

QoS in Node-disjoint Routing for Ad Hoc Networks

Luo Liu

Submitted for the degree of Doctor of Philosophy

Department of Electronic Engineering

Queen Mary, University of London

February 2009

To my beloved fiancé and my parents

ABSTRACT

A mobile ad hoc network (MANET) is a collection of mobile nodes that can communicate with each other without using any fixed infrastructure. It is necessary for MANETs to have efficient routing protocol and quality of service (QoS) mechanism to support multimedia applications such as video and voice.

Node-Disjoint Multipath Routing Protocol (NDMR) is a practical protocol in MANETs: it reduces routing overhead dramatically and achieves multiple node-disjoint routing paths.

Because QoS support in MANETs is important as best-effort routing is not efficient for supporting multimedia applications, this thesis presents a novel approach to provide that support.

In this thesis NDMR is enhanced to provide a QoS enabled NDMR that decreases the transmission delay between source and destination nodes. A multi-rate mechanism is also implemented in the new protocol so that the NDMR QoS can minimise the overall delays. It is shown that these approaches lead to significant performance gains. A modification to NDMR is also proposed to overcome some of the limitations of the original.

ACKNOWLEDGEMENT

I would like to express my sincere appreciation to my supervisor, Professor Laurie Cuthbert, for his supervision and friendly support. His wide knowledge, guidance and continuous encouragements have been a great help throughout my PhD study.

I would like to thank Dr. John Bigham, Dr. Karen Shoop and many others in Queen Mary who kindly have given me help and suggestions in my PhD study. Also I would like to thank the staff of the Department of Electronic Engineering - Melissa Yeo, Kok Ho Huen, Mark Sandler, Phil Willson, Sharon Cording, Theresa Willis, Na Yao and many others for their help during my studies. I would like to thank all the friends in Queen Mary for our friendship.

Finally, I would like to thank my dearest fiancé, Yilu Zhu, who gave me encouragement all the time, and my parents for their endless love and support.

Contents

Abstract	3
Acknowledgement	4
List of Figures	8
List of tables	11
List of Abbreviations	12
1 Introduction	15
1.1 Research Motivation and Objectives	15
1.2 Research Contributions	17
1.3 Author's Publications	17
1.4 Organisation of the Thesis	19
2 Routing Protocols and QoS in Mobile Ad hoc Networks	20
2.1 Mobile ad hoc networks	20
2.2 Classification of routing protocols	24
2.3 Ad hoc on-demand routing protocols	28
2.3.1 Dynamic Source Routing (DSR)	28
2.3.2 Ad Hoc On-demand Distance Vector Routing (AODV)	31
2.4 Ad hoc on-demand multipath routing protocols	32
2.4.1 Ad hoc On-demand Multipath Distance Vector (AOMDV)	32
2.4.2 Multipath Source Routing (MSR)	34
2.5 Node-Disjoint Multipath Routing protocol (NDMR)	34
2.5.1 Packet format	35
2.5.2 Route discovery	37
2.5.3 Route maintenance	41
2.6 Other node-disjoint routing protocols	41
2.6.1 Ad hoc On-demand Distance Vector Multipath Routing (AODVM)	41
2.6.2 Multiple Attempt Multipath Routing (MAMR)	43
2.7 QoS in Multipath routing for MANETs	44
2.7.1 Differentiated services (DiffServ)	44

2.7.2	Multipath QoS routing for supporting DiffServ (MQRD)	45
2.7.3	QoS in Vehicular networks using multipath routing	48
2.8	Problems with current QoS supporting multipath routing protocols	50
3	QoS-enabled NDMR	52
3.1	Introduction	52
3.1.1	RREP with queue length Field	53
3.1.2	RUP packet	56
3.2	QoS enabled NDMR OPNET Simulation Model	58
3.2.1	Network Model	58
3.2.2	Node Model	60
3.2.3	Process Model	61
3.2.4	Mobility and Traffic model	68
3.2.5	Performance Metrics	68
3.3	Simulation validation	69
3.3.1	Step-by-step validation	69
3.3.2	Overall validation	72
3.4	Two source and destination pairs	74
3.5	Simulation results	77
3.5.1	CBR source	77
3.5.2	Exponential source	85
3.5.3	Different number of sources	88
3.5.4	Different priority	90
3.5.5	Packet sequence integrity	94
3.6	Summary	95
4	QoS enabled NDMR in a Multi-rate environment	96
4.1	Introduction	96
4.2	Overview of Multi-Rate ad hoc networks	97
4.2.1	Receiver-Based Autorate (RBAR) protocol	97
4.2.2	A Relay-Based MAC (RMAC) to support multi-rate feature	98
4.2.3	High throughput reactive routing in multi-rate ad hoc networks	98
4.3	Multi-rate QoS enabled NDMR	101
4.3.1	Introduction	101
4.3.2	Multi-rate Simulation Model	103

4.3.3	Mobility and Traffic Model	105
4.4	Simulation results	105
4.4.1	CBR source (constant bit rate)	105
4.4.2	Exponential source	107
4.4.3	Different number of sources	109
4.5	Summary	110
5	NDMR in sparse networks	112
5.1	Introduction	112
5.2	Limitations of NDMR	115
5.3	Modifications for NDMR in sparse node networks	119
5.3.1	Loosening of route constraint	119
5.3.2	QoS enabled NDMR with one common node	122
5.3.3	Two common nodes	124
5.4	Summary	126
6	Conclusions and Future Work	127
6.1	Conclusions	127
6.2	Future Work	127
7	References	129

LIST OF FIGURES

Figure 2.1 Infrastructure network model	21
Figure 2.2 Infrastructure-less network model	21
Figure 2.3 single-link route	26
Figure 2.4 link-disjoint routes	27
Figure 2.5 node-disjoint routes	27
Figure 2.6 Route request propagation (DSR)	29
Figure 2.7 Route reply propagation (DSR)	30
Figure 2.8 Shortest Routing Hops of Loop-free Paths	39
Figure 2.9 Node-Disjoint Paths	40
Figure 2.10 DSCP and ECN	44
Figure 2.11 Priority Scheduler	46
Figure 2.12 Packet Delivery Ratio and Average Delay of MQRD	47
Figure 2.13 End to end delay (VANETs)	49
Figure 2.14 Packet delivery ration (VANETs)	50
Figure 3.1 Queue length in multiple node-disjoint paths	54
Figure 3.2 Simple flow chart comparing NDMR and QoS enabled NDMR	56
Figure 3.3 simple flow chart of RUP	57

Figure 3.4 Average number of hops across the network	59
Figure 3.5 Network Model	59
Figure 3.6 Node Model	60
Figure 3.7 Application Process Model	62
Figure 3.8 QoS enabled NDMR routing process model	64
Figure 3.9 MAC Process Model	66
Figure 3.10 Mobility Process Model	67
Figure 3.11 Comparison of normalized delay time	73
Figure 3.12 Comparison of packet delivery ratio	74
Figure 3.13 Two source and destination pairs with common nodes	75
Figure 3.14 Network topology diagram	76
Figure 3.15 Average delay (CBR)	78
Figure 3.16 Error bar comparison of delay time (CBR)	80
Figure 3.17 Average delay comparison	82
Figure 3.18 Single S/D delay time	82
Figure 3.19 Packet delivery ratio (CBR)	83
Figure 3.20 Packet delivery ratio comparison	84
Figure 3.21 Average delay time (exponential source)	86
Figure 3.22 Error bar comparison of delay time (exponential)	87

Figure 3.23 Packet delivery ratio (exponential source)	87
Figure 3.24 Average delay (varying number of sources)	89
Figure 3.25 Packet delivery ratio (varying number of sources)	89
Figure 3.26 Average delay of different priorities	91
Figure 3.27 Packet delivery ratio of different priorities	92
Figure 3.28 Error bar comparison of average time (different priority)	93
Figure 3.29 Error bar comparison of delivery ratio (different priority)	93
Figure 4.1 Simple topology with Multi-rate support	101
Figure 4.2 Simple topology of Multi-rate	102
Figure 4.3 Flow chart of multi-rate implementation	104
Figure 4.4 Average delay comparison (multi-rate CBR)	106
Figure 4.5 Error bar comparison of delay time (CBR)	107
Figure 4.6 Average delay comparison (multi-rate exponential source)	108
Figure 4.7 Error bar comparison of delay time (exponential)	108
Figure 4.8 Average delay comparison (different number of sources)	109
Figure 4.9 Error bar comparison of delay time (different number of sources)	110
Figure 5.1 Sufficient nodes for NDMR	113
Figure 5.2 Sparse scenario	113
Figure 5.3 Sparse scenario	114

Figure 5.4 Average delay comparison in sparse and dense network	116
Figure 5.5 Average delay in sparse network – different priority	117
Figure 5.6 Average delay of different number of nodes	118
Figure 5.7 One common node	119
Figure 5.8 Average delay of one common node (CN) path	120
Figure 5.9 Number of route breaks in sparse scenario	121
Figure 5.10 Number of route discovery times in sparse scenario	121
Figure 5.11 QoS enabled NDMR with one common node comparison	122
Figure 5.12 Two common nodes	124
Figure 5.13 Average delay time of one and two CN path	125

LIST OF TABLES

Table 2.1 Comparison of Multipath routing protocols	51
Table 3.1 Number of route breaks	54
Table 3.2 (S, D) pairs and transmission paths	76
Table 4.1 MAC delay and link cost	99

List of Abbreviations

ABR	Associativity Based Routing
AF	Assured Forwarding
AODV	Ad hoc On-demand Distance Vector
AODVM	Ad hoc On-demand Distance Vector Multipath Routing
AOMDV	Ad hoc On-demand Multipath Distance Vector
AP	Access Point
AQOR	Ad hoc QoS On-demand Routing
BE	Best Effort
BQ	Broadcast query
CBR	Constant Bit Rate
CSMA	Carrier Sense Multiple Access
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
CTS	Clear To Send
CN	Common node
DARPA	Defence Advanced Research Projects Agency
DCF	Distributed Coordination Function
DCLQ	Distributed Cross-Layer QoS
DIFS	DCF IFS
DiffServ	Differentiated Service
DSCP	DiffServ Code Point
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
ECN	Explicit Congestion Notification
EDCF	enhanced DCF
EF	Expedited forwarding
ETSI	European Telecommunications Standards Institute
FQMM	Flexible QoS Model for MANETs
FSR	Fisheye State Routing
GSR	Global State Routing

HCF	Hybrid Coordination Function
IEEE	Institute of Electrical and Electronics Engineers
IFS	Inter Frame Spacing
IP	Internet Protocol
LAN	Local Area Networks
LQ	Localized Query
MAC	Medium Access Control
MAMR	Multiple Attempt Multipath Routing
MANET	Mobile Ad hoc Networks
MQRD	Multipath QoS Routing for supporting DiffServ
MRL	Message Retransmission List
MSR	Multipath Source Routing
MT	Mobile Terminal
NAV	Network Allocation Vector
NDMR	Node-Disjoint Multipath Routing
OSI	Open System Interconnection
PCF	Point Coordination Function
PDA	Personal Digital Assistant
PHBs	Per-Hop Behaviours
QoS	Quality of Service
RD	Route Delete
RED	Random Early Detection
RN	Route Notification
RRCM	Route Conformation Message
RREP	Route Reply packet
RREQ	Route Request packet
RSVP	Resource reservation Protocol
RTS	Request to Send
RTR	Request to Relay
RTT	Round Trip Time
RUP	Route Update Packet
SIFS	Short IFS

SLA	Service Level Agreement
SNR	Signal Noise Ratio
TTL	Time to Live
WLAN	Wireless Local Area Network
WRP	Wireless Routing Protocol

1 INTRODUCTION

1.1 Research Motivation and Objectives

The properties of mobile ad hoc networks (MANETs) are that they have multihop wireless connectivity, frequently changing network topology and need efficient dynamic routing protocols. Nodes are free to move, which means that the network topology will be changing dynamically. These factors present a big challenge to designers of robust and efficient routing protocols for ad hoc networks, because dynamic routing protocols need to find the best routes efficiently among moving nodes.

On-demand routing protocols are generally used since they consume less routing load (in terms of signalling traffic) than proactive protocols [15]. Probably the two most widely studied on-demand ad hoc routing protocols are Dynamic Source Routing (DSR) [5] and Ad Hoc on-demand Distance Vector (AODV) [2]. However, they both have a limitation in that they build and relay on a single path route for every active source-destination pair. Therefore, if a link breaks on the route, both protocols need to start a new route discovery process and this is the main reason why they have a high routing overhead.

To alleviate these problems on-demand multipath routing protocols are used: they establish multiple routes between source nodes and destination nodes during one route discovery process. Only when all the routes have failed, or if only one route is available, a new route discovery process will be initiated.

In this thesis, an approach built on the Node-Disjoint Multipath Routing Protocol (NDMR) [12], but one that allows for QoS to be controlled, is introduced. Compared with other on-demand multipath protocols,

NDMR has two novel aspects: it supports multiple node-disjoint routes and significantly decreases routing overhead [12].

The need to control QoS is because best-effort services cannot meet the requirement in many situations; for example, multimedia applications in a commercial environment and mission-critical applications for the military. Supporting QoS in mobile ad hoc networks has, therefore, become an important area of research. In general, these applications need new capabilities to guarantee low delay time, high packet delivery ratio and high bandwidth availability.

DiffServ [9] is a standard approach to support QoS in IP networks. It can be applied in MANETs to offer QoS support because it is able to minimize signalling requirements. Multipath QoS Routing protocol for supporting DiffServ, short for MQRD [13], combines the advantages of NDMR and DiffServ. It classifies network traffic into different priority levels, and deploys queuing management and priority scheduling to obtain QoS supports. However, if the delay of the current path cannot meet the SLA requirements, NDMR would not be able to change to another path.

In this thesis, a novel idea - QoS enabled NDMR is presented. It achieves QoS support allowing paths to be changed dynamically among multiple routes for a source-destination pair as network conditions vary.

Recent IEEE 802.11 MAC protocols provide a multi-rate capability for wireless LANs and ad hoc networks. Data packets can be transmitted at different rates according to different channel quality. In this thesis, a multi-rate transmission is implemented based on the QoS enabled NDMR on the physical layer to minimize overall delays.

1.2 Research Contributions

The work reported in this thesis is novel. Multi-rated QoS enabled multipath routing of mobile ad hoc networks are addressed, the main contributions being:

- A novel adaptation of NDMR, QoS enabled NDMR, is proposed in this thesis to overcome the shortcomings of the current NDMR routing protocol. It provides QoS support by introducing an additional field in packets for MANETs to decrease transmission delays. (chapter 3).
- Based on the QoS enabled NDMR, multi-rate transmission is implemented on the physical layer so that packets can be transmitted at different bit rate according to the channel quality and further minimize overall delays. (chapter 4).
- Limitations of Node-Disjoint Multipath Routing (NDMR) protocol are presented. This work shows that NDMR would lose its advantage when the network density is low. This thesis also suggests some improvements for NDMR to overcome the limitations in sparse node networks by allowing common nodes in NDMR. (chapter 5).

1.3 Author's Publications

[Liu-1] L. Liu and L. Cuthbert, "QoS in Node-disjoint Routing for Ad Hoc networks", in proceedings of Networking and Electronic Commerce Research Conference (NAEC 2006), Riva Del Garda, Italy, October, 2006.

[Liu-2] L. Liu and L. Cuthbert, "QoS in Node-disjoint Routing for Ad Hoc networks", in proceedings of Wireless Communications,

Networking and Mobile Computing (WiCom 2007), Shanghai, China, September, 2007.

[Liu-3] L. Liu and L. Cuthbert, "QoS in Node-disjoint Routing for Ad Hoc networks", in proceedings of 4th ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (4th ACM PE-WASUN 2007), Chania, Crete Island, Greece. October, 2007.

[Liu-4] L. Liu and L. Cuthbert, "QoS in Node-disjoint Routing for Ad Hoc networks", International Journal of Communications, Network and System Sciences, Vol.1, No.1, February, 2008.

[Liu-5] L. Liu and L. Cuthbert, "A novel QoS in Node-disjoint Routing for Ad Hoc networks", in proceedings of 13th ICC IEEE International Workshop on Computer-Aided Modeling, Analysis and Design of Communication Links and Networks (13th ICC CAMAD 2008), Beijing, China, May, 2008.

