.

Table 3.1: Traffic Parameters

Parameter	Value
Inter-arrival ¹ pdf of the data packets	Exponential
Inter-arrival args.	0.25 seconds
Packet size pdf	Constant
Packet size args.	512 bytes
Start time	0.0
Stop time	1800 secs (Simulation time)

¹ pdf stands for probability density function

Inter-arrival time for data packets is exponential due to the fact that is assumed that data packets arrive according to a Poisson arrival process. The mean data packet arrival rate per node is 4 packets per second with a constant packet size of 512 bytes. The number of nodes in the network generating traffic continuously during the simulation is specified by the number of flows parameter (a maximum of 30 flows is assumed in this thesis). The traffic parameters used in the simulation are the same as used in the original NIST / AODV.

Application manager handles incoming data packets from upper or lower layers. In case of receiving data packets from traffic source process, it interacts with the routing process to obtain a valid route for transmitting the packet to the destination. On the other hand, if the application manager process receives data packets from lower layers, it only updates statistics related to incoming data packets. The source and destination nodes are randomly selected by the application manager at the beginning of the simulation. One

flow implies one specific source and destination. Source nodes cannot have the same destination in the simulation.

MAC interface interacts with incoming packets from MAC and application processes. Packets received from MAC are sent to upper layers only if the packets are destined for this node, otherwise the packets are rejected or forwarded. Incoming packets from application layer are sent to MAC interface for transmission on the radio link.

Medium access control defines characteristics such as medium access protocol, wireless LAN parameters (i.e., IEEE 802.11), station address, etc. In OPNET, these parameters are specified as part of the physical layer process parameters. Physical layer process is divided into two: Transmitter process (Tx) and receiver process (Rx). Receiver process defines the receiver characteristics: path loss, multi-path fading, and other physical phenomena that affect the signal power. Table 3.2 shows the parameters for data reception used in the simulation.

Table 3.2: Receiver Parameters

Parameter	Value
Maximum data rate	2 Mbps
Min. Frequency (MHz)	2,400 (2.4 GHz, ISM band)
Spreading code	Disabled
Modulation	Bpsk
Power model	dra_power
SNR model (Signal to Noise Ratio)	dra_snr
BER model (Bit Error Rate)	dra_model
Error model	dra_error

The data rate for the simulation is 2 Mbps. The frequency used by the nodes is 2.4 GHz, BPSK modulation and spreading code disabled (frequency hopping). These parameters correspond to the specification of the IEEE 802.11. The power model used in the simulation corresponds to the Free-Space propagation mode. SNR, BER and error control are introduced by the models described in Table 3.2.

Transmitter process provides the ability of transmitting packets among nodes within the network. Configuration of this process is very important to achieve realistic

results from the simulation. The following parameters must be taken into consideration before modeling a wireless system. Table 3.3 illustrates transmitter process parameters.

Table 3.3: Transmitter Parameters

Parameter	Value
Maximum data rate	2 Mbps
Packet formats	All formatted, unformatted
Min. Frequency (MHz)	2,400 (2.4 GHz, ISM band)
Spreading code	Disabled
Power (mW)	100
Modulation	Bpsk
Propagation delay	Specified by the wlan_propdel model in OPNET

The same parameters used in the receiver must be used by the transmitter for compatibility. 100 mW are defined as the transmission power of each node in the network utilized for every packet transmission. The propagation delay, a measure of the distance between a receiver and a transmitter, is introduced by OPNET and it is part of the IEEE 802.11 implementation provided by the simulator.

3.3.2 QoS Routing Process Model

The QoS routing process model relies on twelve states to perform routing within the simulation. Figure 3.2 illustrates the states modified to enable QoS provisioning based on the QoS routing framework.

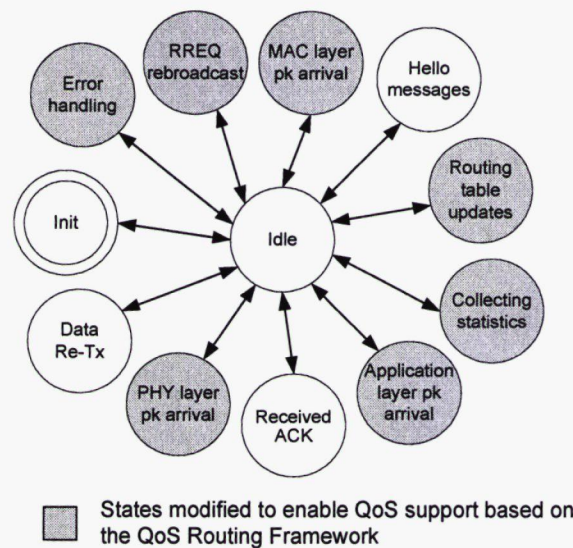


Figure 3.2: OPNET QoS Routing Finite State Machine

A. Initialization (Init)

This state initializes all the parameters needed in the simulation: variables, constants, characteristics, parameters, etc.

B. Physical layer packet arrival

This state is used to determine the local connectivity among neighbors within the wireless network based on the packets arriving at each node from other nodes. Stability metric relies on the physical layer state for updating purposes.

C. MAC Layer packet arrival

This state handles the incoming packets from MAC layer, taking different actions based on the nature of the packets. The main criterion to handle packets is based on whether the packet is for the node receiving the packet or if it is for another node. MAC layer packet arrival also processes routing packets such as RREQ, RERR and RREP. QoS support is also handled by the MAC layer. If the QoS specified can be met, the routing packet is processed otherwise it is dropped.

D. Application layer packet arrival

This state handles the incoming packets from application layer. An application generates data packets and the application layer packet arrival sends them to the lower layer for transmission.

E. Routing table updates

This state handles routing updates. It makes sure that the information contained in the routing table is not stale. The routing table updates occurs when the timer of any entry expires. The following cases are possible:

1. The expired entry is active. Node then invalidates it and schedules its deletion. Besides, if an entry is flagged as broken, the node resets the breakage flag before it schedules the deletion process.
2. The expired entry is under repair. Delay expiration time and wait until the end of the discovery process.
3. The expired entry is invalid (broken, lack of QoS resources, etc.). In this case, the node simply deletes it from its routing table except if it is under repair. This process is very important to release QoS resources that are no longer used.

Routing table updates are also performed to release QoS resources once a route is no longer needed.

F. Error handling

This state handles errors in the network. Errors can be generated by different causes therefore the repair process utilizes different approaches depending on the nature of the error. The causes of error are the following:

1. A node receives a packet and it cannot provide a valid route to destination either because it does not have a route or the route does not meet the bandwidth required.

2. Failure in the repair attempt for a specific destination (The node already tried to repair it)

Link partitions are detected through a MAC layer mechanism called NACK. When a packet is transmitted by a node and its successful arrival is not notified by the medium access layer, then the node itself notifies the upper layer (i.e., routing) by sending a NACK message containing both final and next hop destinations of the lost data packet. Once the NACK reaches the upper layer the state Error Handling is executed. Nevertheless, QoS errors are detected by any node that can no longer provide the QoS level requested.

G. RREQ rebroadcast

This state occurs when a route reply timeout (maximum time a node should wait for a route reply packet, NIST / AODV defines 3 seconds as route reply timeout) expires. The node is allowed to rebroadcast a route request packet if and only if the number of route request re-transmissions is smaller than the maximum route request re-transmissions specified by the routing protocol (i.e., a maximum of 2 retries is assumed in the simulation). Otherwise, the discovery process is terminated.

H. Data retransmission (ACK timeout)

This state occurs when the downstream did not acknowledge a data packet. This state stores in the buffer a copy of the non-acknowledged data packet and attempts to perform local repair on the broken link. Once the route is repaired, a copy of the data packet is re-transmitted.

I. Received ACK

This state handles acknowledgement arrivals. Once an ACK packet is received, this state transmits it to upper layer to indicate that the data transmission was successful.

J. HELLO messages

This state handles HELLO messages [23] depending on the type of messages defined in the simulation init state. RREP or RREQ packets can be chosen to say HELLO. In the simulation of the routing protocol based on the QoS routing framework, HELLO messages use RREP packets.

K. Collecting statistics

The main purpose of this state is to collect statistics from the simulation.

L. Idle

This state is the default state when the QoS routing process is not required.

To activate QoS provisioning in the routing process the following states were modified based on the QoS routing framework: Physical layer packet arrival, MAC layer packet arrival, route request rebroadcast, error handling, routing table updates, application layer packet arrival, and collecting statistics.

3.3.3 OPNET Network Model

Figure 3.3 shows the network model used in the simulation with the general simulation parameters listed in Table 3.4.

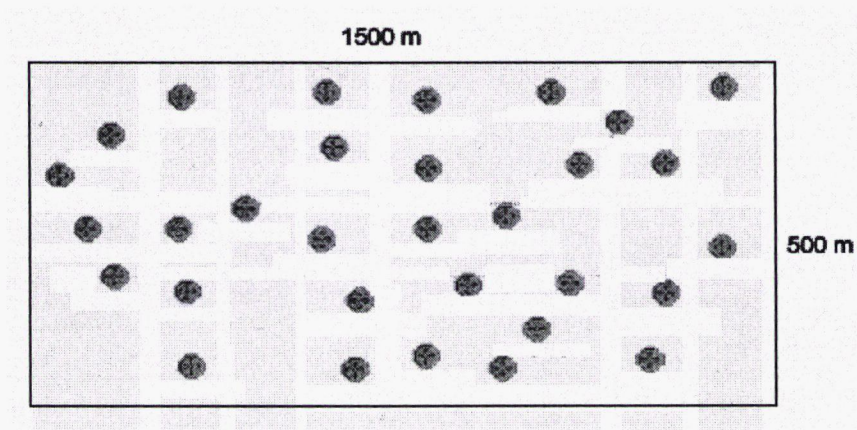


Figure 3.3: Example Network Model Showing Initial Locations of Nodes

It is important to note that the same simulation parameters were utilized in the NIST / AODV OPNET model, hence providing a fair comparison of the results.