[Liu-6] L. Liu and L. Cuthbert, "Multi-rate QoS enabled NDMR for Mobile Ad Hoc networks", in proceedings of 2008 International Conference on Computer Science and Software Engineering (CSSE 2008), Wuhan, China, December, 2008.

[Liu-7] L. Liu and L. Cuthbert, "Multi-rate QoS enabled NDMR for Mobile Ad Hoc networks", in proceedings of 2nd International Conference on Signal Processing and Communication Systems (ICSPCS 2008), Gold Coast, Australia, December 2008.

1.4 Organisation of the Thesis

The thesis is organized as follows:

Chapter 2 gives the background of the research, including a brief summary of mobile ad hoc networks focused on on-demand routing protocols together with the previous research work on NDMR routing protocol. Quality of Service (QoS) and Differentiated Services (DiffServ) are also introduced in this chapter.

In chapter 3, a novel adaptation of NDMR to give QoS support for ad hoc routing – QoS enabled NDMR is proposed. The important components such as route update process, selecting paths are explained. This chapter also describes the network implementation and simulation models using in OPNET and the simulation results.

Chapter 4 introduces multi-rate on physical layer into the QoS enabled NDMR. This enhancement can minimize overall delays when the transmission rate changes as the distance between the nodes varies. The comparing simulation results are shown.

The limitation of NDMR is described in chapter 5. NDMR and QoS enabled NDMR will lose its advantages when the network density is low and, in this chapter, simulation results showing the extent of the problem are discussed, together with the underlying reasons, and some improvements are suggested.

All the work in this thesis is reviewed and concluded in Chapter 6. Some suggestions are made as to how the work could be extended.

2 ROUTING PROTOCOLS AND QoS IN MOBILE AD HOC NETWORKS

Finding a route in wireless ad hoc networks is much more complicated than in traditional infrastructure networks: the routing protocol needs to take into account many factors, such as dynamically changing topology and how to minimise the routing overhead; moreover, a route must be built efficiently and rapidly. This is in addition to finding the best routing path, a feature that is common with traditional networks.

Usually there are fewer accessible resources in mobile ad hoc networks compared with infrastructure networks so that a more efficient routing protocol is needed. Specially designed routing protocols are essential in these networks to achieve routing stability under these highly dynamic conditions.

Node transmission range is limited in wireless mobile ad hoc networks so that nodes may not be able to directly communicate with each other and data packets need to be relayed by intermediate nodes to reach the destination. An ad hoc routing protocol is used to dynamically find and, importantly, maintain up-to-date routes between communicating nodes because there is no fixed topology or fixed routes.

2.1 Mobile ad hoc networks

There are two architectures in wireless mobile networks: infrastructure and infrastructureless. The first one is dependent on fixed equipment such as base stations or access points (AP) to connect mobile terminals (MTs) to the wired infrastructure, as illustrated in Figure 2.1. When a source MT wants to establish a conversation with another MT, it does not need to know routes between each other for the source MT will establish routes with the base station first.

However, ad hoc, the second approach does not contain any fixed infrastructure. All nodes in a mobile ad hoc network can be dynamically connected to each other and are free to move. All nodes in the network are hosts and routers as well.

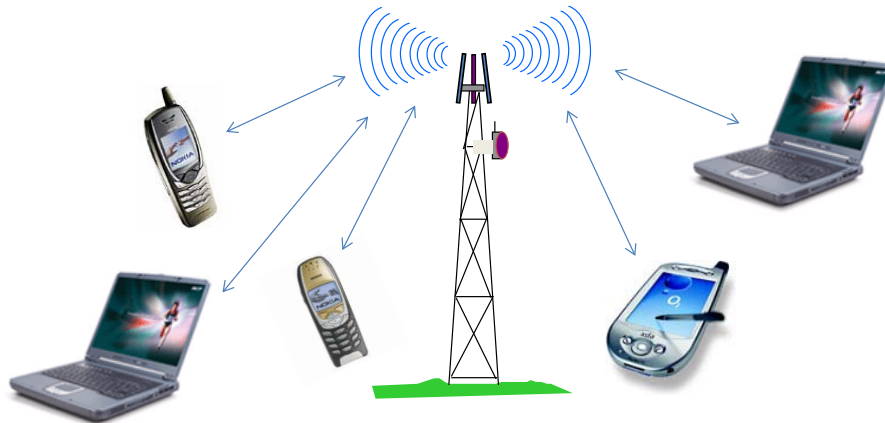


Figure 2.1 Infrastructure network model



Figure 2.2 Infrastructure-less network model

Figure 2.2 is an example of a three-node ad hoc network. If a source node S wants to send data packets to a destination node D which is outside the transmission range of S . Therefore, S needs a relay node R to forward packets to D . R acts as a host and a router at the same time.

Ad hoc networks usually have lower available resources compared with infrastructure networks and the highly dynamic nature of ad hoc

networks means that many special factors have to be considered when designing a routing protocol specialized for them, such as network topology, routing path and routing overhead; also it must find a path quickly and efficiently.

The transmission range of each node is limited in wireless ad hoc networks and thus not all nodes can directly communicate with each other. A node is often required to relay packets to another node to accomplish a communication across the network. An ad hoc routing protocol must dynamically establish and maintain routes between source and destination nodes for there is no static network topology and fixed routes.

Some people believe that ad hoc networks will be the technology for the next generation wireless communication and this has led to a great deal of research in the subject. The advantages of ad hoc networks are:

- Ad hoc networks can be built fast, easily and dynamically. Because there is no fixed infrastructure in the network, no base station is needed, so the network is more flexible, and the nodes in ad hoc network are free to move; moreover they are self configured.
- Ad hoc networks can solve particular problems where it is difficult to connect infrastructure nodes such as, the deployment of a wireless network in historic building or in conferences and trade shows. All that needs to be done is to put in nodes and let them establish communication links between themselves.
- Another important aspect of ad hoc networks is that they are spectrally efficient: as the nodes can communicate directly without the help of base station, more routes can be used for one node to communicate with another. Also, any node can act as a host and a

router at the same time. This reduces the congestion in the wireless network and saves the limited bandwidth resources.

All these advantages make ad hoc networks suitable for many applications, such as:

- **Military applications:** In the modern battle fields, it is very important to keep contact between soldiers, military vehicles, and headquarters or among military vehicles themselves when they are moving. An ad hoc network can perform well in this environment, because it can build a network fast and easily; more importantly, any damaged node will not affect the overall communication.
- **Disaster Recovery:** In a disaster area (such as after a hurricane or earthquake) the basic communication networks are often damaged so that ad hoc technology can be used to build a temporary communication network very quickly. This is very important as lack of a communications infrastructure hampers rescue and recovery operations.
- **Conferences and meetings:** In environments without fixed communication equipments, ad hoc networks can help to set up the communication very quickly between the people who are attending the conference or meeting without any need for a fixed infrastructure.
- **Personal networks:** PDAs, laptops and home appliances can be connected with the help of ad hoc technology at home, making it easy to connect, yet retaining the flexibility to move equipment around.

- Sensor networks: sensor networks consist of tiny, cheap, sensor nodes that are deployed in the area of interest. They are used for applications such as environment monitoring, scientific observation and industrial sensing and often use an ad hoc network structure to allow information to be collected from the sensors for analysis or to be transmitted back to some external point. The special feature of sensor networks is that the sensors are often limited in resources, especially power.

2.2 Classification of routing protocols

An ad hoc network needs a routing protocol to allow information to be transmitted from one node to another. Generally, there are three types of routing protocol: hybrid, proactive and reactive (on-demand).

Hybrid routing, usually called balanced-hybrid routing, combines distance-vector routing and link-state routing. Hybrid routing protocols use distance-vectors to determine best paths to destination nodes, and transmit back routing information only when the network topology changes. An example of a hybrid routing protocol is the Enhanced Interior Gateway Routing Protocol (EIGRP) [79], developed by Cisco.

All nodes in proactive protocols are required to record all routes to the destinations so that the source node knows the route and the route can be immediately used when a source node generates a packet to a destination. All topology changing will be propagated across the network to notify each node record the changing. There are several examples of proactive routing protocols such as: “global state routing (GSR)” [16], “destination-sequenced distance-vector (DSDV)” routing [17], “wireless routing protocol (WRP)” [18], and “fisheye state routing (FSR)” [20].

For reactive (on-demand) protocols, route building happens only when a source node requires communication with a destination node; there are no predetermined routes in the network. If a source node wants to send a packet to a destination node, but there is no route to the destination, the source node will initiate a route discovery process to build a communication route. After the route is established, a maintenance procedure will take place for route continuance until the route breaks. “Dynamic source routing” (DSR) [5], “ad hoc on-demand distance vector routing” (AODV) [2], and “associatively based routing” (ABR) [21] are examples of on-demand routing protocols.

Proactive protocols generally have the advantages that new communications are able to get available routes immediately, rather than waiting for a route discovery process. However they do have the drawback of an extra control overhead, to keep and update routing information across the network. To reduce this limitation, on-demand protocols initiate a route discovery only when necessary. Since on-demand protocols have less control overhead than proactive protocols, they normally require less bandwidth, although delays would take place when building a route to a destination. However, excessive traffic could be generated if route discovery is regularly needed because on-demand protocols need to broadcast route requests, and so lose that advantage over proactive protocols.

As explained above, a design constraint in ad hoc networks is often to reduce the resources required, in order to limit resources and have a lower routing load and on-demand routing protocols are commonly used.

However, when there is a QoS requirement, the delay in setting up a route, or in rebuilding one if the route disappears, can adversely affect the QoS so that this will be an important aspect to be considered.

Several on-demand protocols will be described in the next section.

As there are many nodes in the mobile ad hoc networks, routing paths between each source and destination nodes are different. Therefore, routing protocols could be classified as single-link, link-disjoint and node-disjoint routes according to the number of common nodes or links of each path; the differences are explained here.

As shown in Figure 2.3, for the yellow source-destination pair, there is only one route between them - called a single-link route.

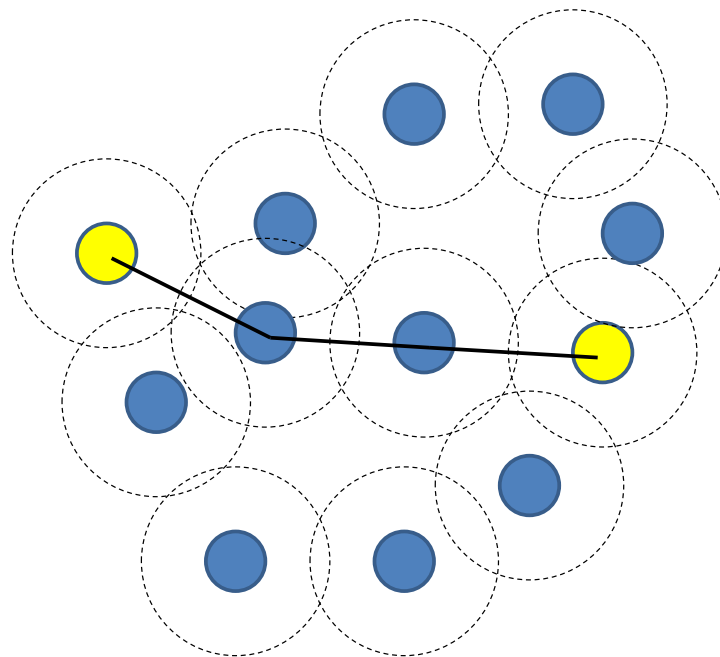


Figure 2.3 single-link route

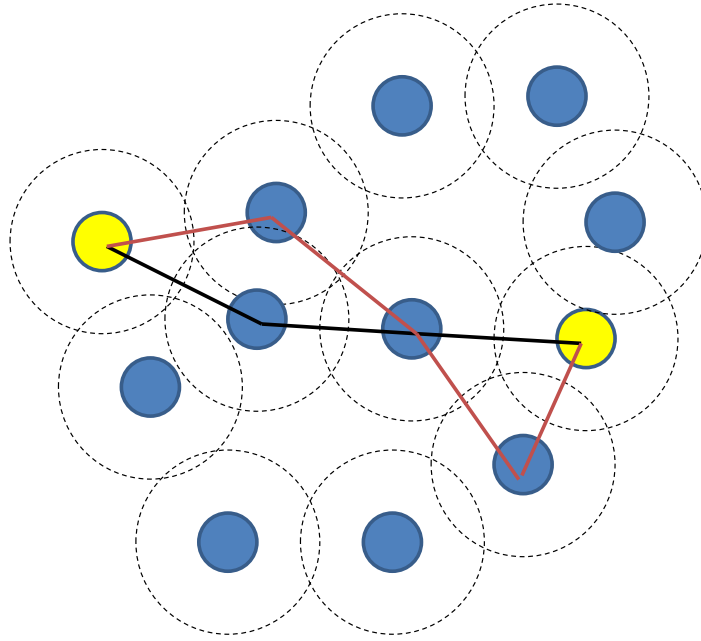


Figure 2.4 link-disjoint routes

In Figure 2.4, there are two routes for the yellow source-destination pair: the black one and red one. There is no common link (two or more adjacent nodes) among these two routes. In this case, these two routes are called link-disjoint routes.

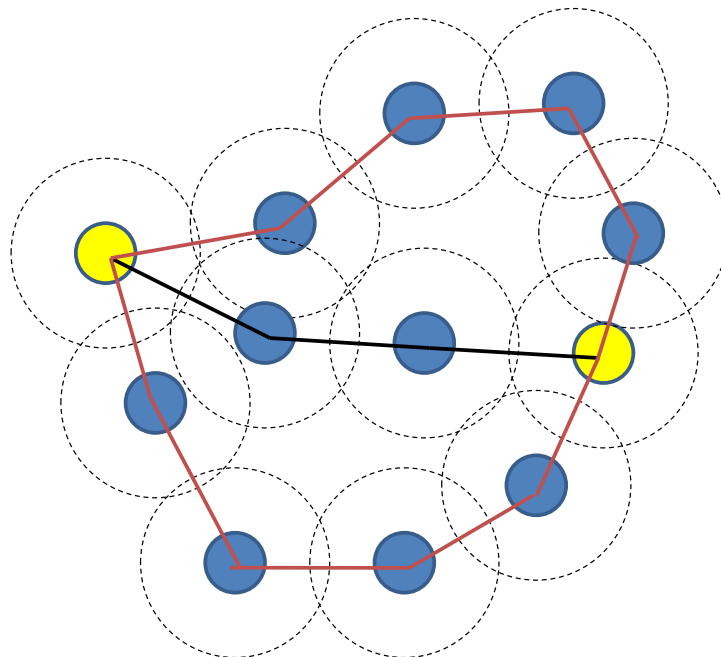


Figure 2.5 node-disjoint routes

In Figure 2.5, there are 3 routes for the yellow source-destination nodes. We can see from the figure, there are no common nodes besides source and destination nodes among these 3 routes, so these 3 routes are node-disjoint routes for that source-destination pair.

2.3 Ad hoc on-demand routing protocols

There have been many routing protocols proposed [2, 5, 17, 19, 22, 23 and 24]. Those described here illustrate the principles of single-link protocols, but they will not be discussed in great detail since the focus of this thesis is on node-disjoint rather than single-link routing.

2.3.1 Dynamic Source Routing (DSR)

The concept of source routing is introduced in dynamic source routing (DSR) [5] as a source node shows the whole routing list in the header of a data packet. Nodes learn the routes to destinations and keep records of those routes in a cache, so that, if a particular route is still in the cache when the source wants to send to that destination, it does not have to rediscover the route.

If there is no route stored in the cache, the source has to start the route discovery procedure. It does this by broadcasting a route request message (RREQ) that is essentially asking other nodes to pass through the network to the destination. The RREQ packet includes a route list, a sequence number and the address of the source node and destination node. The purpose of the route record is to keep a record of the route hops that the RREQ packet transmitted.

When another node receives an RREQ packet, to prevent routing loops, it will confirm its own address is in the route list of the RREQ or not and if so it drops the RREQ.

The intermediate node will affix its address to the route list if the node's address is not included, and then rebroadcasts the RREQ to its neighbours across the network. Figure 2.6 is the RREQ transmitting procedure across the network.