Table 3.4: OPNET General Simulation Parameters

Parameter	Value
Network area	1500 m x 500 m
Number of nodes	30 (maximum)
Number of CBR Sources	¹ Variable [0, 5,10,15,20,25, and 30]
Data packet size	512 bytes
Maximum Capacity Available per node	² Variable 1. Some nodes only support 128 kbps to reduce the number of QoS routes supported by each node. 2. Nodes maximum capacity of 2 Mbps can support multiple QoS requests.
Source data rate	1. Based on the QoS allocated (QoS routing framework). 2. 2 Mbps for NIST / AODV.
Number of back-up routes	2 per node
Medium access protocol	³ IEEE 802.11 CSMA / CA, DCF mode
Routing Protocol	1. QoS aware multiple route protocol based on NIST / AODV (Thesis proposal) 2. NIST / AODV [33]
QoS requirements	1. 64 kbps and 128 kbps for the QoS routing framework routing protocol. 2. No QoS for NIST / AODV
Repair mechanism	1. Back-up routes for the QoS routing framework routing protocol 2. Local repair no multiple routes for NIST / AODV
Wireless range per node	250 m
Mobility Model	Random waypoint mobility model
Pause time	150 seconds (constant)
Max Speed Limit	⁴ Variable [0.1, 0.5, 1.0, 1.5, 3.0, 5.0, 10, 15, and 20 m/s]

¹ Number of CBR sources varies to evaluate the performance of the routing protocol under different congestion levels.

² Maximum bandwidth available creates a heterogeneous MANET by reducing the number of routes supported by the nodes. This is useful to evaluate admission control and QoS support.

³ RTS / CTS was disabled for this simulation.

⁴ Max Speed Limit varies to recreate low and high mobility environments.

3.4 Inputs to the Simulation

This section describes the major inputs employed to simulate the proposed QoS routing framework for MANETs.

3.4.1 Size of the Network

Network size defines the geographic area to be covered by the QoS routing protocol. Based on the assumptions of Section 2.3.1, the QoS routing framework proposed in this thesis operates in campus area environments.

3.4.2 Physical Layer

Physical layer controls parameters such as bandwidth, transmitting power, data rates, modulation schemes, minimum frequency, antenna pattern, radio propagation channel, et cetera. Any variation in one of these parameters affects the overall performance. Wireless range is the geographic distance the nodes can cover with their wireless link. For this simulation a wireless range of 250 m was selected. The data rate supported by the nodes in the network is 2 Mbps. However, the maximum bandwidth available in the nodes is modified to create a heterogeneous MANET. Ten nodes each supporting 128 kbps (to perform less QoS allocations) whereas the rest (twenty nodes) each supporting up to 2 Mbps.

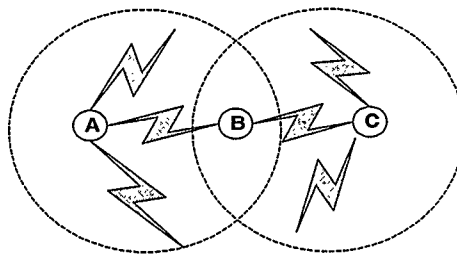


Figure 3.4: Hidden Terminal Problem

3.4.3 Radio Propagation Loss

Radio signals are affected by the environment and distance between transmitter and receiver. To achieve realistic results from the simulation, radio propagation loss must be

considered. Physical phenomena such as multi-path fading, shadowing, and path loss can be applied in the simulation to make it more realistic. The propagation model used is the free space model [1] with a power signal attenuation of $1 / d^2$ where d is the distance between nodes. Shadowing and multi-path fading were not considered in the simulation for consistency with the NIST / AODV model.

3.4.4 MAC Layer

The MAC layer protocol used in the simulations is the IEEE 802.11 CSMA / CA operating in Distributed Coordination Function (DCF) mode. However, previous work has shown that CSMA / CA has a problem with MANETs [45]. IEEE 802.11 relies on RTS / CTS control packets for unicast data transmission between neighboring nodes. RTS / CTS control packets reduce the impact of the hidden terminal problem in the network. Figure 3.4 illustrates the hidden terminal problem. Nodes A and C transmit simultaneously to node B. Because both nodes can reach node B, data packets collide at node B. If one signal is stronger than the other, then node B can successfully receive the data packet. Otherwise, both transmissions are lost. Thus, each transmission failure at the MAC layer (medium access control process) is reported to the routing layer (QoS routing process) as a link breakage. In such a situation, the latter undertakes the maintenance procedure which leads to new route discoveries. Hence, link partitions detected by the nodes are also due to the hidden terminal problem.

Similar to NIST / AODV implementation, the RTS / CTS handshaking feature was not implemented in our simulation in order to make a fair comparison between NIST / AODV and AODV modified based on the proposed QoS routing framework. Note that the impact of not implementing RTS / CTS is degraded network performance (i.e., more collisions and high end-to-end delay).

Table 3.5 depicts the parameters utilized in the simulation. Some parameters might vary depending on the scenario being evaluated (mobility, congestion level, etc.).

Table 3.5: MAC Layer Parameters

Parameter	Value
Physical characteristics	Frequency Hopping
Short retry limit (slots)	7
Long retry limit (slots)	4
Access point functionality	Disabled (DCF)
RTS / CTS	Disabled

To adopt the same MAC layer characteristics specified in the NIST / AODV OPNET model, the following parameters are used in the simulation. The multiple access mechanism used is frequency hopping with a short retry limit of 7 slots and a long retry limit of 4 slots, according to the specification of the IEEE 802.11. Access point functionality is disabled to operate in ad-hoc network mode. RTS / CTS are disabled as stated above.

3.4.5 QoS Routing Protocol

The NIST / AODV model for OPNET created by Guemari and Miller [33] was modified following the proposed QoS routing framework. Thus, QoS support was enabled in a routing protocol that originally does not provide that functionality. Table 3.6 depicts the difference between the original AODV and the AODV modified based on the QoS routing framework.

Table 3.6: Routing Protocols Comparison

Criteria	¹ AODV modified based on the QoS routing framework	AODV [23]
Routing philosophy	Flat	Flat
Routing Algorithm	² DBF	DBF
Loop-Free	Yes	Yes
Multiple routes	³ Yes	No
Reactive/Proactive	Reactive	Reactive
Link Support	Only symmetric	Only symmetric
Periodic Broadcasts	⁴ Yes	Yes

Beaconing Requirements	No	No
Routes maintained in	Route tables	Route tables
Routing metric	Freshest and Shortest path, ⁵ stability, and QoS metrics such as bandwidth	Freshest and shortest path
⁶Critical Nodes	No	No
QoS support	⁷ Yes	No

¹ This routing protocol was built based upon the proposed QoS routing framework for MANETs.

² DBF stands for Distributed Bellman-Ford based.

³ Multiple routes are supported, following the techniques described in the QoS routing framework.

⁴ HELLO messages are used to determine connectivity among nodes.

⁵ Stability metric indicates the reliability of the route to be selected. It is a new addition based on the QoS routing framework.

⁶ Critical nodes are nodes that the network requires to operate, such as cluster-heads.

⁷ QoS is supported by the QoS routing framework.

3.4.6 Application Layer

Traffic is necessary to evaluate protocol performance. For this simulation constant bit rate (CBR) sources are utilized. The number of CBR sources varies depending on the level of congestion desired. In this thesis, a flow is defined as a CBR source (node) continuously transmitting packets to a particular destination. Application layer is defined by the traffic parameters in Table 3.1.

Application layer also defines the QoS requirements for the data flow. Two groups of nodes are defined in the simulation. The first group (ten nodes) requires 64 kbps as a QoS constraint whereas the second (20 nodes) require 128 kbps. The connections are accepted or discarded based on the maximum available bandwidth of each node specified in the physical layer. It is important to state that this configuration (10 and 20 nodes) is similar to the configuration for maximum available bandwidth (Section 3.4.2). However, they are independent concepts. The former specifies the capacity of the nodes in terms of bandwidth whereas the latter specifies the QoS requirements for the data flow.

A. Number of traffic sources

This parameter allows the evaluation of the QoS routing framework under different levels of congestion. The number of sources is varied in the simulation to create different traffic levels in the network. A flow is defined as a node continuously transmitting data packets. In the simulation of the QoS routing framework a node cannot be a flow if there is no QoS route available. Hence, the number of flows accepted is equal to the number of flows specified in the simulation due to the fact that the nodes can find QoS routes through different nodes in the network. Thus, the simulation results reporting routing efficiency for the QoS routing framework are based on the number of flows accepted.

A maximum of 30 flows (30 nodes transmitting data packets) is allowed in this simulation because of the very long computation time exhibited by OPNET simulator when the number of flows exceeds 30. Multiple flows from the same source are not allowed in this thesis.

3.4.7 The Mobility Model

The mobility model is very important in simulating MANETs. This model is in charge of node mobility within the network. In this thesis, random waypoint mobility model [34] has been used for motion in MANETs. For the waypoint mobility model, the nodes are initially placed randomly in the network. Next the nodes select a random destination (within the specified area) and a random speed v in the range $(0 < v \leq \text{max_speed})$. Once the node reaches the destination, it pauses a constant time p $(0 < p \leq \text{max_pause_time})$. When the pause time expires the node picks another random destination and speed and repeats the process until the simulation ends. In the simulation, max_speed varies to create different levels of mobility in the network (Table 3.4). Max_speed is a parameter that affects all the nodes in the simulation. Pause time is constant in the simulation with a value of 150 seconds. Figure 3.5 depicts random waypoint mobility model.