After the destination node receives the RREQ, it recognises from the address in the message that the message is for it and sends back to the source a new message, a route reply message (RREP), containing the route as discovered with its own address appended (Figure 2.7). It knows the route from the sequence of nodes in the message and if it receives more than one route request message via different routes it can choose the shortest.

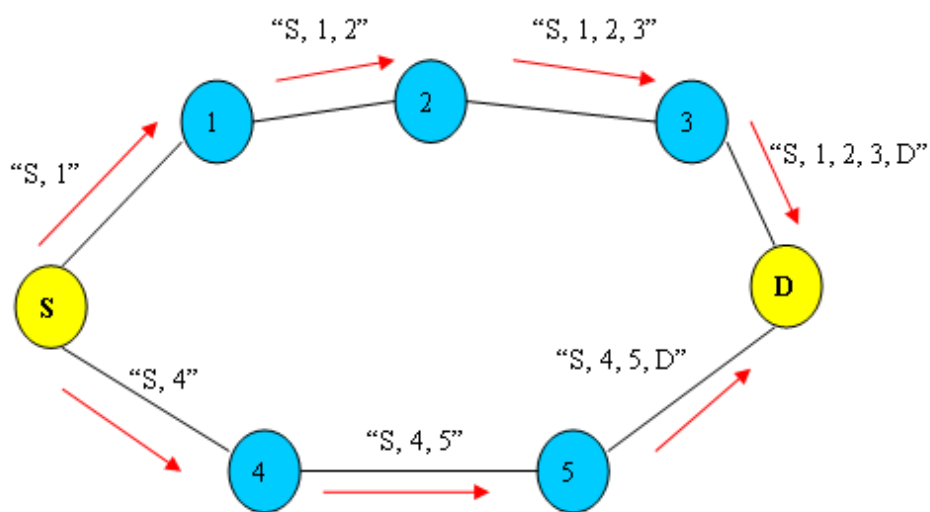


Figure 2.6 Route request propagation (DSR)¹

¹ From [5]

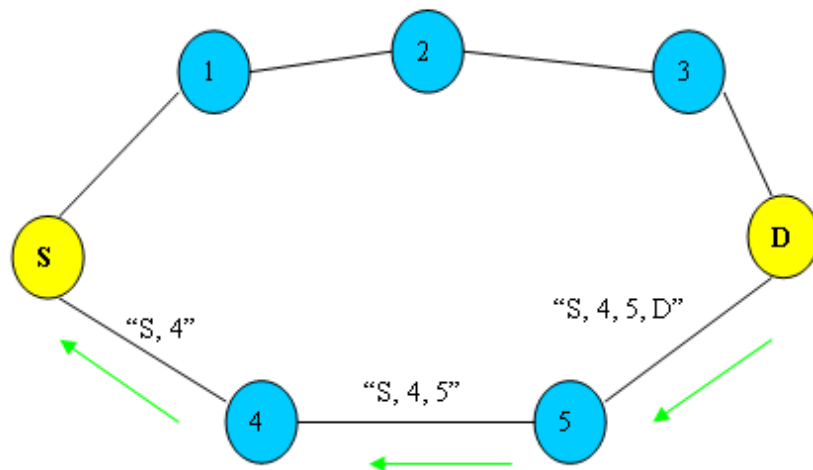


Figure 2.7 Route reply propagation (DSR)²

Route maintenance is used to cope with failures. A node will create a route error packet to the source node if a link failure is found. The route error packet contains the broken node address and its own address.

After the source node receives a route error packet, it knows where the failure is and clears from the cache those routes that are affected by the failure. It will initiate a new route discovery process if needed.

Acknowledgements can also be used to confirm that links are operating correctly.

To reduce routing overhead of searching for a new route, intermediate node can send route reply packet to the source node if it already has an up-to-date route to the destination.

² From [5]

2.3.2 Ad Hoc On-demand Distance Vector Routing (AODV)

Ad hoc on-demand distance vector (AODV) routing [2] implements the destination sequence number used in “destination-sequenced distance-vector routing” (DSDV) [17] and a modified route discovery process in DSR [5].

If a source node wants to send a data packet but there is no existing route to the destination, it will initiate a route discovery process by broadcasting a route request (RREQ) packet to all of its neighbours. In that message there is the source and destination addresses and the most recent value that the source has of a sequence number for the destination. Nodes receiving the RREQ would forward that to their neighbouring nodes until the RREQ packet arrived at the destination node or there is an up-to-date route to the destination in an intermediate node – how it determines whether it is “up-to-date” is given later. These intermediate nodes can see the route the packet has taken to reach them and so can set up route information to that source in their own tables, as well as adding their information to the RREQ before forwarding the packet.

A node receiving the RREQ may send an RREP if it is either (i) the destination or (ii) if it has an up-to-date route to the destination, “up-to-date” being defined as having a destination sequence number greater than or equal to the sequence number in the RREQ.

If one of these conditions is true, the node sends an RREP back to the source. Nodes prevent looping by discarding any RREQ they have already processed.

Intermediate nodes will set up forward path entries in their routing tables at the time when the RREP packet is sent back along the reverse path.

If the source later receives an RREP containing a greater sequence number or the same sequence number but with a smaller hopcount, it can update its route table.

Link maintenance is done with link failure notifications initiated when an intermediate node detects it cannot reach its downstream neighbour. If a link failure happens, the node noticing the failure will send a link failure packet to all upstream neighbours, which these too pass backwards the reverse path until it reaches the source node.

2.4 Ad hoc on-demand multipath routing protocols

AODV and DSR are on-demand routing protocols in MANETs that rely on a single route path between a source and destination pair. The limitation of both of them is that, because of the single path, if there is a link break on the route, both of the protocols must initiate a new route discovery process and that is the main reason for these protocols having a high routing load.

It is more efficient to establish multiple paths from a source node to a destination node during one route discovery process as the multiple paths can cope with dynamic network topology changes in the ad hoc network.

This section will introduce some multipath routing protocols designed for mobile ad hoc networks. There is one routing protocols based on AODV [2]: AOMDV [19], and one that is based on DSR [5]: MSR [22].

2.4.1 Ad hoc On-demand Multipath Distance Vector (AOMDV)

Ad hoc On-demand Multipath Distance Vector (AOMDV) [19] is a protocol supporting multiple loop-free and link-disjoint paths set up through a multipath addition to AODV protocol. Loop-freedom is

guaranteed by using an invariance based on a new concept of “advertised hopcount”. Using a special flooding algorithm, link-disjointness of multiple paths is achieved in AOMDV.

Calculating multiple paths during the route discovery process is the main idea in AOMDV. A feature of AOMDV is reusing the routing information available in the AODV protocol as much as possible. The routing paths to each destination have a list of next-hop node and the hop counts respectively for building multiple routes between a source and destination nodes. Each destination node keeps information of the maximum hop count of multiple paths, described as advertised hopcount. If a node receives a copy of route request packets, it will be added in the route table as an alternative route to the destination. The destination node only accepts an alternative route with lower hop count than the advertised hop count so that loop freedom is guaranteed.

Link-disjointness is also achieved in AOMDV. RREQ packet holds an extra “first hop” field to show the first hop after the source node. Each intermediate node does not immediately drop duplicate RREQs in order to establish link-disjoint routes. A first hop list of each RREQ is kept by each node to record neighbours addresses which received copies of the RREQ of the source node. The destination node replies to duplicate RREQ packets in spite of their first hop with the aim of establishing multiple link-disjoint paths. However, the destination node only responds RREQ packets coming from a different last hop (the hop before destination node) to guarantee link-disjointness for the first hop of the RREP packet.

2.4.2 Multipath Source Routing (MSR)

Multipath Source Routing (MSR) [22, 24] is a protocol based on DSR [5] that achieves multipath routes. It is based on the measurement of RTT and focuses on the adaptive distribution of load between multiple paths across the network. MSR improves performance by using multiple paths.

MSR applies the same route discovery mechanism in DSR but multiple paths can be returned. Each route discovered is stored in the route cache with a unique route index where is trouble-free to select multiple routes. After receives a RREQ packet, an intermediate node appends its own address to the route record in the header of RREQ packet. After the RREQ packet arrives at the destination node, an RREP packet will be generated and transmitted back through the reverse route of the RREQ.

Path disjointness is supported in MSR as independence among routes is very important in multipath routing. In MSR, looping can be immediately detected and eliminated for it uses the identical route discovery process in DSR.

All path calculation is done in the source node. In MSR, source nodes are also responsible for load balancing. A special table in the source containing multiple path information to the specific destination: the destination node, the delay time, the load distribution weight of a route and also the path index.

2.5 Node-Disjoint Multipath Routing protocol (NDMR)

Several existing multipath routing protocols and their benefits were introduced in the previous section. However, they still have some problems while they build on-demand multiple routing paths.

During the process of building multiple disjoint paths, they will generate a broadcast storm. The source node will flood the RREQ packets across the whole network if it wants to build a communication with a destination but no route exists. All the multipath routing protocols in the previous section do not drop duplicate RREQ packets at intermediate nodes so that the destination node is able to select disjoint routes. This results in a huge amount of routing overhead and routing load across the entire network. A challenging issue is how to reduce routing overhead when planning routing protocols due to the restricted resources in wireless ad hoc networks.

All the multipath routing protocols above do not take actions of how to achieve multiple node-disjoint routing paths or how to reduce routing overhead. Next, a Node-Disjoint Multipath Routing Protocol with low control overhead to solve these problems, Node-Disjoint Multipath Routing (NDMR) proposed by Li [14], will be described in detail.

NDMR is an extension and modification of AODV and the path accumulation aspect of RREQ packets of DSR is also included to efficiently establish multiple node-disjoint routing paths with limited broadcast overhead and reduced routing delay time.

2.5.1 Packet format

There are five types of control packets (RREQ, RREP, RERR, RERR_1, and HELLO), and one type of data packet in NDMR.

- RREQ (Type, Src, SrcSeqNum, Dest, DestSeqNum, BroadcastID, TTL, last_hop, RoutePath): The field Type is the type of packet. An RREQ packet transmits from source Src to destination Dest. SrcSeqNum is the sequence number of source; DestSeqNum is the sequence number of destination; BroadcastID is the broadcast ID of the RREQ. TTL is

Time to Live value of the RREQ; last_hop is the last hop where the RREQ comes from, this is to maintain local connectivity between this node and the last hop node. RoutePath is path node list of the route path.

- RREP (Type, Src, SrcSeqNum, Dest, DestSeqNum, NextHop, last_hop, RoutePath): The field Type is the type of packet. An RREP packet transmits from destination Dest to source Src along the reverse path. SrcSeqNum is the sequence number of source; DestSeqNum is the sequence number of destination; NextHop is the next hop where the RREP should be forwarded; last_hop is the last hop where the RREP comes from, this is to maintain local connectivity between this node and the last hop node. RoutePath is the path node list of the route path.
- RERR (Type, Src, Dest, Nexthop, last_hop, Rerr_Src): The field Type is the type of packet. The route error packet includes the address of source Src and destination Dest node of the failure link. Rerr_Src is the failure node that generates the RERR packets; last_hop is the last hop where the RERR comes from, this is to maintain local connectivity between this node and the last hop node. The RERR packet is generated when a link failure to a neighbouring node is happened to notify source node the broken link.
- RERR_1 (Type, Src, Dest, Nexthop, last_hop, Rerr_Src): The field Type is the type of packet. Other fields are the same as RERR packet. The RERR_1 packet will be generated also at the link failure node but will be forwarded along the route path to the destination Dest node to notify the Dest to delete the failure link.

- HELLO (Type, Src): The field Type is the type of packet. Each node broadcasts HELLO packet in a hello interval to advertise its presence to the neighbour. Src is the node that generates the HELLO packet.
- DATA (Type, Src, Dest, last_hop, Payload): The field Type is the type of packet. It should be transmitted from Src to Dest; last_hop is for route maintenance. Payload is the payload of data packet.

2.5.2 Route discovery

In NDMR, the source node will first check its route table to look for a valid route path to the destination node if it is required to send a data packet to that destination. If there is valid route path, it forwards the data packet to the next hop in the route table to finally reach the destination, otherwise, there is no available route in the route table, and it will initiate a route discovery process.

In the route discovery process the source node generates a route request packet (RREQ), which includes packet type, source node address, destination node address, current sequence number, and the broadcast ID; it will be broadcasted to all of the source neighbouring nodes. Each time the source node creates an RREQ packet, the broadcast ID will be incremented by 1

Because of the network topology changing dynamically, achieving low overhead when establishing node-disjoint multiple routes in mobile ad hoc networks is a challenging issue. There are three key factors [14] in NDMR that avoid broadcast flood and guarantee limited broadcast overhead in mobile ad hoc networks:

- Path accumulation

- Decreasing multipath broadcast routing packets (using shortest routing hops)
- Selecting node-disjoint paths

AODV is adapted and the concept of path accumulation in RREQ packets is included in NDMR. Intermediate nodes will affix their own address to the RREQ packet after receiving the packet from the network. In NDMR, the destination node is in charge of evaluating whether the route path is a node-disjoint path or not after the RREQ packet finally reaches its destination.

If it is a node-disjoint path, the destination will create a route reply packet (RREP) including the whole route list and transmit it back along the reverse route path to the source that generated the RREQ packet. The intermediate nodes do more than just forwarding the RREP packet after receiving it. They will use the information in the RREP (such as route list and next hop) to update the route table.

There is no possibility of building node-disjoint multiple paths if every duplicate RREQ packet is dropped because some may be transmitted from another route. However, if all duplicate RREQ packets are broadcasted, a broadcast storm will be created, significantly reducing the performance of mobile ad hoc network. A new method is implemented here in NDMR to solve that problem by recording the shortest number of hops so as to reduce the routing broadcast overhead and guarantee freedom from route loops.

An intermediate node will calculate the number of hops from the source node to itself by the route list in the RREQ after the node receives an RREQ packet for the first time. This number represents the shortest number of hops in its route table. After the node receives a duplicate

RREQ packet, it first calculates the number of hops and then compares it with the number recorded in its route table. The RREQ packet will be dropped if the new number of hops is greater than the shortest number of hops in the route table. Only when the new one is less than or equal to the shortest number of hops, the node affixes its own address to the route list in the RREQ packet and re-broadcasts it to neighbouring nodes across the network again.

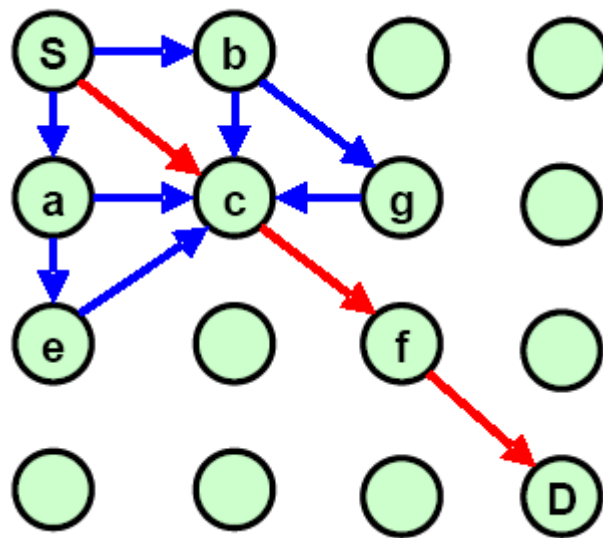


Figure 2.8 Shortest Routing Hops of Loop-free Paths³

For example, in Figure 2.8, there are five route paths between source node *S* to an intermediate node *c*: *S-c*, *S-b-c*, *S-a-c*, *S-b-g-c*, and *S-a-e-c*. The numbers of hops are 1, 2, 2, 3 and 3 respectively. Node *C* will calculate and record 1 as the shortest number of hops in its route table after it receives the first RREQ packet from path *S-c*. After the node *C* receives duplicate RREQ packets from other neighbours, it computes the new number of hops and compares it to the shortest number of hops in

³ figure 3.3 from [14]

its route table (it is **1** in this example). In this example (Figure 2.8), all other duplicate RREQ packets are dropped because the numbers of hops are all greater than **1**. It can be seen that many RREQ packets are discarded in this process under the approach. Route looping can also be avoided by this method which is proposed by Li [12] to reduce routing overhead and achieve route loop freedom in the network.

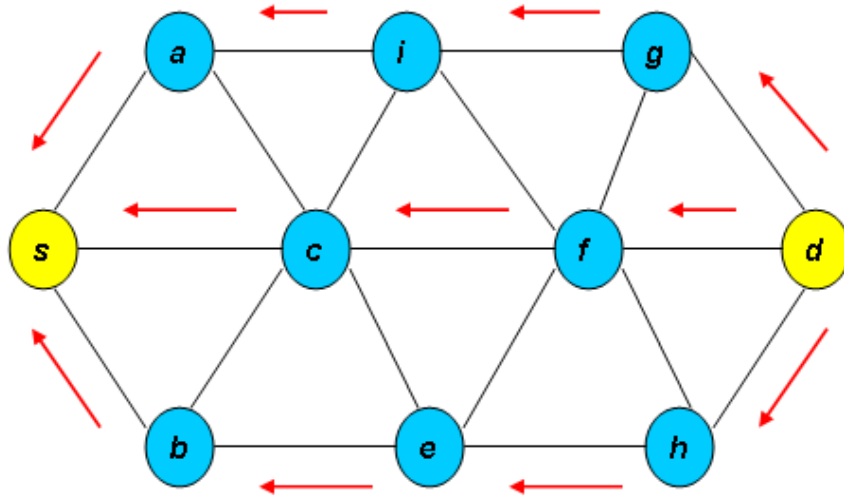


Figure 2.9 Node-Disjoint Paths

In NDMR, the destination node is in charge of choosing and recording multiple node-disjoint paths. In Figure 2.9, the destination node will record the entire route path containing each node IDs in its reverse route table after receiving the first RREQ packet from the shortest path *S-c-f-D*. And an RREP packet will be generated and transmitted down the reverse path. The destination node will evaluate the whole route path in the new RREQ packet to all the node-disjoint paths in its reverse route table if it receives a duplicate RREQ packet. The new routes (*S-a-i-g-D* and *S-b-e-h-D*) are a node-disjoint path if there is no common node (besides the source and destination node) between the new route path and every

node-disjoint path in the destination's route table, in which the new finding node-disjoint paths are recorded. Otherwise, the route path (*S-c-i-g-D*, *S-c-e-h-D*, *S-c-f-g-D*, and *S-c-f-h-D*) are not node-disjoint and hence their RREQ packets are dropped.

2.5.3 Route maintenance

Route maintenance is necessary in MANETs because of the frequently changing network topology. NDMR also relies on sending out a HELLO packet at a fixed interval to maintain local connectivity, just like AODV. "Link broken" is detected if, after a certain, pre-determined, period of time it cannot receive a HELLO packet from the neighbouring node. The detecting node will create and forward a route error packet (RERR) to the source node of the link failure route.

An intermediate node will indicate its route to the destination as being invalid if it receives an RERR packet and transmits the RERR packet to the source node along the reverse failure link. Finally, the RERR packet reaches the source node, the route to the destination is invalidated and another valid node-disjoint path will be chosen to continue transmitting packets. A new route discovery process will be initiated if there is no route valid or if only one is valid. Another route error packet (RERR_1) is also generated at the same time of RERR generation, but it is forwarded along the route path finally arriving at the destination node to notify the destination node to delete the broken link.

2.6 Other node-disjoint routing protocols

2.6.1 Ad hoc On-demand Distance Vector Multipath Routing (AODVM)

Ad hoc On-demand Distance Vector Multipath Routing (AODVM) [23] is based on AODV for supporting multiple node-disjoint routes between a

source and destination node. AODVM does not discard the duplicate RREQ packets. After packet, the intermediate node must pick up the information in the RREQ packet, such as the source, destination and neighbour node of that RREQ packet, and other additional link information.

The destination node will update the sequence number and generate an RREP packet after the first RREQ arrives. There is a “last-hop-ID” field in the RREP packet to trace the last hop of the RREQ packet. If the destination node receives duplicate RREQ packets from different neighbours, it will update the sequence number and generate RREP packets respectively. The RREP packets will be transmitted back along the reverse path by the RREQ to the source node.

In AODVM, intermediate nodes can choose where to transmit the RREP packets. An intermediate node will erase the routing entry to its neighbouring node from the RREQ table after it receives an RREP packet from a neighbour, and a new routing entry will be added showing the new path to the destination. The node looks for the next hop in the RREQ table which has the smallest number of hops along the path to the source node, and then transmits the RREP packet to that node.

In order to archive multiple node-disjoint paths, which means a node does not participate in multiple paths, when nodes overhear any node broadcasting an RREP message; they delete the entry corresponding to the transmitting node from their RREQ tables.

Because the destination node does not know whether the RREP packet is reaches the source node or not, the source node must acknowledge every RREP packet it received. The acknowledgement can be piggybacked with the first data packet sent on the route path and could also include

information such as the first hop, last hop and hop count of the route.

2.6.2 Multiple Attempt Multipath Routing (MAMR)

Some node-disjoint multipath routing protocols cannot guarantee to find all the node-disjoint paths between a source and destination pair in a single route discovery process, although some of them can find all the paths in several attempts at the route discovery process. MAMR [81] is an improvement in multipath routing protocol to guarantee discovering the maximal set of node-disjoint paths between a source and destination pair with a lower number of route discoveries.

To reduce the number of route discoveries, the protocol will discover paths as many as possible in the first route discovery. To achieve that, a forwarding scheme known as Only the First Copy (OFC) [80] is used. In OFC, an intermediate node only forwards the first copy of an RREQ and discards other copies of the RREQ from the same source node. The destination computes a maximal set of node-disjoint paths and sends multiple RREPs, one along each path to the source node. Each subsequent route discovery process discovers only one path if there is path between the given pair of nodes. The set of node-disjoint paths after the last route discovery produces the final set of node-disjoint paths - so the discovery of the maximal set of node-disjoint paths is achieved.

AODVM does not discard the duplicate RREQ packets, which will leads to a high routing overhead across the network. Although MAMR can find the maximal set of node-disjoint paths with lower routing overhead (lower number of route discoveries), it is not guaranteed that every path is with the lowest number of hops as it only forwards the first copy of an RREQ. NDMR solves those problems as discussed in section 2.5.

2.7 QoS in Multipath routing for MANETs

The notion of QoS, is a guarantee by the network to satisfy some predetermined service performance constraints for the user in terms of the end-to-end delay, available bandwidth, probability of packet loss, and so on [10]. Future ad hoc mobile networks will carry diverse multimedia applications such as voice, video and data. In order to provide quality delivery to delay sensitive applications such as voice and video, it is necessary to provide quality of service (QoS) in MANETs.

2.7.1 Differentiated services (DiffServ)

Differentiated Services (DiffServ) [25] is a standard approach to achieve QoS in any IP network and could potentially be used to provide QoS in MANETs. QoS is achieved in DiffServ by different traffic classification which is the foundation of distributing network throughput. The classification is done in a 6 bit DiffServ Code point (DSCP) field [26] in the packet. The DSCP field is part of the original type of service (ToS) field in the IP header. The IETF redefined the meaning of the ToS field, splitting it into the 6-bit DSCP field and a 2- bit unused field. The unused field is being allocated to the Explicit Congestion Notification (ECN) mechanisms [26], as shown in Figure 2.10.

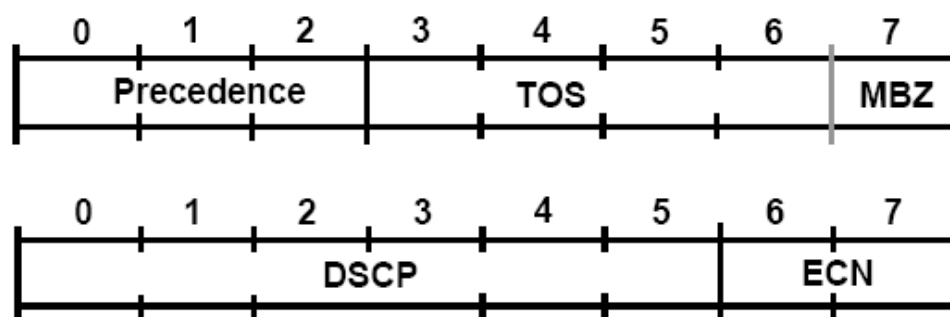


Figure 2.10 DSCP and ECN

Meeting the different performance requirements of different users is the main idea of Differentiated Services. It classifies traffic into different priority levels and achieves QoS support by applying priority scheduling and queuing management procedures.

Per-hop behaviour (PHB) is the QoS behaviour of a packet at a specific node in the network determined by DSCP and is expressed through the packet scheduling and dropping method during a communication. In general, the PHB is the packet queue at a node, the drop probability when the queue exceeds a certain threshold, the buffers and bandwidth allocated to each queue, and the service rate of each queue. DiffServ provides differential traffic forwarding action, therefore, providing QoS for different traffic types.

DiffServ is a fully distributed and stateless model. Each node does not need to keep state information. Moving the complexity to the edge nodes in the network is the main intention of this method so that the intermediate nodes can forward packets as simply and fast as possible.

2.7.2 Multipath QoS routing for supporting DiffServ (MQRD)

NDMR is a best-effort routing protocol and does not achieve QoS though it provides efficient node-disjoint multipath routing in MANETs. MQRD [13] is an appropriate routing protocol in MANETs to provide QoS support by taking the advantages of both NDMR and DiffServ: efficient small routing overhead node-disjoint multipath routing and standardized QoS approach.

End-to-end QoS support is achieved in MQRD by adjusting priority levels of different traffic flows. There are two important issues of service differentiation: scheduling and queuing management. Scheduling is done by a priority scheduler that includes two queues: a high-priority queue

and a low-priority queue. The packets in the high-priority queue must be served before packets in the low-priority queues. Queuing management will drop some packets from the buffer in cases of congestion.

Take class *A* and *B* as an example. *A* denotes voice and video traffic, which generally require low delay, low loss, so it is an Expedited Forward traffic. *B* denotes best effort traffic such as HTTP or FTP, which usually have no specific requirement on delay and loss. The priority scheduler should transmit class *A* packet ahead of class *B* packet. Figure 2.11 shows the priority scheduler.

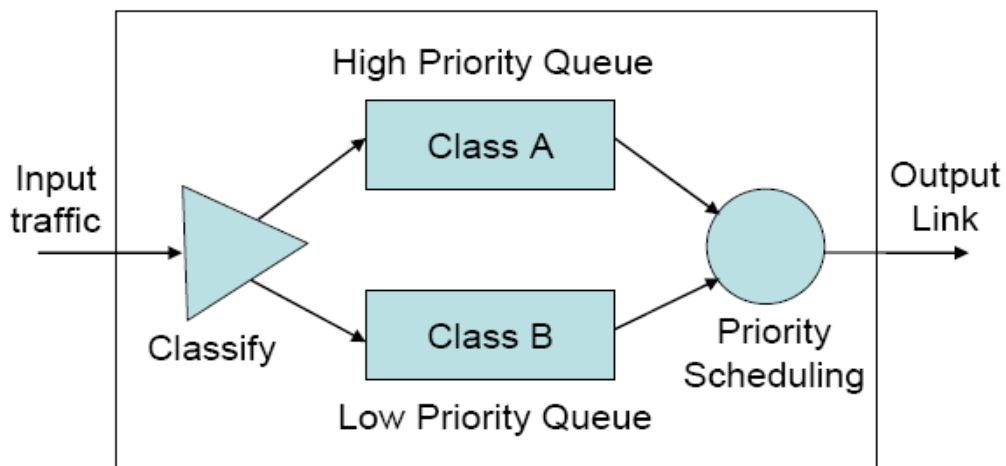


Figure 2.11 Priority Scheduler ⁴

Only when there is enough empty space in the queue for taking the incoming packets, the priority scheduling is useful. If the queue is full, the high priority packets are still lost which is not what we expected. Making sure the queue always has room for high priority packets is the main goal of queue management. For solving this problem, random early detection (RED) [7] is used in MQRD. There is a minimum threshold that is the number of packets in a queue before any dropping. The dropping

⁴ Figure 4.2 from[14]

probability increases from the minimum threshold of the queue length until it reaches the maximum threshold. After that, all new arriving packets are discarded.

The figures below (Figure 2.12) show the packet delivery ratio and average delay after implementing QoS in NDMR [13]. It is clearly shown that after QoS support, the network achieves better QoS performance in order to meet different SLAs for packets of different classes. Because of the implementation of a priority scheduler in MQRD, EF traffic achieves a better delivery ratio than BE traffic. There are more packets lost in NDMR than in MQRD as the number EF traffic sources increases; this is because there is more congestion when the number of EF sources increases. EF packets of MQRD have a much lower average delay than BE packets because the priority scheduler in MQRD lets EF packets be forwarded more quickly. With the increase in the number of EF sources, the average delay-time of BE packets in MQRD increases more quickly than that of EF packets.

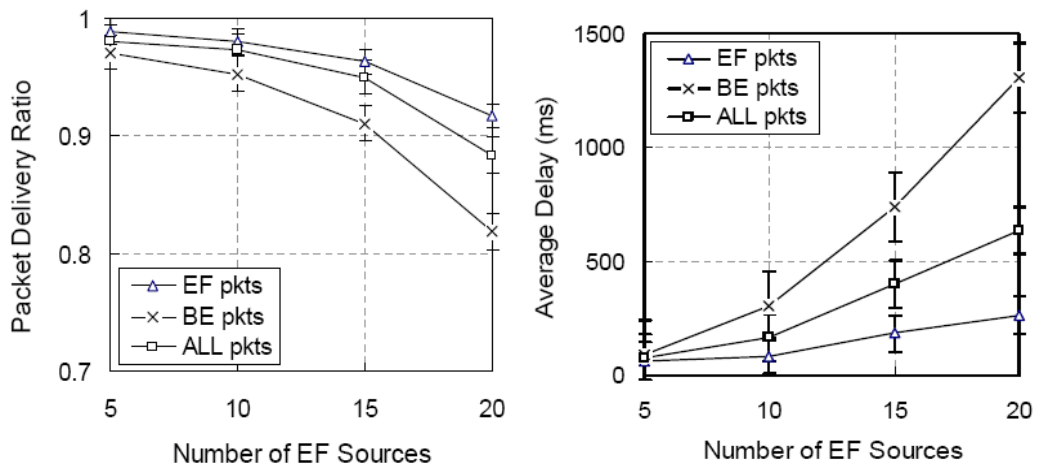


Figure 2.12 Packet Delivery Ratio and Average Delay of MQRD ⁵

⁵ Figure 4.4 and 4.6 from [14]

As this approach in [14] is on the MAC layer rather than the network layer (the focus of this thesis) here is only a brief introduction of Diffserv.

2.7.3 QoS in Vehicular networks using multipath routing

Vehicular Ad hoc NETWORKS (VANETs) consist of spontaneous groups of vehicles that dynamically change their position and exchange data between each other through wireless links. These networks cover many applications such as safety on the road, warning drivers about accidents, congestion ahead on the road or sending information to allow a server to centralize information from all vehicles about such matters as mechanical state, position, street status and emergency situations.

The main feature of the routing protocols used in VANETs is the ability to keep efficiently and reliably the communication between a source and destination node, even if the position and speed that they follow change very quickly. Thus, when the communicating nodes are not directly connected, the information exchange is achieved by relaying packets through intermediate vehicles.

A simulation study of the behavior and performance of the multipath routing protocol (modified AODVM) for a vehicular network supporting data transmissions directly between mobiles (Vehicle to-Vehicle) and between the fixed network segment and the VANET network, via the gateway nodes, were done in [82]. The simulations pay special attention to the impact of the multipath routes on the fundamental QoS metrics (delay, packet delivery fraction).

As Figure 2.13 shows, as the network size increases, the power of the multipath routing protocol begins to show itself. The node density allows the protocol to more easily find multiple network paths between a source and destination pair. The use of multipath in the gateway is not a key

factor, because the multipath capability in the mobile suffices to exploit the multipath routes. In some cases, the delay is even lower than with multipath in the gateways.