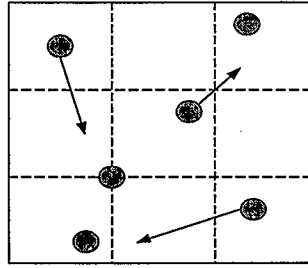


Figure 3.5: Random Waypoint Mobility Model

3.5 Simulation Outputs

The outputs produced by computer simulations show the performance of the QoS routing protocol. Simulations provide lots of data that is often hard to analyze. Before analyzing the data extracted from the simulation it is therefore necessary to establish which parameters must be evaluated and then collect the statistics needed to initiate the analysis. In case of QoS routing framework evaluation, the performance metrics are the following:

1. *Throughput* is the amount of data packets moved successfully from source to destination during the simulation period and is given by:

$$\text{Throughput} = \frac{\text{Received Data Packets}}{\text{Simulation Time}}$$

expressed in the unit of packets per unit time.

2. *Average End-to-End Delay* per packet includes all possible delays from the moment a packet is generated to the moment it is received by the destination node. This metric is expressed in seconds and takes into account buffering times.
3. *Average route discovery time* indicates the average amount of time the nodes spend in the route discovery process (total route discovery time divided by the number of route discovery attempts). Route discovery can be triggered due to new route requests or to re-arrange the routes after a link partition.
4. *Efficiency* indicates the percentage of data packets (including re-transmissions) that arrive successfully to the destination and can be expressed as:

$$\text{Efficiency} = \frac{\text{Data Packets Received by Destination}}{\text{Data Packets Transmitted by Source}}$$

5. *Routing overhead* is the amount of routing packets (route request, route reply, and route error) transmitted during the simulation.
6. *Power Consumption* indicates the average battery power consumed by each node during the simulation and it is expressed in mW (milliwatts). Power consumption is calculated based on the transmission power (i.e., 100 milliwatts are assumed in the simulation) and the amount of packets (i.e., routing and data packets) transmitted during the simulation.

3.6 Verification of Simulation Outputs

Verification of simulation outputs is important to ensure that the simulation is performing as intended. The OPNET model developed for this thesis was verified by tracing approach.

On-screen tracing verify that the program is actually following the steps it is supposed to do. For example, random selection of the destination node, random selection of the CBR sources (to avoid biased results due to the geographic location of the nodes), data packet transmission, routing packets transmission, routing table generation, et cetera. This process can be visualized on the screen.

As the simulation is driven by random numbers (assuming the same seed), several runs were performed (assuming the same seed but varying the run time) to ensure that the simulation is in steady state region. Simulation runs of duration 150, 300, 600, 900 and 1800 seconds were performed. It was found that the outputs of simulation of lengths 900 and 1800 are within +/- 10 % which is good enough to identify the steady state region in the simulation. A simulation time of 900 seconds after transient period of 900 seconds was chosen. The results presented in this thesis are the average of ten simulation runs (each lasting 1800 seconds) with independent random seeds. The random seeds are kept the same for NIST / AODV and QoS routing framework simulation runs. The error bars in the graphs illustrating the results in this thesis correspond to confidence intervals of 95 %.

3.7 Validation of Simulation Outputs

The purpose of the validation is to check the correctness of simulation outputs. Two methods were used to validate the simulation outputs. It is important to state that the parameters utilized in both simulations are the same.

1. Comparison with NIST / AODV results.

To perform this validation, two simulations under the same scenarios were performed. The first simulation uses the NIST / AODV as a routing protocol whereas the second simulation uses the modified NIST / AODV based on the QoS routing framework. Figure 3.6 illustrates that the efficiency achieved by NIST / AODV is within the error interval of the results for NIST / AODV based on the QoS routing framework. This good agreement provides some confidence in the outputs of our simulation.

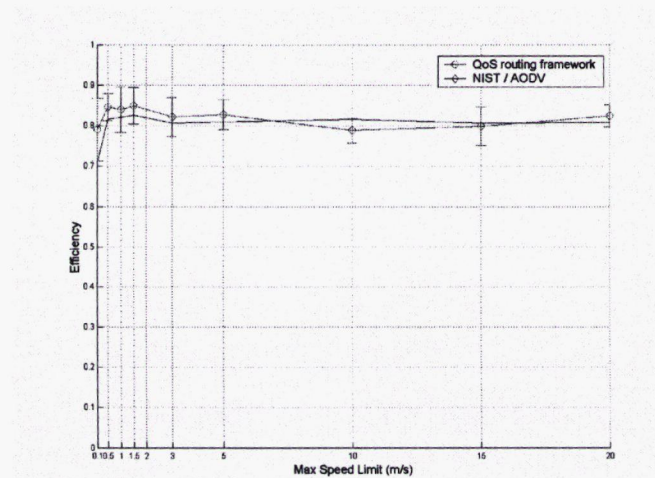


Figure 3.6: QoS Routing Framework Validation

2. Comparison of simulation results with analytical results.

The second method used to validate the results from the simulation is comparison of simulation results with analytical results. First, we note that end-to-end analytical modeling of delay is difficult because of many nodes (queues) in the network. Hence, the

mathematical analysis performed to validate the simulation results was based on a simplified analytical model focusing on the performance of one node using the queuing model shown in Figure 3.7. The analytical results were compared with the simulation results to validate the simulation outputs. The analysis objective is to determine the average delay experienced by a packet at a node.

A. Assumptions for Analysis

In order to analyze the routing process mathematically on a per node basis some assumptions have been made.

1. Poisson packet arrival process, with a mean rate obtained from the simulation,
2. Exponential service time with a mean service time determined in the simulation.
3. One server
4. Finite queue size K , with capacity 64 packets.

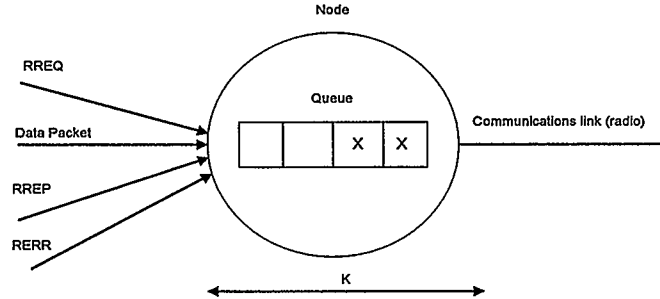


Figure 3.7: M/M/1/K Queueing Model

From the above assumptions, the M/M/1/K queueing model is selected for analysis. The Poisson arrival process is justified by the large number of packets generated. Service time is assumed to be exponential because the packets (data packets or routing packets) are served based on whether or not the node receiving the data packet is the final destination or an intermediate node. If the node is the destination the packet is processed in a time t (packet processing time by the upper layers). But if the node is an intermediate node, it must process and forward the data packets to the next hop. Forwarding the data packet

implies the following: if the node does not have a valid route it has to initiate the route discovery process if local repair is enabled. Hence, the data packet is processed in a time $t + dt$ (where dt , is the route discovery time). If after certain period of time (established by the routing protocol) a route cannot be found, the packet is removed from the buffer and the error handling process is triggered. Therefore, the service time cannot be assumed as constant. We assume exponentially distributed service time for analytic tractability. The performance measures for the M/M/1/K queueing model are summarized in Table 3.7 [38].

Table 3.7 M/M/1/K Queueing Model Parameters

Parameter	Value
K	64 packets
λ	Average packet arrival from computer simulation.
$1 / \mu = (t + dt)$	Service time from computer simulation (average discovery time).
$1 < \rho < 1$	$^2E[n] = (\rho / (1 - \rho)) - ((K+1) * (\rho^{K+1}) / (1 - \rho^{K+1}))$ Eq. (2) $^3P(K) = \rho^K * (1 - \rho) / (1 - \rho^{K+1})$ Eq. (3) $^4E[t] = E[n] / \lambda * (1 - P(K))$ Eq. (4)
$\rho = 1$	$E[n] = K / 2$ Eq. (5) $E[t] = E[n] / \lambda * 1$ Eq. (6)

¹ ρ is the utilization of the node.

² $E[n]$ is the expected number of packets in the node.

³ $P(K)$ is the probability of queueing.

⁴ $E[t]$ is the average packet delay in a node.

Figure 3.8 presents packet delay at a node obtained through both analytical and computer simulation at various congestion levels within a MANET. For the simulation results, both the average and 95 % confidence limits are presented. As seen from Figure 3.8, the analytical results are close to the simulation results. The agreement between analytical and simulation results at the node level serves to validate the simulation outputs for end-to-end delay.

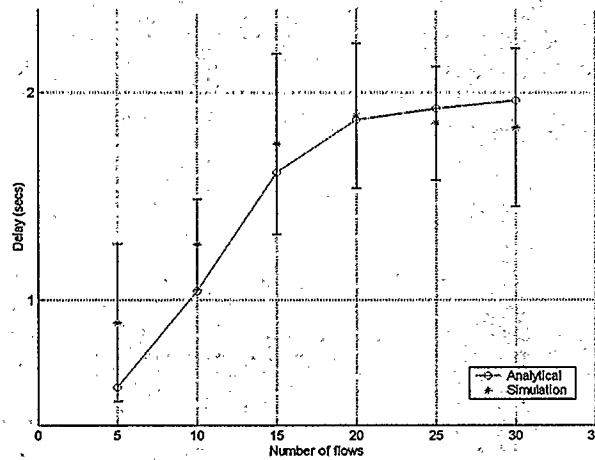


Figure 3.8: End-to-End Delay, Analytical vs. Computer Simulation

3.7 Summary

Chapter 3 introduces the QoS routing framework simulation environment. Simulation inputs and outputs are important to define the scope of the simulation process. Physical layer, medium access layer, routing, and traffic characteristics are defined in this chapter based on the specifications of IEEE 802.11 DCF mode (ad-hoc mode). NIST / AODV OPNET model and QoS routing framework OPNET model rely on the same general simulation parameters to operate, the main difference is found in the routing process. NIST / AODV relies on AODV routing protocol whereas QoS routing framework relies on a modified version of AODV based on the recommendations of the framework.

Throughput, routing efficiency, end-to-end delay, average route discovery time, routing overhead, and power consumption are the outputs obtained through the simulation. These outputs are used to evaluate the performance of the proposed QoS routing framework. Chapter 3 also provides the verification and validation of the outputs with the aim of assuring the simulation is performing as intended and the outputs are correct.

Chapter 4

QoS Routing Framework Performance

4.1 Introduction

This chapter presents results achieved through the simulation of the NIST / AODV (without QoS support) and the modified NIST / AODV based on the QoS routing framework under different network conditions such as number of flows (congestion level) and maximum speed limit (mobility level). A performance comparison with NIST / AODV is made to demonstrate the trade-offs for the QoS enabled AODV routing protocol.

4.2 QoS Routing Framework Performance Simulation Results

4.2.1 Impact of mobility on performance

Mobility leads to updates in network topology that must be addressed through efficient routing. Hence, two levels of mobility are defined: low mobility (0.1, 0.5, 1.0, 1.5, 3.0, and 5.0 m/s) and high mobility (5, 10, 15, and 20 m/s).

5 flows (CBR sources) are used in the simulation to evaluate the impact of mobility on the performance of the QoS routing framework.

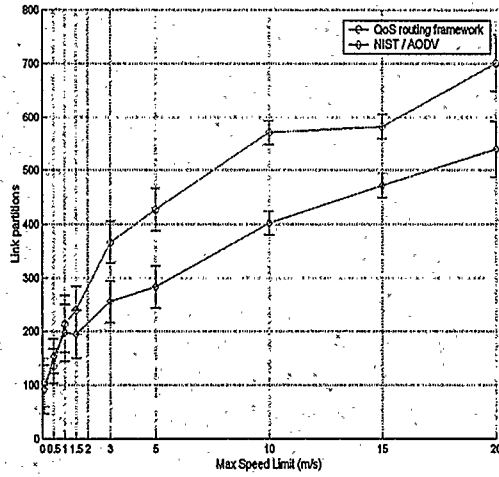


Figure 4.1: Link Partitions, Mobility

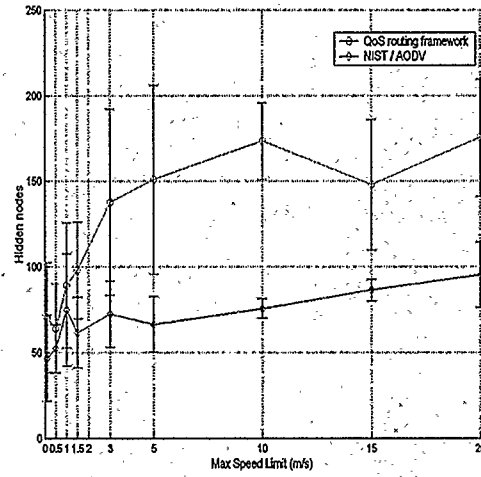


Figure 4.2: Link Partitions due to Hidden Nodes, Mobility

Figure 4.1 illustrates the total number of link partitions observed in the network due to mobility and hidden nodes. For low mobility, the number of link partitions is high for NIST / AODV and the QoS routing framework considering that in some cases the nodes are barely moving. Link partitions in low mobility are also produced by the effect of hidden nodes as illustrated in Figure 4.2. In low mobility, 0.1 m/s, the number of link partitions due to hidden nodes is close to 50 % of the total number of link partitions in Figure 4.1. The number of hidden nodes increases as the maximum speed limit increases due to the effect of the mobility model in the network. Random Waypoint mobility model predisposes the nodes to choose destinations that are either in the centre of the area, or that can be reached by traversing the centre [37]. Thus, nodes converge in the centre of the area increasing the likelihood of hidden nodes produced by the closeness among the nodes. Hence, at high speeds the nodes converge often and faster increasing the number of link partitions due to hidden nodes.

In high mobility, link partitions become more frequent in the network, not only caused by the hidden node problem but also by topology updates. The number of link partitions in both low and high mobility levels will be reflected in performance of the routing protocol (i.e., routing efficiency, route repair attempts, end-to-end delay, et cetera). For high mobility the QoS routing framework suffers more link partitions than

NIST / AODV. This is due to the utilization of multiple routes when a hidden node is causing a link partition. Once a link partition is detected, the node uses the back-up route to transmit a data packet, if the hidden node is still around the data packet will not be acknowledged and another link partition will be assumed by the MAC layer.

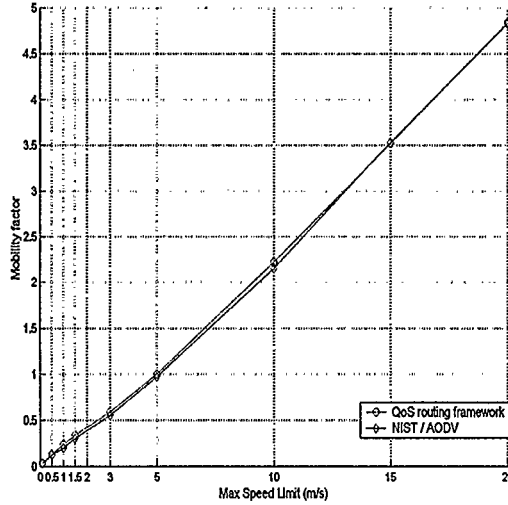


Figure 4.3: Mobility Factor

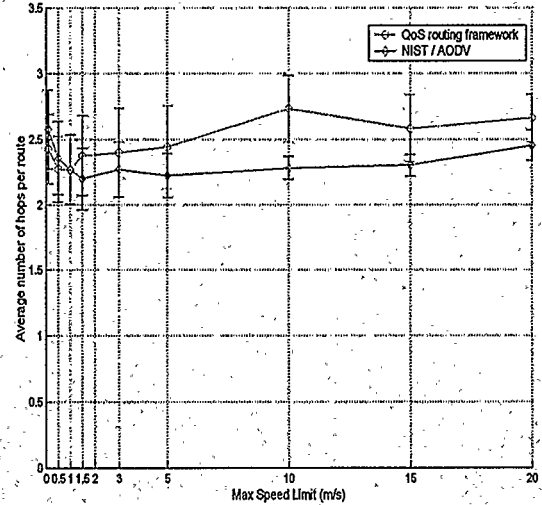


Figure 4.4: Average Number of Hops, Mobility

Node movements lead to changes in the network topology. Figure 4.3 illustrates the mobility factor of the nodes within the network. Mobility factor indicates the average node displacements during the simulation. Thus, in low mobility the nodes are continuously moving whereas in high mobility the nodes reach the destination faster and spend more time paused, consequently performing more displacements. In low mobility (0.1 m/s to 5 m/s) the nodes barely perform one movement. In high mobility (5 m/s to 20 m/s) the nodes perform from 1 to almost 5 movements during the simulation.

Figure 4.3 shows that the QoS routing framework and NIST /AODV achieved the same mobility factor due to the utilization of the same mobility model (Random Waypoint) in the simulation.

Average number of hops indicates the average path length in the routes computed during the simulation. Figure 4.4 illustrates that for NIST / AODV and the QoS routing framework the path lengths are almost the same, varying from 2 to 3 hops. QoS routing

framework exhibits longer routes than NIST / AODV due to the fact that sometimes the route that meets the QoS (bandwidth) required is not the shortest.

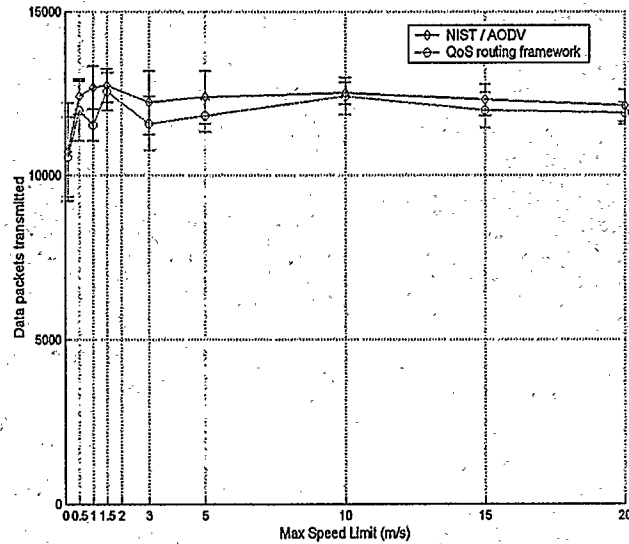


Figure 4.5: Data Packets Transmitted, Mobility

As stated earlier, 5 flows were used to evaluate the performance of the QoS routing framework under different levels of mobility. Figure 4.5 illustrates the number of data packets transmitted by the CBR sources during the simulation. It is important to state that if the source node cannot find a valid route to the destination (i.e., due to hidden nodes) data packets cannot be transmitted. Thus, the hidden node problem may affect seriously MANETs formed by source nodes continuously transmitting data packets (i.e., CBR sources transmitting real-time video).