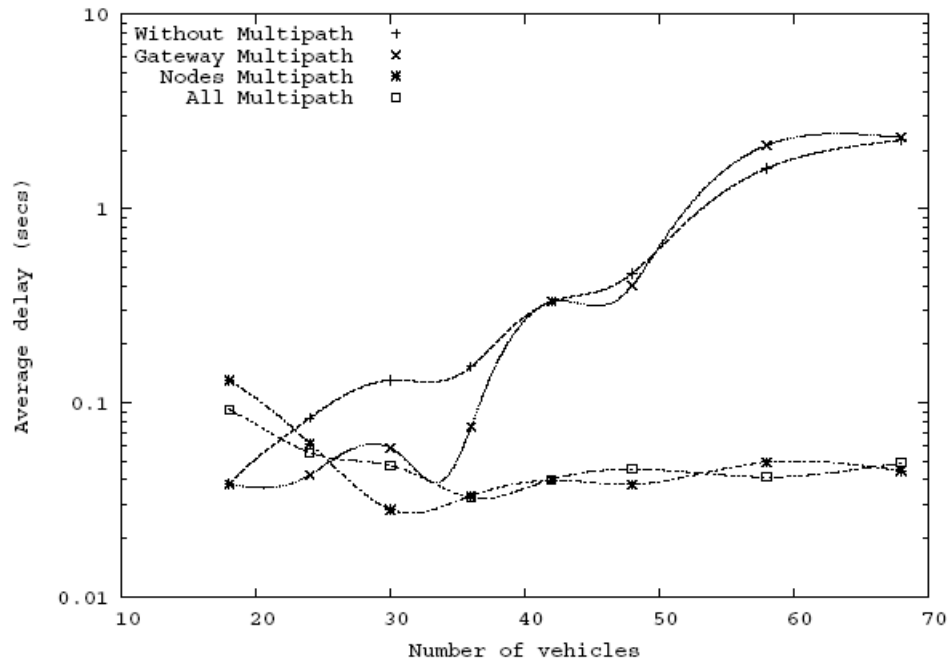


Figure 2.13 End to end delay (VANETs) ⁶

In networks with more than 50 mobiles, the use of multipath routing offers much better packet delivery ratio, shown in Figure 2.14. As with the delay, in this case the scheme with only multipath nodes gives better results than other cases.

⁶ Figure 3 from [82]

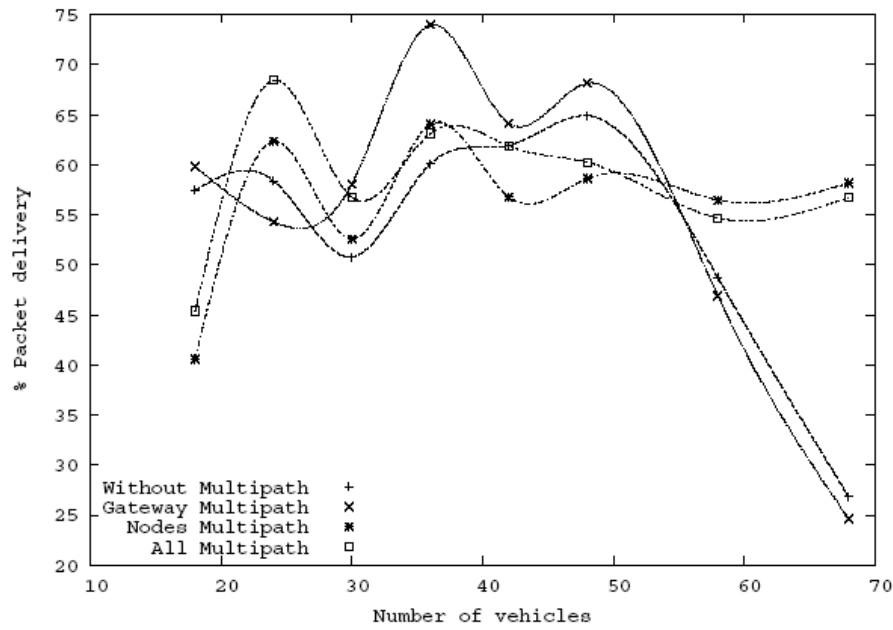


Figure 2.14 Packet delivery ration (VANETs)⁷

As this routing protocol uses gateway to relay packets in the network during communication, this is different from the typical case in ad hoc networks. The research done in this thesis is in the situation where all communications are done by pure node multiple paths.

2.8 Problems with current QoS supporting multipath routing protocols

Several multipath routing protocols and QoS supporting have been presented in the previous sections. Although these protocols can acquire better network performance by providing QoS support, they will always keep transmitting packets along the same path as long as the path is not broken no matter other multiple disjoint routing paths have better link quality.

⁷ Figure 4 from [82]

Transmission delay of each path will be varied during the process of forwarding packets. MQRD providing QoS support to NDMR transmits data on the first available multiple disjoint path. If other path has lower delay time, transmission on that path will achieve better performance. How to adjust packets be transmitted on the lower delay link has to be considered when designing QoS support routing protocols.

Table 2.1 Comparison of Multipath routing protocols compares the multipath routing protocols for MANETs.

Table 2.1 Comparison of Multipath routing protocols

	AOMDV	AODVM	MSR	NDMR	MQRD
Loop Free	Yes	Yes	Yes	Yes	Yes
Routing Overhead Control	No	No	No	Yes	Yes
Node-disjoint Paths	No	Yes	Yes	Yes	Yes
Complete Routes known at Source	No	No	Yes	Yes	Yes
QoS Support	No	No	No	No	Yes
Periodic Update	No	No	No	No	No
Security Support	No	No	No	No	No
Muti-rate Forwarding	No	No	No	No	No

3 QoS-ENABLED NDMR

3.1 Introduction

The NDMR routing protocol can efficiently discover multiple-route paths between source and destination nodes in a procedure that has a low overhead and that leads to minimal routing latency. However, it is not very efficient if NDMR needs to change to another path when the delay of the current path cannot meet the SLA requirements.

In NDMR, after deciding a path is a node-disjoint path, the destination will create a route reply packet (RREP) that contains the node list of the whole route path and unicasts it back to the source. However, an RREP only currently contains the route path; it does not include anything about the state of that path, such as delays or available bandwidth. It, therefore, does not support any information that will allow the QoS of the MANET to be maintained.

This thesis describes a novel extension to NDMR first proposed by the author in [Liu-2] that allows the source to be able to choose the “best” route, not necessarily the shortest, to the destination. The definition of best can vary according to the specific requirements of an implementation; for example, queue length, blocking probability, available bandwidth could all be use, but the initial work in this thesis used queue length as a measure of delay time⁸.

⁸ It is recognised that other factors, such as service rate at a node (which may not be the same in all nodes), affect overall delay but the initial work assumed uniform service times and uniform transmission rates between nodes so that queue length is a reasonable approximation.

To implement this approach, the RREP packets have a queue length field added and when each intermediate node receives an RREP packet, it adds its queue length to the queue length field in the packet. Thus, when the source node receives the RREP from the destination node it knows the exact queue length along the path.

It is recognised, of course, that the queue lengths will change and may even have changed by the time the RREP reaches the source, but it does provide an indication of the delays along each route. Later in this chapter, simulation results show the performance gains that can be achieved using the approach.

An additional type of packet, a Route Update Packet (RUP), is added to carry up-to-date values of queue length, thus letting the source node know which path has the lowest transmission delay in order to achieve better transmission performance.

3.1.1 RREP with queue length Field

In the implementation of this routing protocol, each source keeps three node-disjoint paths for a particular destination. The reasons for keeping three node-disjoint paths are: i) two paths are the minimum as two types of priority traffic need at least two paths for transmitting packets and ii) one more path is needed for a backup route if other paths fail. Therefore, three paths gives the network with the “minimum protection”.

If more paths exist, there will be more routing overhead and more packet loss because of congestion.

Simulations have been done for recording the number of route breaks in the three-path system; the results are shown in Table 3.1. During the 600s simulation 120,000 packets were generated (20 sources generating at

10 packets/s); in that time the largest number of route breaks was 377 (at the maximum velocity of 20 m/s) which is 0.3% of the total number of packets. Therefore, keeping 3 paths is reasonable in the simulation to get reliable results.

Table 3.1 Number of route breaks

Max velocity (m/s)	0	5	10	20
Number of route breaks	0	79	203	377

With the queue length field in the RREP packet, the RREP will show the sum of the individual queue lengths in each node along the path to give the total queue length of the path. When the RREP arrives at the source node, the source node will know the queue length of that path at the same time as it knows the path route. If the source receives another RREP packet, it will choose the path with the minimum queue length, irrespective of route which has the smaller number of hops. This allows it to minimise the delay time thus providing better QoS.

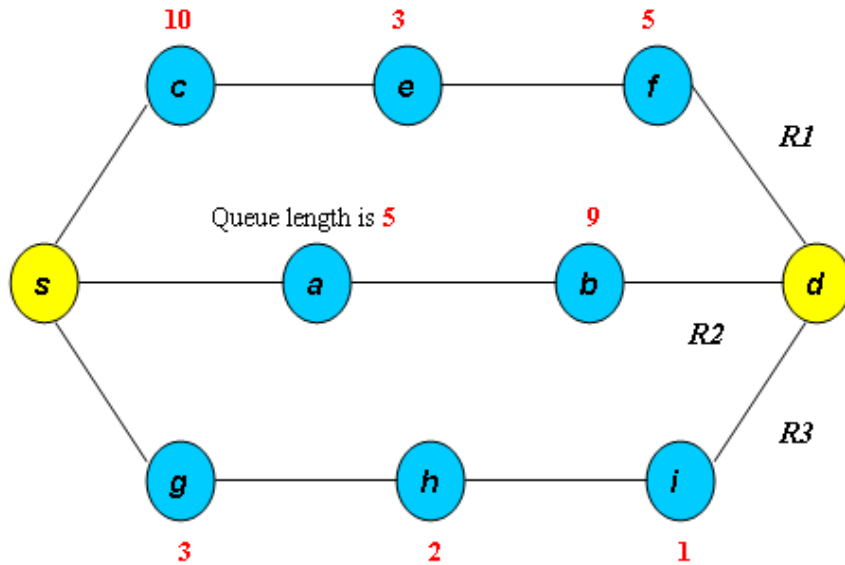


Figure 3.1 Queue length in multiple node-disjoint paths

Figure 3.1 shows an example of queue length along the multiple paths. Assume source node *S* first receives the RREP from route 2 (*R2*) (*s-a-b-d*). In standard NDMR, *S* will always transmit data on that route as long as no link break happens, even though route 3 (*R3*) (*s-g-h-i-d*) has a smaller queue length and hence a lower delay. With the introduction of the queue length field in RREP, *S* will initially choose route 2 (*R2*) (*s-a-b-d*) to transmit data as it receives an RREP from that route first. After receiving the RREP from route 3 (*R3*) (*s-g-h-i-d*), the source node will compare the queue length of the existing routes, then change to route 3 (*R3*) (*s-g-h-i-d*) to continue transmitting data. Using this approach can reduce the transmission delay and offer better QoS.

Figure 3.2 is the comparison flow chart at the source node to explain the differences in operation between QoS enabled NDMR and standard NDMR. With QoS support, the source node is able to choose the path with the lowest queue length to transmit data; as explained later, this choice can be made dynamically so that data is always transmitted along the quickest path, even though conditions on the MANET may change.

In standard NDMR, after receiving the RREP from the shortest route path – here denoted as route path 1 (RREP1), the source node will always transmit packets along path 1 as long as path 1 is available, no matter whether the delay time is long or short. Paths 2 and 3 (if available) are only the back up paths in the case of a break. On the other hand, in QoS enabled NDMR, all three paths are available for transmitting data as long as they are not broken and a source node can choose the path with the shortest queue length (as returned in the RREP packets).

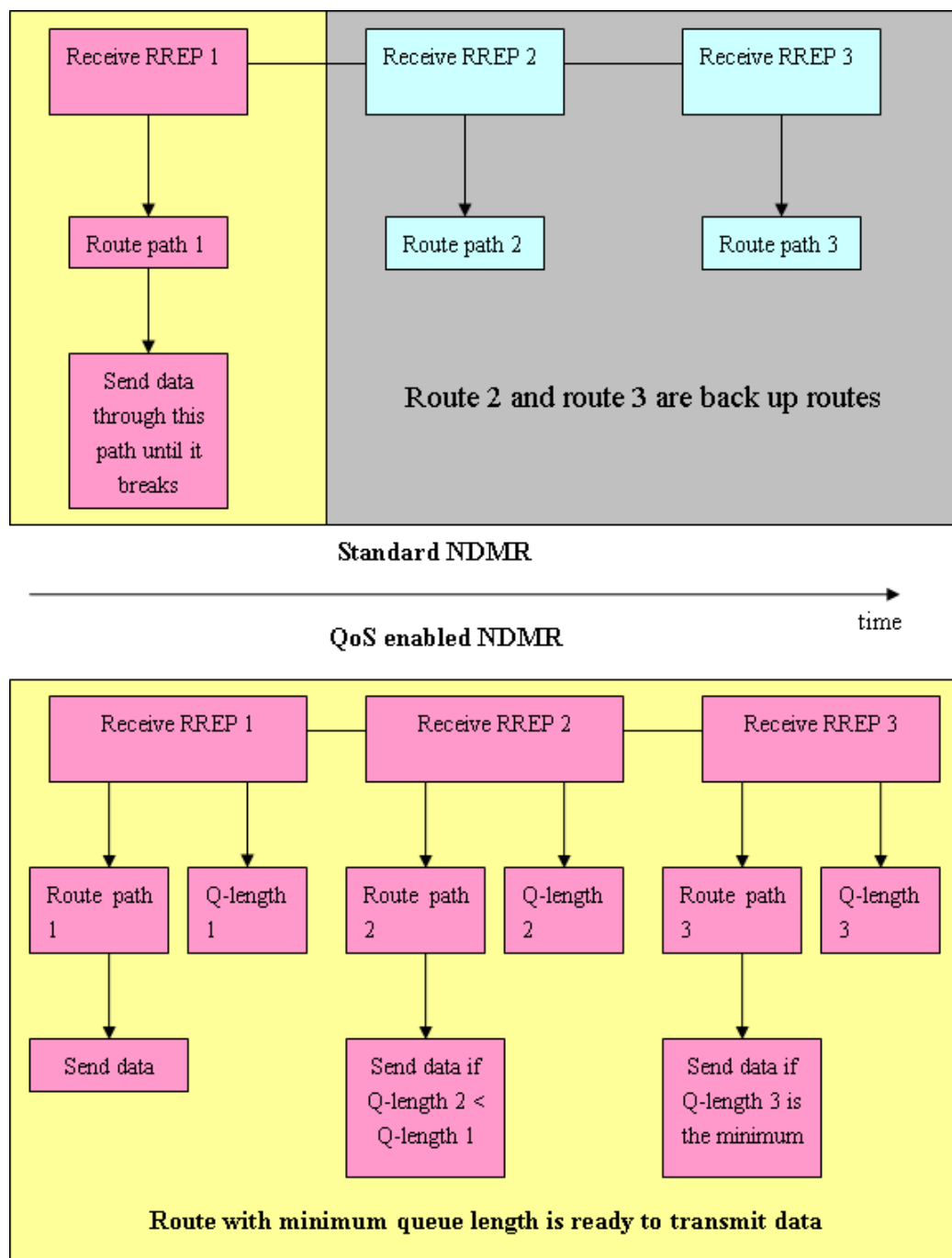


Figure 3.2 Simple flow chart comparing NDMR and QoS enabled NDMR

3.1.2 RUP packet

As an RREP is generated only in the route discovery process, the protocol cannot refresh the queue length of each path in real time – and obviously queue lengths will change. As part of the enhancement to NDMR, the

need for a similar packet, RUP, route update packet, containing the same queue length field used in an RREP packet has been identified. This allows more frequent updates by getting the destination node to periodically unicast an RUP packet that will collect up-to-date queue length information on its way to the source node. The source will be able to again choose the best path according to the new values of queue length.

For standard NDMR, the source node will transmit data along the first valid path until it became invalid (link breaks), but with the RUP packet introduced in the protocol the source node will be able to frequently change to the best path in order to get better transmission delay, shown in Figure 3.3.

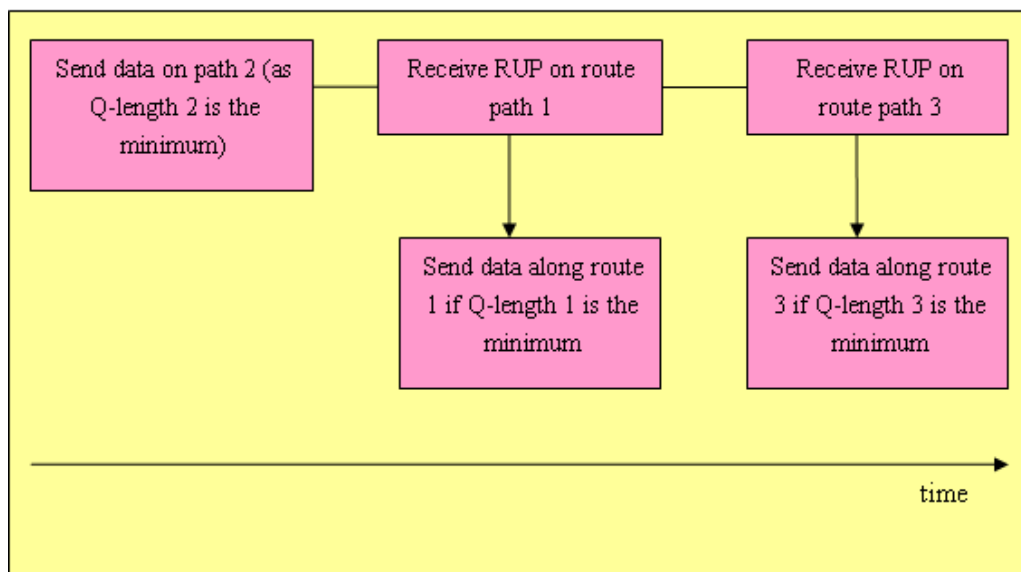


Figure 3.3 simple flow chart of RUP

3.2 QoS enabled NDMR OPNET Simulation Model

In order to allow a structure that could easily incorporate QoS control, the simulation used in this thesis was written by the author from scratch – the simulator used by Li [14] was not used. Although this makes it less easy to compare results it does allow a simulation architecture tailored to adding the extra functionality to be produced.

The commercial discrete event simulator OPNET 10.5 was used to produce the simulator. In this section detailed descriptions are given for QoS enabled NDMR network model and other major process models.

3.2.1 Network Model

The network model is shown in Figure 3.5. Because there is no structure in ad hoc networks and all the nodes have the same functionality, the whole network consists of only one kind of node. The maximum path length (number of hops) across the network is 10 and the average number of hops is shown in Figure 3.4 below. As is shown in Figure 3.5, there are 50 mobile nodes that can move freely in a 1000m * 1000m square area; the wireless node transmission range is 250m. This is the same scenario used in [14].

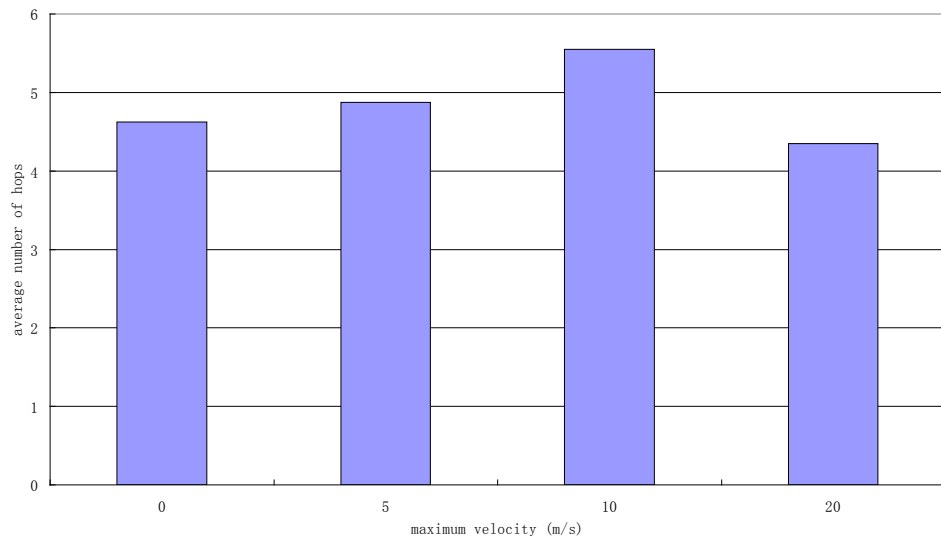


Figure 3.4 Average number of hops across the network



Figure 3.5 Network Model

3.2.2 Node Model

As shown in Figure 3.6, the QoS enabled NDMR node model is built on the OSI layer stack, the focus being the QoS enabled NDMR routing implementation on the network layer. Each node in the network is uniquely identified by its IP address.

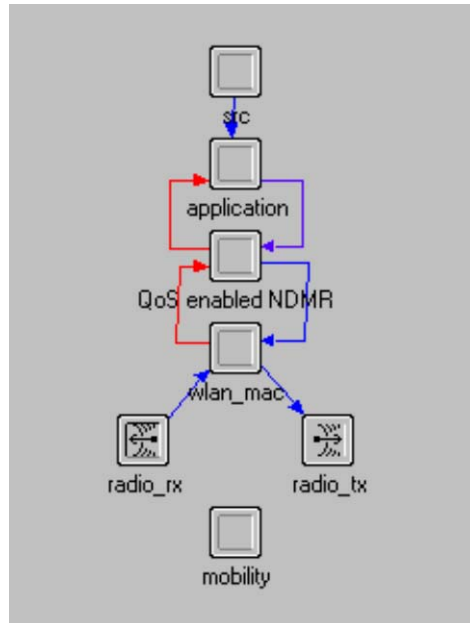


Figure 3.6 Node Model

Each module in the node model is described below.

- **src** module: This module generates packets with a specified packet size and inter-arrival time, which are then forwarded to the lower layer (**application** module) immediately after generated.
- **application** module: This module allocates a random destination address from **src** module to arriving packets.
- **QoS enabled NDMR** module: This is the main module in the simulation, being used to discover and maintain the routing information of a mobile ad hoc network. The module firstly checks

its route table after receiving a data packet from the application layer. If there is a route to the destination in the route table, the module transmits the data packet via the next hop in its route table. Otherwise, the module will execute the QoS enabled NDMR algorithm to discover multiple routes to the destination node.

- *wlan_mac* module: This module is an implementation of IEEE 802.11 [27] standard medium access control (MAC) protocol.
- *radio_tx* module: This module receives packets from the *wlan_mac* module, does required calculations like broadcast frequency, estimated transmission delay and then sends packets on the radio channel.
- *radio_rx* module: This module receives packets from the radio channel, does some calculation and checks the received power. If the received power is over the threshold, it then forwards packets to the *wlan_mac* module, otherwise the packets are dropped.
- There is no explicit antenna module, which will cause the simulation to act as if an isotropic antenna was present.
- *mobility* module: All nodes in the network move freely within the area. This module controls the movement of the current node in terms of speed, direction and distance.

3.2.3 Process Model

The functionalities of each module are achieved in the process model. A process model is represented by finite state machine (FSM), and it is created with icons that represent states and lines that represent transitions between states. Operations performed in each state are described in embedded C or C++ code blocks.

3.2.3.1 Application Process Model

The main function of the application process model (shown in Figure 3.7) is to allocate a random destination address for each packet passed down from the *src* module.

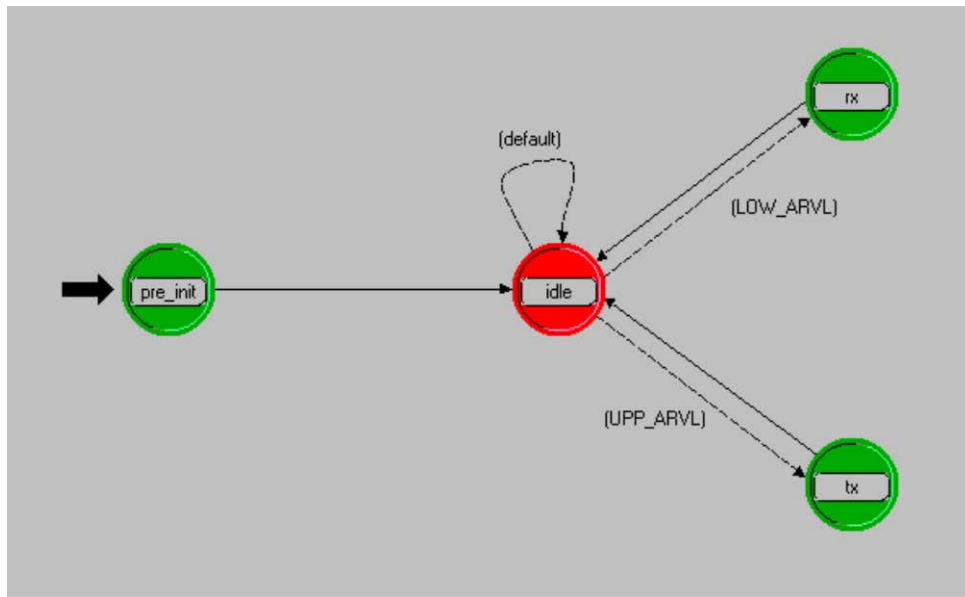


Figure 3.7 Application Process Model

The Model Attribute is “Destination”. When the “Destination” is set to NONE, the node cannot send data packets to other nodes. If the “Destination” is RANDOM, the node will choose a random destination ID as the destination node for communication. If the “Destination” is a certain node address, the node can only send data packets to that node.

Firstly, each node at the *pre-init* state gets the object attributes and then checks its “Destination” attribute. If the value of “Destination” is set to NONE, no action should be undertaken and the state transits to *idle*.

When the current node is in the *idle* state it can transit either to the *rx* (packet passed from the route layer) state or *tx* (packet passed from the source layer) state.

When a data packet is received from the upper layer *src* module, it transits to the *tx* state. The current node checks its “Destination” attribute and if the value is RANDOM, the current node picks a random node address as the destination address. If the value is not NONE, that is, it has been given a destination address from RANDOM or has been assigned to a specific node address, the current node sends the data packet from the *src* module to the destination IP address to the *QoS enabled NDMR* module.

In the *rx* state, the current node has just been passed a packet from the lower layer, *QoS enabled NDMR* module. As the packet has reached its destination, the current node simply destroys it and collects some statistics for later processing.

3.2.3.2 QoS enabled NDMR Process Model

QoS enabled NDMR routing algorithm is implemented in QoS enabled node-disjoint multipath routing process (Figure 3.8). It has been explained in the previous section.

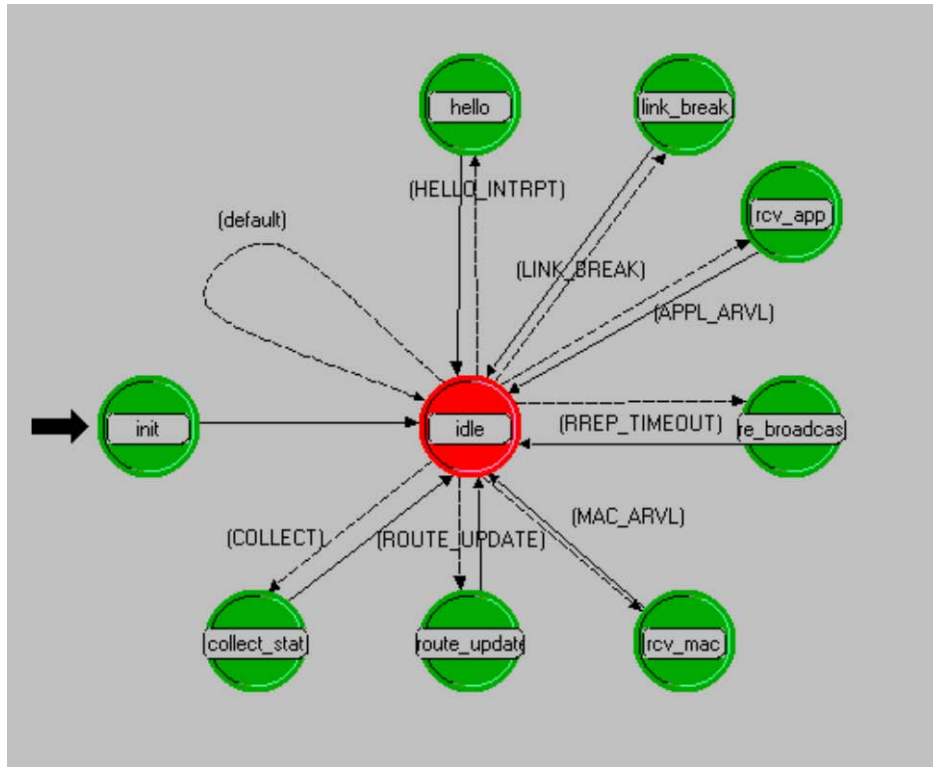


Figure 3.8 QoS enabled NDMR routing process model

The states in this model are as follows:

init state: This state does the initialization of the process. Routing information tables are initialized (route table, reverse route table, sequence number). The first HELLO self interrupt is scheduled in this state. Once the initialization is completed, the process transits to *idle* state.

rcv_app state: When this state receives a packet passed down from the upper layer, the process transmits a data packet to a given destination. The state gets the destination address from the received packet and then checks the route table. If there is a route in the route table, the state will insert the next hop address obtained from the route table and forward it to the next hop node. If there is no route in the table, the state will save the packet in a waiting queue and initialize a route discovery process.

rcv_mac state: This state receives the packets passed up from the lower layer. It first checks the data type of the incoming packet, and then calls the corresponding function for that data type to proceed. If the data packet has reached the final destination, the state will pass it to the upper (application) module. This is the most complicated state in the process for there are different types of packets and different corresponding functions to deal with them.

re_broadcast state: When an RREP_TIME_OUT timer expires (scheduled when initializing the route discovery process) for a given destination, the process transits to this state. The current state checks whether a re-broadcast is needed or not. The discovery process will be terminated and any data packets waiting for this route are dropped if the number of retries achieves the maximum value. If it has not been reached, an RREQ packet will be rebroadcast.

hello state: Each node broadcasts a HELLO message in every HELLO INTERVAL time to advertise its presence to the neighbourhood.

link_break state: When a node fails to receive HELLO from a neighbouring node after a given time, it is considered there is link break between these two nodes. An RERR packet will be sent along the reverse path to the source node.

collect_stat state: The process periodically transits to this state to collect global statistics. These statistics are added to a text file created at the beginning of each simulation run.

route_update state: The destination node periodically unicasts RUP packets containing up-to-date queue length data to the source node. The source will be able to dynamically choose the best path according to the new queue length information.

3.2.3.3 Medium Access Control Process Model

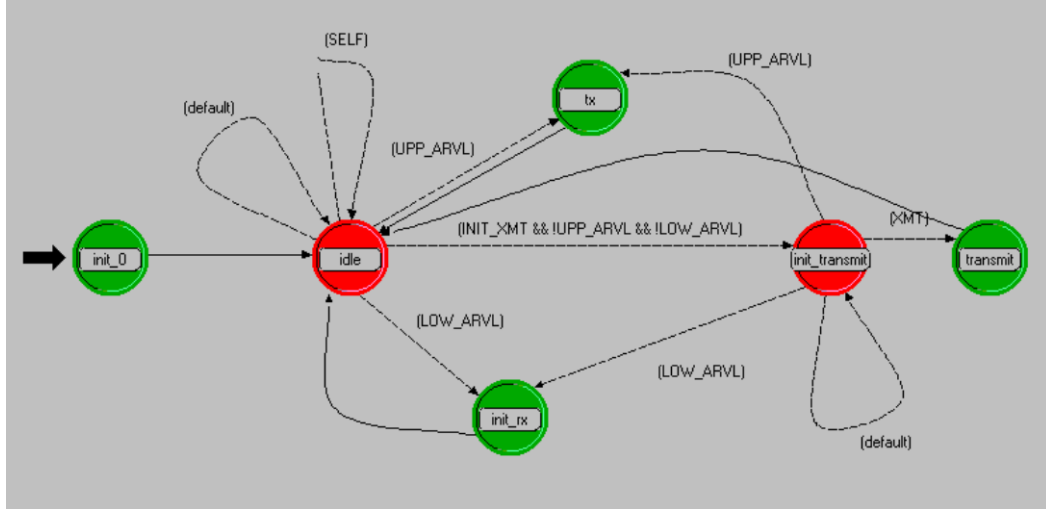


Figure 3.9 MAC Process Model

The IEEE 802.11b MAC protocol with Distributed Coordination (DCF) is implemented as the MAC layer in the simulation. DCF is a standard access scheme used by mobiles to avoid hidden and exposed terminator problems and distribute the wireless channel [28].