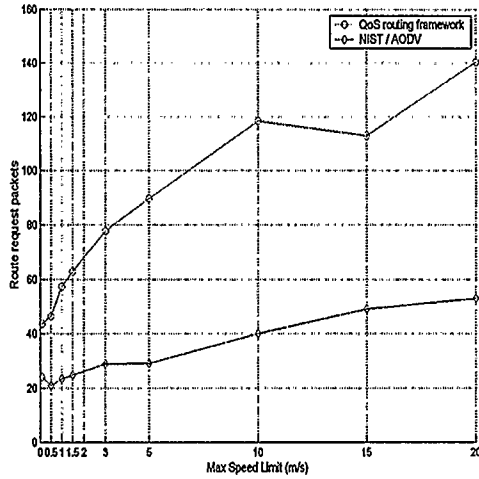


Figure 4.6: Route Request Packets, Mobility

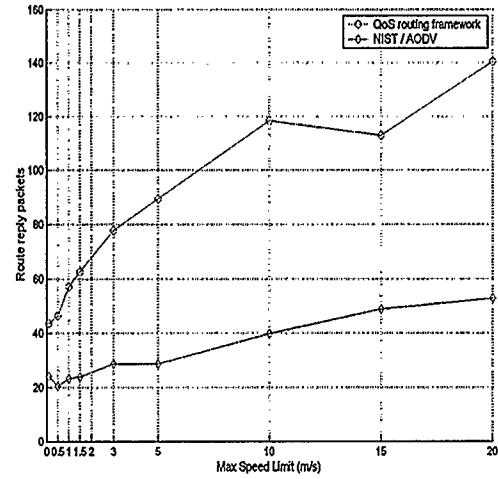


Figure 4.7: Route Reply Packets, Mobility

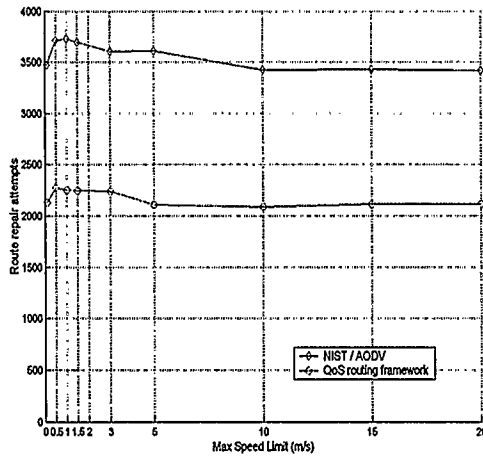


Figure 4.8: Route Repair Attempts, Mobility

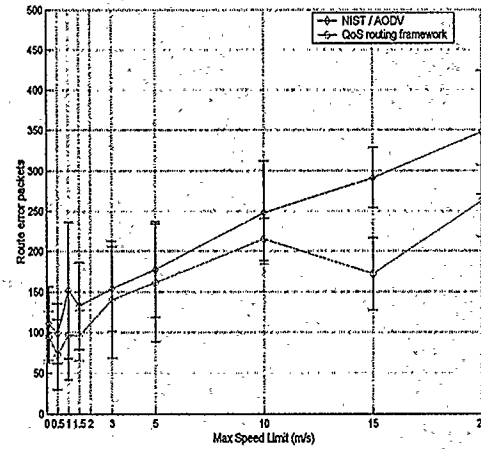


Figure 4.9: Route Error Packets, Mobility

Figures 4.6, 4.7, 4.8 and 4.9 illustrate important differences between the NIST / AODV and the QoS routing framework. Figure 4.6 shows the total number of route request packets transmitted by the nodes in the simulation (route discovery and error control process). QoS routing framework generates more RREQ packet during the simulation due to the multiple route discovery process. As explained in Section 2.4.3, intermediate nodes

allow the propagation of RREQ packets to the same destination until the maximum number of back-up routes is satisfied (2 back-up routes). Multiple route support allows the re-transmission of multiple RREQ packets to the same destination. This is illustrated by the results presented in Figure 4.6. For low mobility, both approaches increase the amount of RREQ packets due to link partitions as depicted by Figures 4.1 and 4.2. Nevertheless, the QoS routing framework exhibits dramatic increase in the RREQ when the mobility level becomes higher.

Figure 4.7 shows that the number of RREP packets in both approaches is close to the number of RREQ packets in the simulation. This is a good indicator of the efficiency of the protocol in terms of route acquisition (the number of routes requested is close to the number of routes obtained). Even with the QoS routing framework requesting routes with a QoS constraint in terms of bandwidth (64 kbps – 128 kbps) Figure 4.7 confirms that most of the QoS routes can be obtained.

Figure 4.8 shows the number of route repair attempts per simulation. It is observed that in low mobility the number of route repairs is higher than in high mobility for both approaches. Hidden node problem also affects repair mechanisms. Once a link partition is detected the node tries to repair the error (local repair or back-up route). Due to the low mobility of the nodes, hidden nodes are more frequent because the nodes stay close for longer periods and affect the repair mechanism regardless of the approach utilized. QoS routing framework outperformed the NIST / AODV due to the utilization of back-up routes. Thus, the route repair attempts are decreased dramatically. This improvement also impacts the routing overhead generated in the simulation.

Route error packets are generated if a repair mechanism fails or when a data packet arrives at a node and it does not have a valid route to the destination (route expired). The route error packets are transmitted to the source generating the data packets to indicate that the route can no longer be provided. Figure 4.9 illustrates that mobility impacts the number of route errors in the network due to the increasing link partitions.

Figure 4.10 illustrates the efficiency. At low mobility, the QoS routing framework and NIST / AODV showed poor efficiency due to the link partitions produced by hidden nodes. QoS routing framework achieves similar efficiency as NIST / AODV,

demonstrating that routing efficiency is not affected by the application of the QoS routing framework. The efficiency exhibited by NIST / AODV and consequently by the QoS routing framework is not as good as expected (85 % of the transmitted data packets were delivered at most). This is principally due to the bad performance of the 802.11 implementation used in the NIST / AODV OPNET model. Poor efficiency impacts the end-to-end delay in the network which is not adequate for QoS provisioning in case the applications are delay-sensitive.

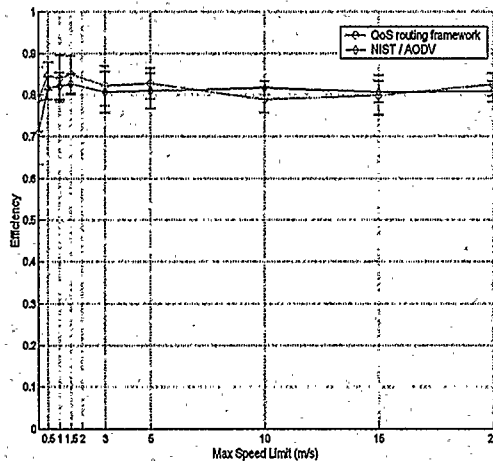


Figure 4.10: Efficiency, Mobility

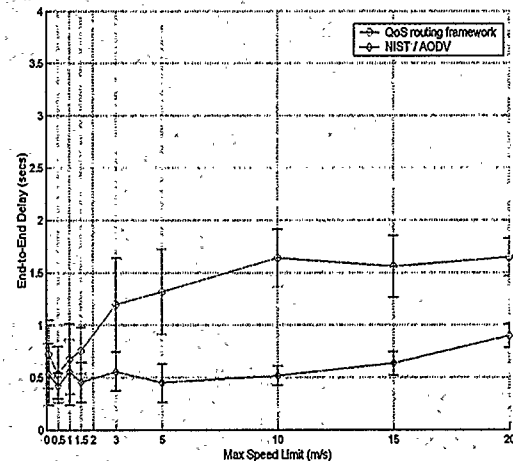


Figure 4.11: End-to-End Delay, Mobility

Figure 4.11 depicts that for low and high mobility, the QoS routing framework has higher delay than NIST / AODV. The end-to-end delay increases as the maximum speed limit increases. This behavior follows from the link partitions and the use of back-up routes to overcome these partitions. Back-up route selection process increases the end-to-end delay in the network due to invalid back-up route information (routes that no longer exist). Each node stores a maximum of two back-up routes. If the first back-up route selected is no longer valid then the node utilized a second back-up route to transmit the data flow. If the second back-up route is not valid, then the route discovery process must be re-initiated by the source or the node detecting the error in case that the local repair mechanism is enabled. NIST / AODV achieves lower end-to-end delay due to the fact that there is no back-up route selection process.

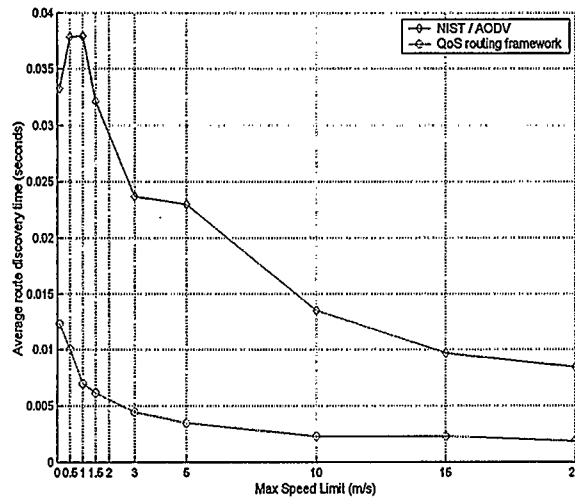


Figure 4.12: Average Route Discovery Time, Mobility

Initiating a route discovery process every time a node detects a link partition not only increases the average route discovery time but also the routing overhead in the network. Figure 4.12 shows the average route discovery time under different scenarios of mobility. It is observed from Figure 4.12 that the average route discovery time decreases as the maximum speed limit of the nodes (mobility level) increases. This is caused by the waypoint mobility model [38]: nodes moving at high velocity reach their destination points faster than nodes moving at low velocity hence, they spend more time in stationary position (pause time) than in motion during the simulation. The QoS routing framework showed a big improvement in average route discovery time in comparison to NIST / AODV because of the back-up routes feature (which reduces the number of route discoveries triggered by the route repair mechanism). This feature (back-up routes) impacts important parameters such as routing overhead and power consumption as shown in Figures 4.13 and 4.14. Figure 4.13 shows that routing overhead due to route repair attempts caused by topology updates is reduced drastically by applying the QoS routing framework. At low mobility, the routing overhead is higher due to the high rate of route repair attempts. Figure 4.14 shows that by applying routing overhead reduction techniques defined in the QoS routing framework, average power consumption per node is also reduced dramatically. For low mobility, the power consumption was higher due to

repair attempts triggered by hidden nodes. Thus, efficient medium access mechanism such as RTS / CTS to eliminate the hidden node problem can improve the performance of the routing process [35].

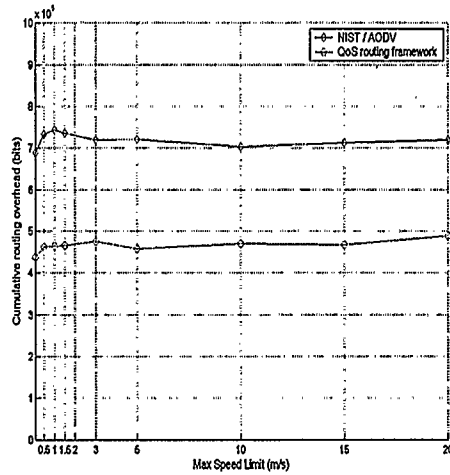


Figure 4.13: Cumulative Routing Overhead, Mobility

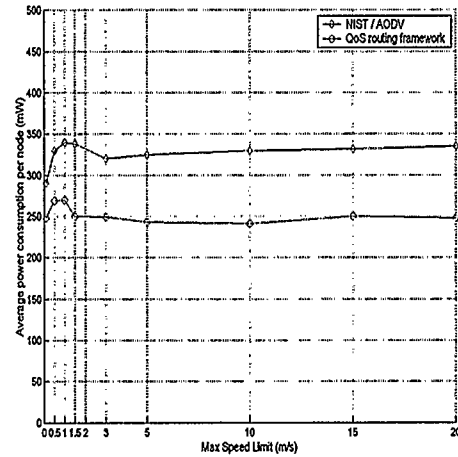


Figure 4.14: Average Power Consumption per Node, Mobility

4.2.2 Impact of congestion on the performance

Congestion impacts the overall efficiency of routing protocols in a different manner. This section analyses how the QoS routing framework minimizes the effect of congestion levels in MANETs. The number of flows used to evaluate the impact of traffic on the performance of the QoS routing framework is variable (0, 5, 10, 15, 20, 25 and 30 flows).

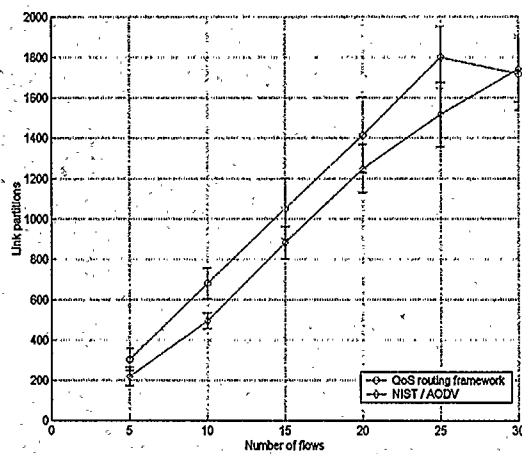


Figure 4.15: Link Partitions, Congestion

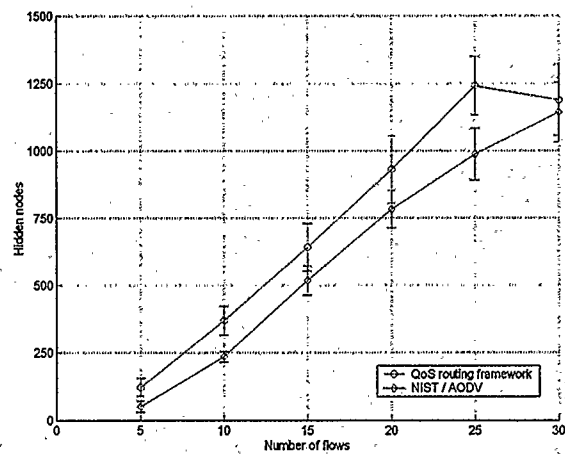


Figure 4.16: Link Partitions due to Hidden Nodes, Congestion

For simulation of congestion levels a maximum speed limit of 0.2 m/s is used. The congestion level in the network varies based on the number of flows. Figure 4.15 shows the total number of link partitions in the network (due to congestion and hidden nodes). As illustrated in Figure 4.15, even in low mobility (0.2 m/s) several link partitions occur due to topology updates and hidden nodes. Figure 4.16 illustrates the number of link partitions due to hidden nodes produced by the non-implementation of medium access mechanisms in the OPNET simulation such as RTS / CTS. The link partitions due to hidden nodes are related to the number of flows (sources transmitting packets) in the network. The number of link partitions increases as the number of CBR sources (flows) increase. This behavior is due to the higher likelihood of collisions and hidden nodes when more nodes are trying to gain channel access. QoS routing framework showed more link partitions (Figures 4.15 and 4.16) due to its fast response to link partitions (i.e., utilization of back-up routes) when a hidden node is still around (a valid route is obtained from the back-up routes list but due to the hidden node problem the data packets cannot be acknowledged).

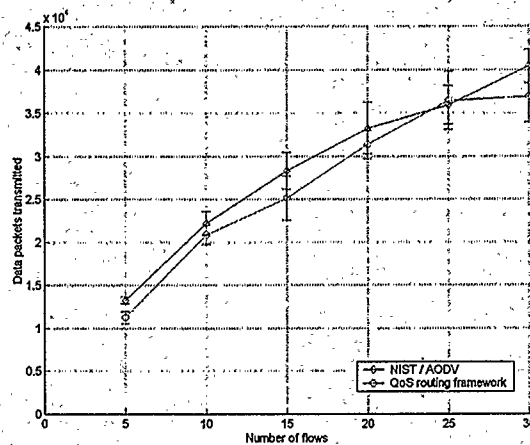


Figure 4.17: Data Packets Transmitted, Congestion

Figure 4.17 presents the total number of data packets transmitted by all the CBR sources in the simulation. The number of data packets transmitted in the network increases as the number of flows (congestion level) increases in QoS routing framework and NIST / AODV. Data packets cannot be transmitted if there is no route to the destination. Hence,

link partitions produced by hidden nodes also affect the number of data packet transmitted. Thus, the results presented in Figure 4.17 do not show a linear behavior.

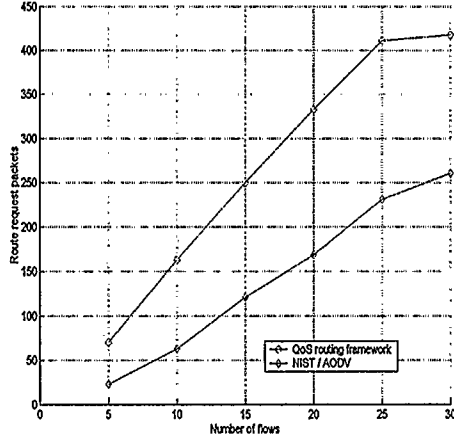


Figure 4.18: Route Request Packets, Congestion

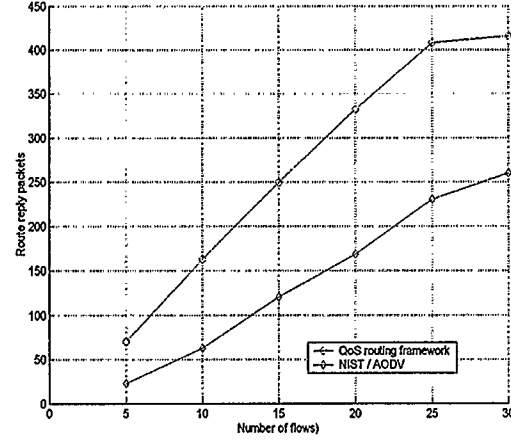


Figure 4.19: Route Reply Packets, Congestion

Figure 4.18 shows the number of route requests per simulation. The number of route requests is affected by the number of flows needing a route to the destination. The QoS routing framework shows more route request packets in the network due to the multiple route(s) discovery process. The intermediate nodes allow the propagation of route request packets from the same source to the destination until the maximum number of routes is reached (i.e., three, one primary and two back-up routes) or until a RREQ with a known value in the *First_Hop* field reaches the intermediate node. NIST / AODV only allows the propagation of one RREQ packet from one specific source to the destination.

Figure 4.19 illustrates the routing efficiency of the QoS routing framework and NIST / AODV. The number of route reply packets is really close to the number route request packets. In the QoS routing framework, a route reply packet contains a route that meets the QoS requirements specified by the source. Thus, Figure 4.19 demonstrates that almost all the QoS routes requested were obtained. NIST / AODV only obtains routes to the destination without QoS levels. The number of route requests and route replies are related by the constraint imposed by the maximum number of back-up routes supported

by each node. Thus, a source node requesting routes through 3 intermediate nodes will receive only 3 routes from the destination in the best case scenario.