3.2.3.4 Mobility Process Model

A random waypoint mobility scheme [29] is implemented in mobility process model (Figure 3.10). In order to compare the results with standard NDRM, this is the same mobility model used in [14].

The node movement process is implemented by a sequence of discrete small steps. A moving node updates its position every step time which is set to 0.2s in the simulation here.

The attribute of the model is “Mobility Parameters”, a compound parameter containing:

- “MOBILITY”: indicates whether the node is fixed or mobile.
- “PAUSE_TIME”: the waiting time after the node reaches the target

(30 sec).

- "MVT_STEP": 0.2s, as noted previously.
- "SPEED_LIMIT": the maximum speed that a node can reach.
- "XMIN","XMAX": the min and max distance along the x axis that a node can move.
- "YMIN","YMAX": the min and max distance along the y axis that a node can move.

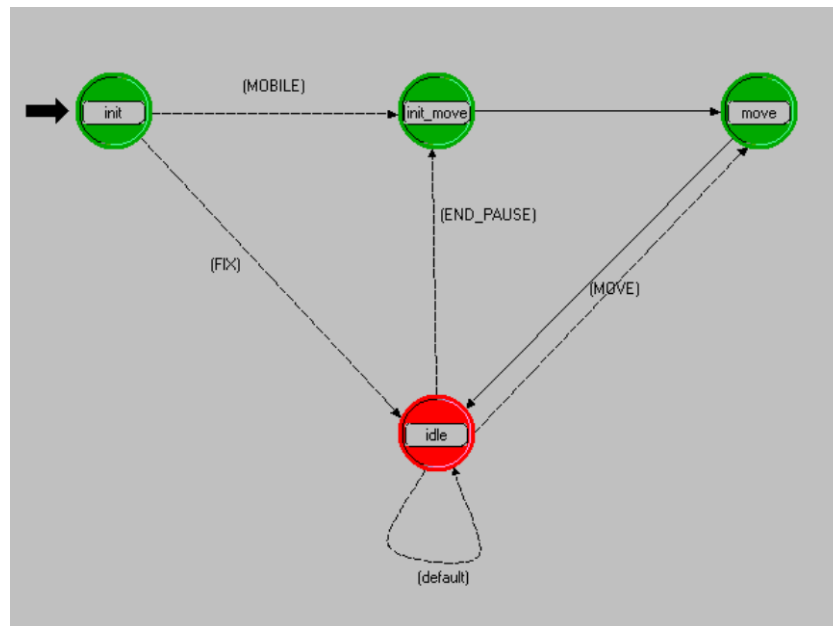


Figure 3.10 Mobility Process Model

In the *init* state, every node chooses a random position within the moving area. In the initial position, each node decides whether it can move by checking its "MOBILITY" attribute. If the attribute is set to **1**, the node goes into the *init_move* state to initialize some movement parameters. If it is set to **0**, which means the current node cannot move and should keep static during the simulation, it will go into the *idle* state in this occasion.

In *init_move* state, a moving node chooses a random destination position within the area range and a random speed between 0 and the "SPEED_LIMIT" value. Then the moving node will transit between the

idle state and the *move* state at regular intervals (every step time) until it reaches the destination point. Each step of a moving node travels by a distance given by

$$\delta = ("MVT_STEP" \times \text{speed})$$

in the direction of the destination position.

The node will check if it has reached the destination or not after each step movement. It will transit to the *idle* state and wait for a pause timeout if it has reached the destination. After timeout, the node will transit to the *init_move* state to prepare a next trip. Otherwise, if it has not reached the destination, the current node pauses for a step time and then goes back to the *move* state again until it reaches the destination position.

3.2.4 Mobility and Traffic model

Random waypoint model [29] is applied to simulate nodes movement. Each node moves from a random start position to a random destination at a certain speed. After reaching the destination, another random destination is selected after a pause time (here 30s).

3.2.5 Performance Metrics

A simulation runs for 600 simulation seconds. Papers [12], [64] and [65] uses 500 to 800 simulation seconds; 600 seconds is chosen here a reasonable time for achieving rational results and is in line with the literature. A guard period of a further 10 seconds is allowed at the end of the simulation to collect packets in transit. One result point is the average value of five runs with different random seeds. The maximum and minimum values (error bars) are shown in the figures.

The main performance metrics used are:

- *Average delay time of data packets*: The average delay time from the

moment the packet is generated at the source node to the moment it reached the destination node. In order to investigate the network performance and have a result covering the whole network, it is the average delay time over all the generated packets that is measured, not just that over a single source-destination pair.

- *Packet delivery ratio*: The ratio of the data packets reached at the destination nodes to those generated by the source nodes. Again, it is the average packet delivery ratio over all the generated packets.

3.3 Simulation validation

Some validation steps are required to verify and validate the simulation to prove that implementation of QoS enabled NDMR and the simulation model execute correctly. OPNET debugger (ODB) is used to trace every step in simulation process. As the MAC layer is different for every simulation implementation model, it is difficult to calculate the delay time on the MAC layer so that using a mathematical model for validation is not suitable in this situation.

3.3.1 Step-by-step validation

This approach traces the operation of the simulation in detail and allows simulation results and intermediate results to be seen and also provides tools such as traces and breakpoints. A set of validation checks were defined (listed below) and the results are “True” for all cases so there is confidence the simulator is working properly (part of validation rules are similar to Li’s [14] as they are the validation of NDMR):

1. If a node wants to build communication with a destination node but there is no existing route, the node will generate an RREQ packet and broadcast it to all its neighbours.

2. An intermediate node is required to affix its own address to the route list and increase the value of TTL, and then rebroadcasts the RREQ packet to its neighbours.
3. The Time to Live (TTL) value from any received RREQ packet must be less than the maximum allowed TTL value; otherwise, the received RREQ packet is dropped.
4. If the broadcast ID of a received RREQ is greater than the Request ID in its source_broadcastID table, the new Request ID must replace the original the broadcast ID value; otherwise, the RREQ packet must be dropped.
5. If the hopcount of the route path of a received RREQ is less than or equal to the reverse shortest routing hopcount recorded in the routing table, the intermediate node must affix its address to the RREQ and forwards the RREQ to its neighbours. Otherwise, the RREQ packet must be dropped.
6. If an RREQ packet arrives at the destination node and the broadcast ID is greater than the Request ID in the routing table, the destination node will generate an RREP and send it back to the source node along the reverse routing path. If these two values are equal, the destination node will decide the current routing path is a node-disjoint path or not.
7. If the routing path of a received RREQ is a node-disjoint path, the destination node will generate an RREP packet and transmit it back to the source node along the reverse routing path; otherwise, the RREQ packet must be dropped.

8. After an intermediate node receives a RREP, it records the next hop to the destination node to the route table and the next hop to the source node to the reverse route table before it forwards the RREP packet.
9. When an RREP arrives at the source node and the destination sequence number in the RREP is greater than or equal to the destination sequence number in route table, the source node will record the next hop in its route table. Otherwise, the source node drops the RREP packet.
10. When a source node receives an RREP, it updates the queue_length field with the lowest queue length recorded in the queue length field in the RREP, and transmits data along the path with the lowest queue length.
11. When an intermediate node receives a data packet, it picks up the destination address from the data packet and checks its route table to get the address of the next hop and forwards the data packet to the next hop node.
12. When a source node receives an RUP, it updates the queue_length field with the lowest queue length recorded in the queue length field in the RREP, and transmits data along the path with the lowest queue length.
13. When there is only one route available in the route table of a source node, the source will start a route discovery process to build new multiple node-disjoint paths.
14. Every node sends HELLO packets at a fix interval time to keep local connectivity. Failing to receive a HELLO packet from a neighbour after a certain period, the node sends an RERR packet back to the

relative source node and an RERR_1 packet to the corresponding destination node.

15. If an intermediate node receives an RERR_1 packet, it forwards it to its next hop along the route path to the destination. After receiving an RERR_1 packet at the destination, the destination node will delete the route path to the source node in the routing table.
16. If an intermediate node receives an RERR packet, it marks its route to the destination node invalid and then forwards the RERR to its next hop along the reverse route path to the source node. After receiving an RERR, the source invalidates the route path to the destination and chooses a valid node-disjoint route from the route table to continue transmitting data packets.

3.3.2 Overall validation

The overall results were also validated by comparing with the published NDMR results from Li [14].

The first aspect to be compared was the delay time. In order to validate the delay time of NDMR simulation results using the QoS enabled NDMR simulator (i.e. with no queue length information being passed) compared with the NDMR results of Li, the delay time has to be normalized in the graph, for delay time depends on many detailed implementation aspects: for example using a different MAC layer is likely lead to different delay time.

The result of Li's simulation is that the average delay of NDMR is basically the same for all mobile nodes. From Figure 3.11, it is shown that the simulation result is similar. They both show the delay time at all mobile velocities is almost equal. The difference between these two is

mainly because the using of different MAC layers, which contribute a lot to the transmission delay time.

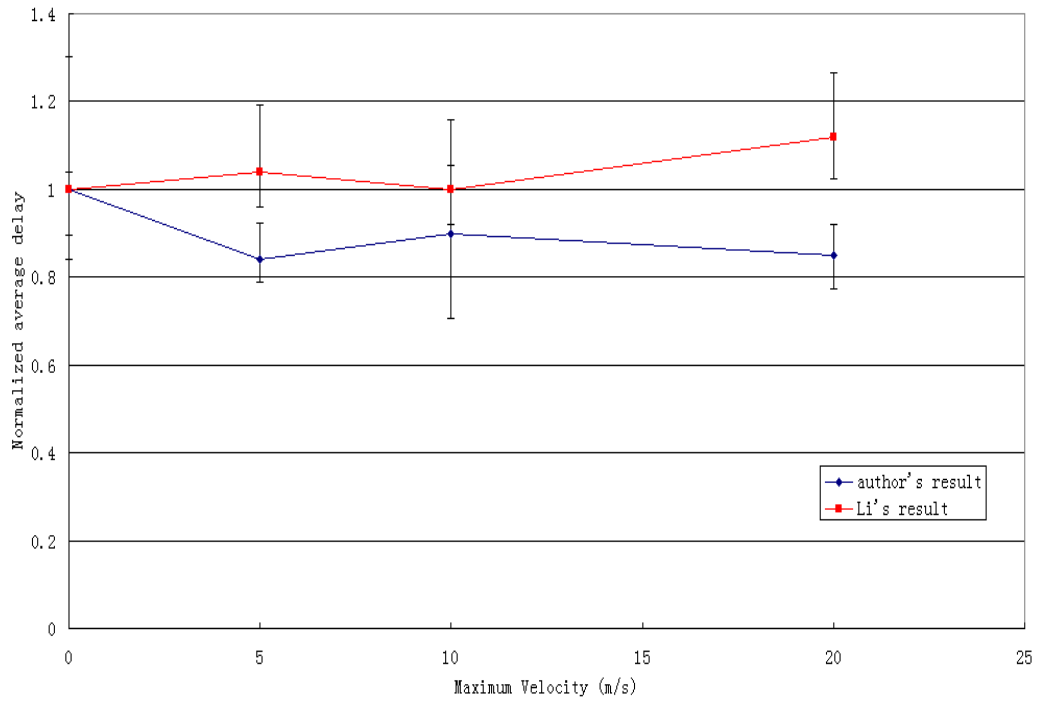


Figure 3.11 Comparison of normalized delay time

Li [14] also showed that more than 90% data packets for NDMR are sent successfully to the destinations at all moving speed under the conditions he specified. Figure 3.12 compares his results with those of the author and again it must be remembered that the details of simulation are not identical.

The packet delivery ratio achieved with the simulation here is slightly lower than Li's result. The main reason is that in the QoS enabled NDMR simulation, the maximum hop count is 10 while in Li's simulation it was 15. Those packets that require more than 10 hops to reach the destination will be lost in the simulation here. However, overall the results are very similar, especially as the y-axis on the figure is exaggerated.

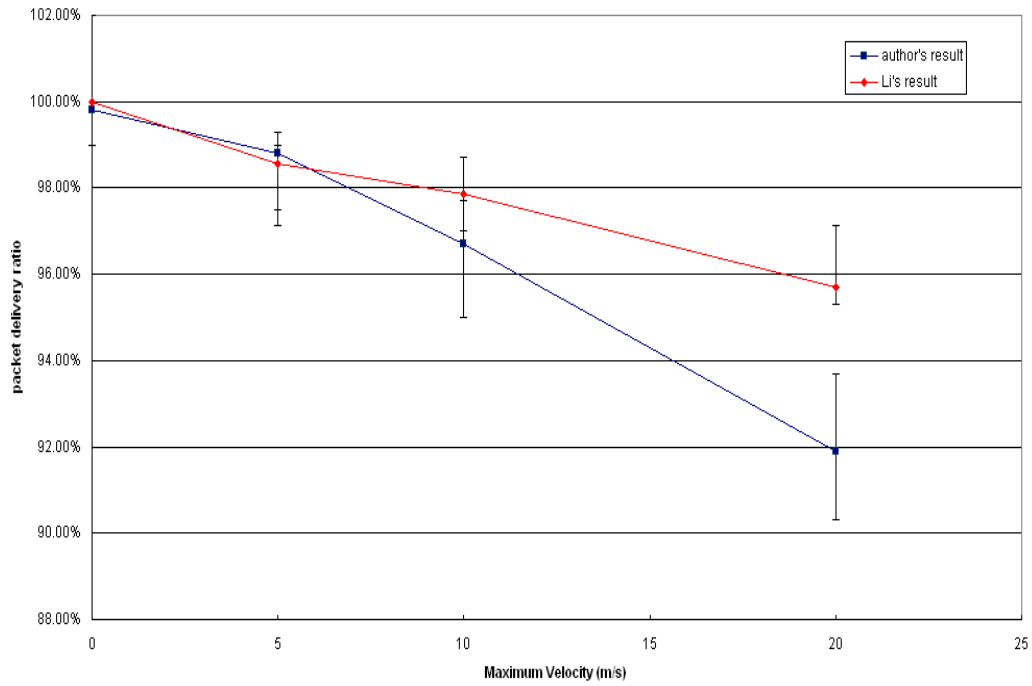


Figure 3.12 Comparison of packet delivery ratio

3.4 Two source and destination pairs

Although there are no common nodes in QoS enabled NDMR, there may be common nodes and common paths for *different* source and destination pairs. In the original NDMR, a large queue length in one common node will lead to a delay in all the paths (from different source-destination pairs) that contain that node as the source nodes will only change the path when the current path is not available anymore – and then the criterion for selecting the “best” path will be the number of hops, not the queue length or delay.

In QoS enabled NDMR, if the queue length of one node is very high, the first source node to be notified by an RREP or RUP packet will change to another low-delay path to transmit data, thus achieving a better end-to-end transmission delay. Figure 3.13 shows a simple network topology of two source and destination pairs with common nodes.