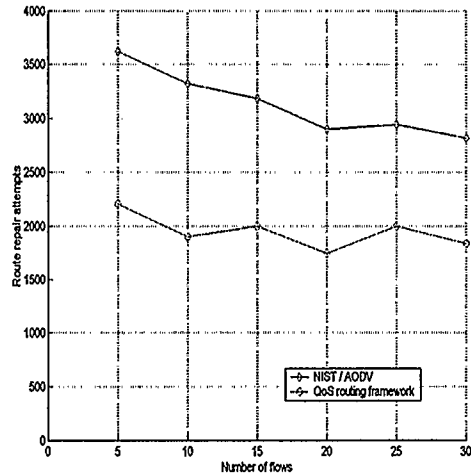


Figure 4.20: Route Repair Attempts, Congestion

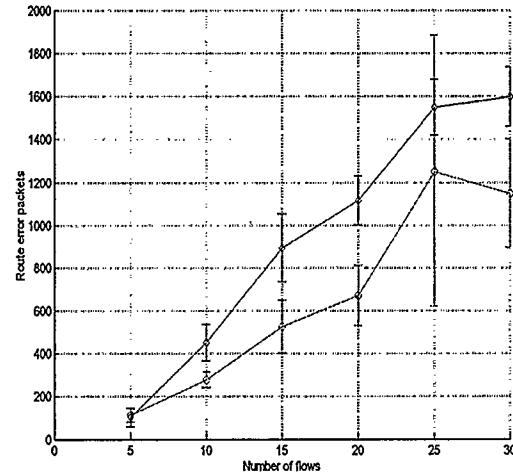


Figure 4.21: Route Error Packets, Congestion

The number of route repair attempts per simulation indicates how efficient a protocol is to overcome link partitions. Figure 4.20 shows that in all the congestion levels (5, 10, 15, 20, 25 and 30 flows) the QoS routing framework uses fewer route repair attempts than NIST / AODV. This is due to the utilization of back-up routes once a link partition is detected. NIST / AODV triggers the route repair process once a link partition occurs. Continuous route repair attempts increase the end-to-end delay, the average route discovery time, and the number of route error packets in the network. Figure 4.21 shows that in some congestion levels (10, 15 and 20 flows) NIST / AODV produced more route error packets than the QoS routing framework. Thus, the utilization of back-up routes reduces the number of route error packets in the network. Once a back-up route is selected there is no need for route error packets.

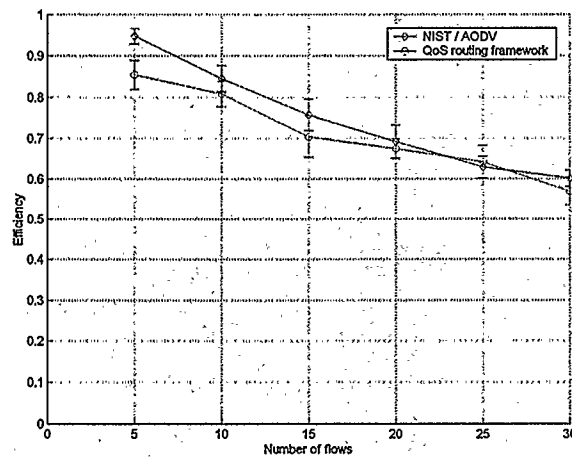


Figure 4.22: Efficiency, Congestion

Figure 4.22 illustrates the routing efficiency. Routing efficiency in low mobility (0.2 m/s) is dramatically affected by congestion in the network due to link partitions induced by the hidden node problem as seen in Figure 4.16. The efficiency of QoS routing framework and NIST / AODV decreases as the number of flows in the network increases. Even under different conditions of congestion, the routing efficiency is not dramatically affected by applying the QoS routing framework. A routing efficiency of 60 % is definitely not suitable for QoS provisioning in MANET due to the high packet loss in the network.

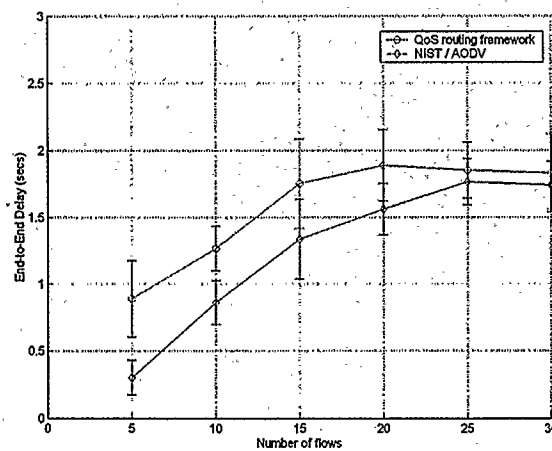


Figure 4.23: End-to-End Delay, Congestion

Figure 4.23 depicts the end-to-end delay in the network. For different traffic scenarios, the end-to-end delays obtained with QoS routing framework and NIST / AODV have similar behavior. End-to-end delay increases as the number of flows in the network increases. This similar behavior is caused by repair attempts due to collisions or hidden nodes at low mobility.

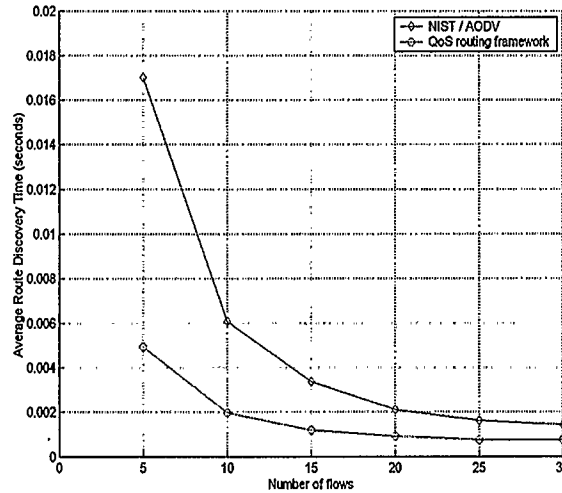


Figure 4.24: Average Route Discovery Time, Congestion

Multiple route support (back-up routes) improves the average route discovery time in the QoS enabled routing protocol as depicted in Figure 4.24. Continuous link partitions due to the hidden node lead to several route repair attempts. Hence, the nodes spend more time discovering routes to overcome the link partition. The average route discovery time in NIST / AODV and QoS routing framework decreases as the number of flows increases due to the fact that more routing information is generated within the network by the increasing number of source nodes in the MANET. Hence, the likelihood of having a route once a local repair is initiated is higher.

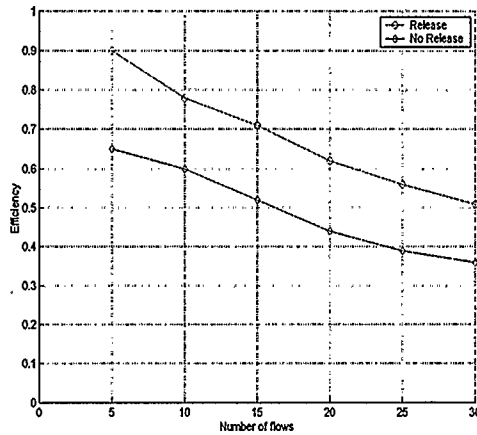


Figure 4.25: Efficiency, Resource Release

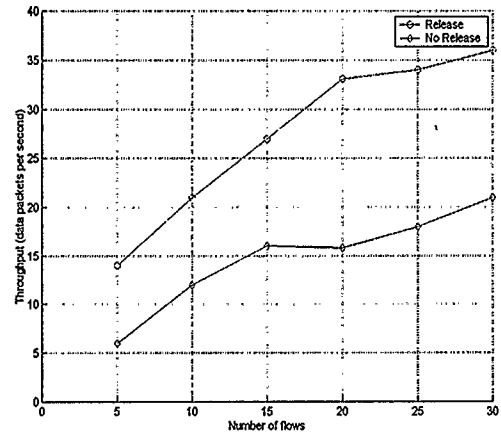


Figure 4.26: Throughput, Resource Release

To evaluate the importance of resource release mechanism in the proposed QoS routing framework two modified versions of QoS enabled AODV protocol are utilized. The first version never releases QoS resources whereas the second version releases QoS resources every time a route expires either based on a timeout parameter (6 seconds) in the routing table or in the presence of link partitions. As illustrated by Figures 4.25 and 4.26, efficiency and throughput (in data packets per second) are seriously affected if efficient mechanisms to release resources are not applied. Figure 4.26 shows that by applying resource release mechanisms, the throughput in packets per second increases. Releasing the QoS resources (bandwidth) once a route is no longer used increases the bandwidth available in the network. Hence, more QoS routes are supported in the network.

4.3 Summary

The results obtained through the simulation process show that in case of mobility important parameters such as routing overhead, average discovery time and power consumption are improved by the application of the QoS routing framework. End-to-end delay is the trade-off of the routing framework. The end-to-end delay in the network increases due to the delay introduced by the back-up route selection process. However, the routing efficiency is not affected by the QoS routing framework.

For congestion levels, the routing framework showed the same behavior as in the case of mobility. High congestion levels in MANETs increase the likelihood of collisions and hidden node problem. Hence, efficient medium access mechanisms are needed. Resource release mechanisms are also important to maintain acceptable QoS within the network. Routing efficiency and throughput are improved by applying resource release mechanisms.

The QoS routing framework improves the QoS routing process by reducing the routing overhead, average route discovery time, and power consumption. Thus, the utilization of resources (i.e., bandwidth) is performed efficiently allowing QoS provisioning.

Chapter 5

Conclusion

5.1 Thesis Conclusions

In this thesis QoS support is addressed from the network layer of the OSI model in the form of a QoS routing framework to enable QoS support in the routing process regardless of the underlying routing protocols in MANETs. The aim of the framework is to improve the routing process and enable QoS support in routing protocols that do not support it originally.

The QoS routing framework improves routing overhead, average route discovery time, and power consumption of the QoS routing process through the following processes: QoS route discovery, routing overhead reduction techniques, QoS robustness enhancement ideas, and QoS route maintenance techniques. The NIST / AODV protocol is modified based upon the QoS routing framework demonstrating that efficient local repair mechanisms improve the average discovery time by providing routes even in the presence of continuous link partitions. The application of the QoS routing framework increases the end-to-end delay in the network. Nevertheless, routing efficiency is not affected by the application of the routing framework.

The performance of routing protocols varies depending on mobility and congestion levels. Our results show that efficient routing mechanism such as multiple-route support allows fast adaptation to changes in the network topology induced by mobility. Finally, it is important to state that the QoS routing framework proposed in this

thesis contributes to achieving QoS support in MANETs by addressing and overcoming QoS routing issues from the network layer.

5.2 Suggestions for Future Work

OSI inter-layer cooperation is required to provide robust QoS provisioning in MANETs. Routing is a process that must be improved to address QoS provisioning in the network layer of the OSI model. Nevertheless, more than a single layer approach is needed to guarantee quality of service in highly dynamic networks such as MANETs. Future work must be performed in upper and lower layers to improve QoS support. Applications, medium access protocols, scheduling algorithms, and transport protocols must be QoS aware and interact to create a QoS architecture that can be robust enough to deal with the challenges posed by mobility and wireless environments in MANETs.

In this thesis the QoS routing framework was evaluated utilizing bandwidth as a QoS metric. Future work is to evaluate combined QoS metrics such as delay, jitter, or packet loss to extend the capabilities of the QoS routing framework to multimedia applications. Expansion to environments other than campus is also desirable, to evaluate the scalability of the QoS routing framework.

The evaluation of the QoS routing framework under different mobility models is useful to see if the QoS routing framework is applicable to different scenarios of mobility. Propagation effects such as shadowing and multipath fading also impact the performance of the routing process. Thus, the evaluation of the QoS routing framework through different propagation models can provide more realistic results from the simulation.

Bibliography

- [1] Wireless Communications, T. S. Rappaport, 2nd Edition, December 31, 2001. Prentice Hall.
- [2] J. Jubin, J. D. Tornow, "The DARPA Packet Radio Network Protocols", in *IEEE Proceedings*, Jan. 87, pp.21-32.
- [3] Specification of the Bluetooth System [Online]. Available <http://www.bluetooth.com/developer/specification/specification.asp>
- [4] Digital Communications: Fundamentals & Applications, B. Sklar, 2nd Edition, January 11, 2002. Prentice Hall.
- [5] CISCO, "Quality of Service Networking" [Online] Available: http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/qos.pdf
- [6] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," 1998, IETF RFC 2475.
- [7] R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: an Overview," 1994, IETF RFC 1633.
- [8] X. Hannan, W.K.G. Seah, A. Lo, K.C. Chua, "A Flexible Quality of Service Model for Mobile Ad-hoc Networks", *Vehicular Technology Conference Proceedings*, 2000. *VTC 2000-Spring Tokyo. 2000, IEEE 51st*, Volume: 1, 2000, Page: 445-449
- [9] S-B. Lee and A. T. Campbell, "INSIGNIA: An IP-Based Quality of Service Framework for Mobile Ad-hoc Networks", *Journal of Parallel and Distributed Computing (Academic Press)*, Special issue on Wireless and Mobile Computing and Communications, Vol. 60 No. 4, pg. 374-406, April 2000
- [10] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: A Core-Extraction Distributed Ad-hoc Routing Algorithm," *IEEE Journal on Selected Areas in Communications*,

Special Issue on Wireless Ad-hoc Networks, vol.17, no.8, pp.1454-1465, August 1999

- [11] G.-S. Ahn, A. T. Campbell, Andras Veres and Li-Hsiang Sun, "Supporting Service Differentiation for Real-Time and Best Effort Traffic in Stateless Wireless Ad-hoc Networks (SWAN)", *IEEE Transactions on Mobile Computing*, (Special Issue of Best Wireless Papers from IEEE INFOCOM 2002), Vol. 1, No. 3, pp. 192-207, July-September 2002.
- [12] Elizabeth M. Royer and C.-K. Toh. "A Review of Current Routing Protocols for Ad-hoc Mobile Wireless Networks." *IEEE Personal Communications Magazine*, April 1999, pp. 46-55.
- [13] David B. Johnson, David A. Maltz, Yih-Chun Hu, Jorjeta G. Jetcheva, "The Dynamic Source Routing Protocol for Mobile Ad-hoc Networks (DSR)", draft-ietf-manet-dsr-07.txt, February 21st, 2002. IETF Internet Draft (work in progress).
- [14] Samir R. Das, Charles E. Perkins, and Elizabeth M. Royer. "Performance Comparison of Two On-demand Routing Protocols for Ad-hoc Networks." *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, Tel Aviv, Israel, March 2000, pp. 3-12.
- [15] Z. Haas, M. Pearlman and P. Samar, "The Zone Routing Protocol (ZRP) for Ad-Hoc Networks," draft-ietf-mane-zone-zrp-04.txt, August 12, 2002. IETF Internet Draft (work in progress).
- [16] C.-C. Chiang, H.K. Wu, W.Liu, and M. Gerla, "Routing in Clustered Multihop, Mobile Wireless Networks," *ACM / Baltzer Wireless Networks Journal*, Vol. 1, No. 1, pp. 61-68, February 1997.
- [17] S. Chen and K.N. Nahrstedt, "An Overview of QoS routing for Next Generation High Speed Networks: Problems and Solutions," *IEEE Network*, Nov/Dec. 1998, pp. 64 – 79.
- [18] S. Chen and K. Nahrstedt, "Distributed Quality-of-Service Routing in Ad-hoc Networks," *IEEE JSAC*, vol. 17, no. 8, Aug 1999, pp. 1488 – 1505.
- [19] C.R. Lin and J.-S. Liu, "QoS Routing in Ad-hoc Networks," *IEEE JSAC*, vol. 17, no. 8, Aug. 1999, pp. 1426 – 1438.

- [20] C. Perkins, E. Royer and S.R. Das, "Quality of Service for Ad-hoc On-Demand Distance Vector (AODV) routing", July 2000, IETF Internet Draft (work in progress).
- [21] D.D. Perkins and H.D. Hughes, "A Survey on QoS Support for Mobile Ad-hoc Networks," *Wireless Communications and Mobile Computing*, no. 2, 2002, pp. 503 – 513.
- [22] A.J. Goldsmith and S.B. Wicker, "Design Challenges for Energy-Constrained Ad-hoc Wireless Networks," *IEEE Wireless Communications*, no. 4, Aug 2002, pp. 8 – 27.
- [23] C. E. Perkins, E. M. Belding-Royer, S. R. Das, "Ad-hoc On-Demand Distance Vector (AODV) Routing," draft-ietf-manet-aodv-11.txt, June 19, 2002. IETF Internet Draft (work in progress).
- [24] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *Computer Communications Review*, pp 234-244, October 1994.
- [25] S. Chakrabarti and A. Mishra, "QoS Issues in Ad-hoc Wireless Networks," *IEEE Communications Magazine*, February 2001, pp. 142 - 148.
- [26] S. Chen and K. Nahrstedt, "Distributed Quality-of-service Routing in ad-hoc networks," *IEEE JSAC*, vol. 17, no. 8, Aug 1999, pp. 1488 – 1505.
- [27] J. Broch et al, "A Performance Comparison of Multi-Hop Wireless Ad-hoc Network Routing Protocols," Computer Science Department, Carnegie Mellon University
- [28] D. Nikaein and C. Bonnet, "A Glance at Quality of Service Models for Mobile Ad-hoc Networks," Institut Eurecom, Sophia Antipolis, France.
- [29] M. K. Marina and S. R. Das. "On-demand Multipath Distance Vector Routing in Ad-hoc Networks," in *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, pages 14-23, 2001.
- [30] R. Jain, "Congestion Control and Traffic Management in ATM Networks: Recent Advances and a Survey," *Computer Networks and ISDN Systems*, vol. 28, no. 13, pp. 1723–1738, Feb. 1995.
- [31] H. Lundgren, D. Lundberg, J. Nielsen, E. Nordström, C. Tschudin, "The AODV Routing Protocol Implementation at Uppsala Univeristy [Online]" Available: <http://user.it.uu.se/~henrik/aodv/>

- [32] OPNET simulator [Online]: <http://www.opnet.com>
- [33] NIST, Wireless Communications Technologies Group, OPNET Simulation Model for the AODV MANET Routing Protocol [Online] Available: http://w3.antd.nist.gov/wctg/manet/prd_aodvfiles.html
- [34] J. Song and L. E. Miller, "Empirical Analysis of the Mobility Factor for the Random Waypoint Model," OPNETWORK 2002, Washington, August 26-30, 2002.
- [35] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LAN's," in *Proc. ACM SIGCOMM'94*, pp.212-225, 1994
- [36] J. Medhi, December 1991, "Stochastic Models in Queuing Theory", Academic Press.
- [37] E. M. Royer. "Routing in Ad hoc Mobile Networks: On-Demand and Hierarchical Strategies." *Ph.D. Dissertation*, December 2000.
- [38] J. Yoon, M. Liu and B. Noble, "Random Waypoint Considered Harmful", in *Proc. IEEE Infocom*, April 2003, San Francisco, CA.