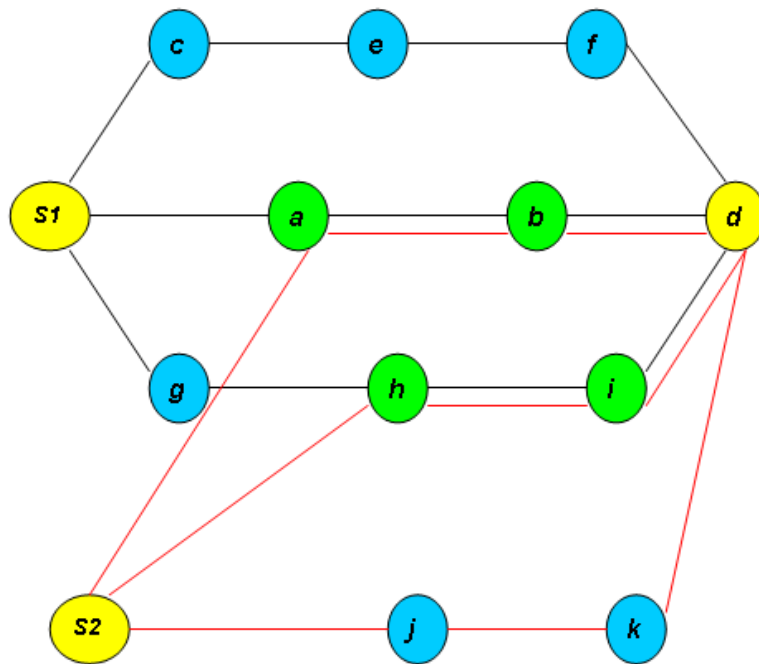


Figure 3.13 Two source and destination pairs with common nodes

Source and destination pair $(s1, d)$ and $(s2, d)$ both have three node-disjoint paths, but with four common nodes: a , b , h , and i . If there is a lot of traffic passing through node i and the queue length of node i is high, source node $s1$ or $s2$ will receive an RUP packet that contains the high queue length. The first source node receiving that notification would change its transmission path to another available route to balance the traffic flow through nodes and get a lower transmission delay time.

A simulation has been done to show the path change due to the high traffic at a common node. There are 50 nodes in the network and 20 nodes generate 500 packets to a random destination node. Figure 3.14 shows the network topology of the related nodes and Table 3.2 shows the source and destination pairs and their transmission paths.

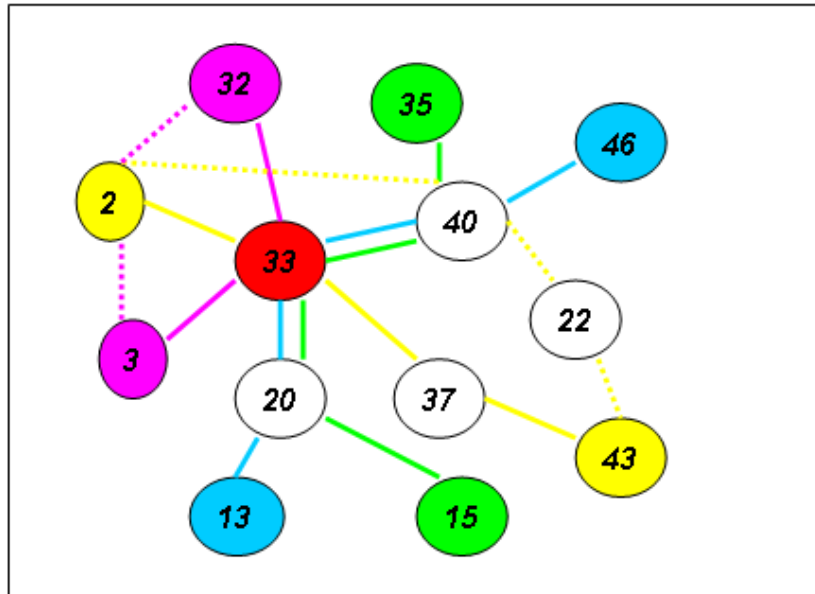


Figure 3.14 Network topology diagram

Table 3.2 (S, D) pairs and transmission paths

(S, D) pair	First Path(queue length)	Second Path(queue length)
(2, 43)	(2, 33 , 37, 43) (17)	(2, 40, 22, 43) (12)
(3, 32)	(3, 33 , 32) (10)	(3, 2, 32) (5)
(13, 46)	(13, 20, 33 , 40, 46) (22)	(13, 41, 37, 17, 46) (28)
(15, 35)	(15, 20, 33 , 40, 35) (20)	(15, 49, 18, 17, 35) (27)

From the table, node 33 is the common node of four source and destination pairs. At first, they all use their first path (which contains node 33 to transmit data). After sometime, with more packets generated for the pair (3, 32), source node 2 is the first to receives the RUP packet says and determines the queue length (17) is too long on this path, so it changes to the second path (2, 40, 22, 43) with queue length of 12 to continue forwarding data.

Source node 3 receives an RUP packet later as well and changes to path (3, 2, 32) for it has shorter queue length of 5 on that path. When source

node **13** and **15** receives the RUP packet, node **33** no longer has a longer queue length than other paths as the traffic has been distributed and they will still use the first path.

The simulation results shows in the following section will prove the advantage of QoS enabled NDMR in more detail.

3.5 Simulation results

In order to investigate the performance of QoS enabled NDMR under different network conditions, different parameters are varied in different tests to show how QoS enabled NDMR compares with other protocols.

These tests are grouped as follows:

1. Constant Bit Rate (CBR) source
2. Exponential Source
3. Different number of sources
4. Traffic with different priorities

3.5.1 CBR source

The first experiment is with CBR sources. This is the common type of data source in the simulation implementation. Traffic sources use 512 byte data packets, with a sending rate of 10 packets per second. There are 20 source nodes within the 50 node-networks, and in the network source-destination pairs are randomly spread. Results of running the same simulation to get the average delay with NDMR and QoS-enabled NDMR are shown in Figure 3.15.

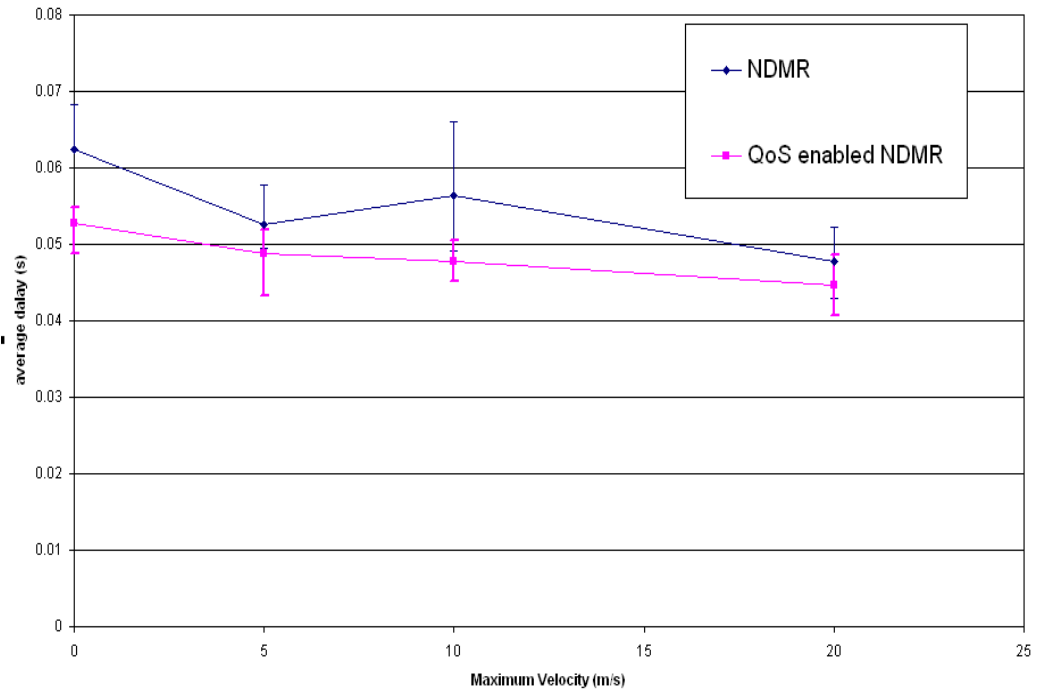


Figure 3.15 Average delay (CBR)

Figure 3.15 shows that for QoS enabled NDMR the average packet delay is smaller than that of NDMR. Average end-to-end delay time involves all the delays across the network, since the packet is produced at the source node to the moment it reaches destination node. In general, there are four aspects influencing delay time across the network:

- Route discovery time. It is the time packets waiting in a queue before a source node finds a route path.
- The number of hops in a routing path. It takes longer for a data packet to travel through a path with more hops (neglecting different queuing delays).
- The queuing delays across a routing path.
- Buffering waiting time. It is the time packets waiting in a queue before they can be served.

In Figure 3.15, the delay time for all mobile velocities tends to be almost equal for QoS enabled NDMR. The reason is that with RREP and RUP packets carrying real-time delay time back to the source, the source node is able to choose the best path according the queue length to transmit data, thus the packets will always be transmitted along the lowest congestion path and get a better end-to-end delay time.

The delay time goes down a bit as the speed of the nodes reaches 20m/s in both NDMR and QoS enabled NDMR scenario.

The main reason that it goes down in this simulation is that as the velocity increases the topology changes more quickly – this means that there will be (i) more route breaks but also (ii) more possibilities of routes with a small number of hops. Because NDMR has the fall-back routes in place already, the route break does not have a big impact so the ability to use shorter routes dominates – but it should be noted that the effect is small.

In Li's simulation of NDMR, the delay time goes up as speed goes up to 20m/s (Figure 3.16 in [14]). This is because he allowed a maximum of 15 hops in his simulation compared with 10 here. This reduces the number of breaks in the network, but allows longer routes and if a longer route is maintained then the delay will increase.

The error bars of these two protocols overlap a bit when maximum nodes speed are 5m/s and 10m/s and overlap even further when the speed increases to 20m/s; there is no overlap if the nodes are static.

When the nodes are moving fast, the network topology is changing a lot and different random seeds will lead to very different network situations being present – this increases variability of the results. When nodes are not moving there is no variability during the simulation so the variability

is much less.

The variability of QoS enabled NDMR is lower than that of NDMR which shows that with QoS support, the average delay time is decreased and network gets better performance because the system is always choosing the best path – i.e. the one with the shortest delay.

However, because of the variability, the seed leading to the worst case (highest delay) with QoS enabled NDMR is compared with the corresponding seed for standard NDMR.

Figure 3.16 shows this: the highest value of QoS enabled NDMR is marked ■ with the corresponding value from standard NDMR (for the same seed) being shown as ◆ - it can be seen that, as expected, the QoS enabled NDMR has a lower delay for the same run.

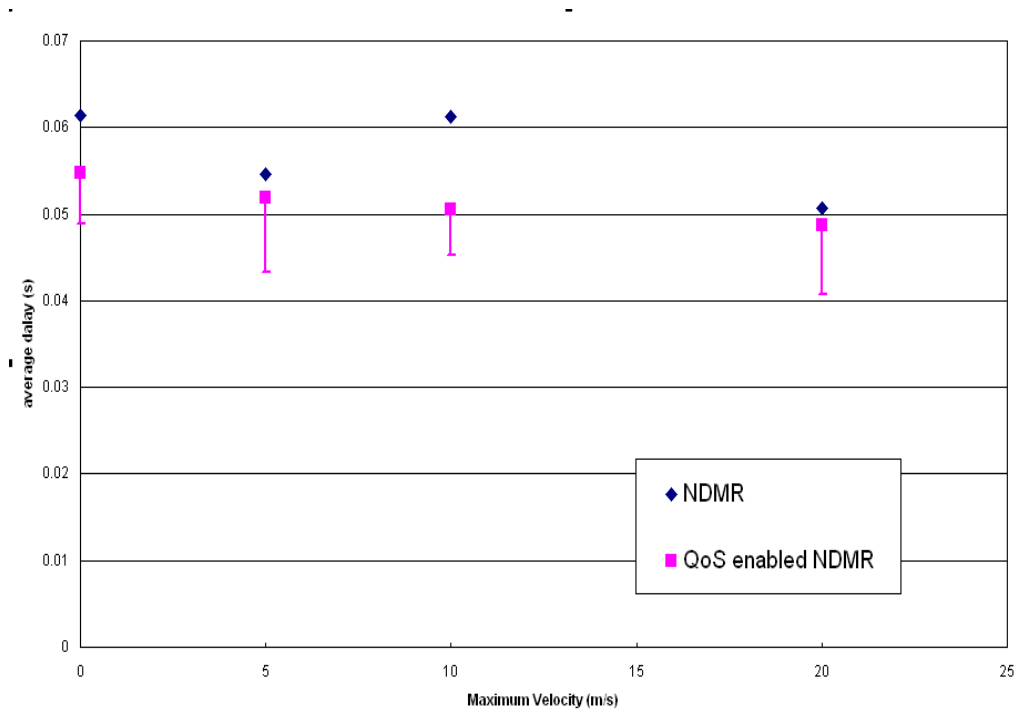


Figure 3.16 Error bar comparison of delay time (CBR)

The average transmission delay time of AODV, DSR, AOMDV and QoS enabled NDMR are also compared in Figure 3.17. The delay time of

AODV, DSR and AOMDV are the results from [14] and the delay time of QoS enabled NDMR is the author's simulation results run on the same scenario. Because the results in [14] included NDMR, the results here can be normalized to account for any small differences as some details of the simulation scenario may be slightly different. The results of NDMR from [14] and the author's NDMR results are normalized first, and QoS enabled NDMR was normalized as the same proportion as author's NDMR results. QoS enabled NDMR shows a better transmission delay time at all mobile velocities than AODV, DSR and AOMDV clearly.

The delay of QoS enabled NDMR is basically the same for all moving velocities and, as discussed above, is even better when the velocities increase. Delay time in DSR and AODV increases quickly with the increasing of moving velocity: when velocity is over 10m/s, the delay time of DSR and AODV increase even faster than that of QoS enabled NDMR.

It is for the reason that availability of multiple node-disjoint routing paths and the RUP packet in QoS enabled NDMR supports data being transmitted along the better path (as in a single path protocol, there is no better path to choose) and thus reduces route discovery latency and queue waiting time.

In QoS enabled NDMR, a source node will allocate data packets to the other node-disjoint paths according to the information in the RUP to avoid blocking when there is a congestion in the network. This approach decreases the packets delay time again.

AOMDV also provide smaller delay time than AODV because AOMDV also has backup routes and needs smaller route discovery overheads just as QoS enabled NDMR.

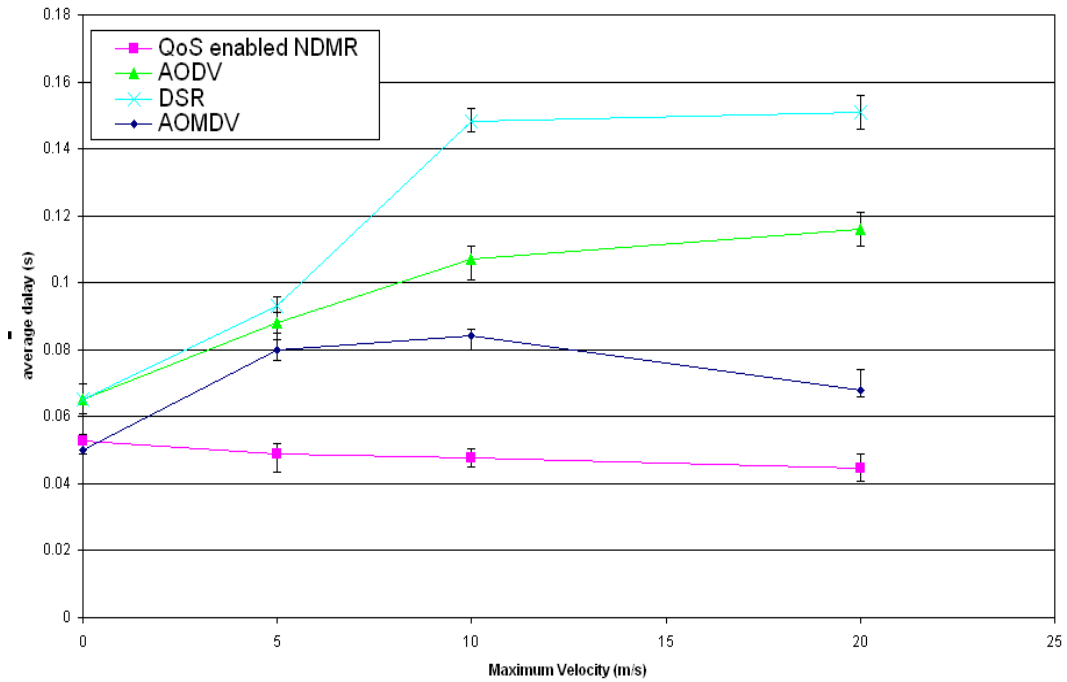


Figure 3.17 Average delay comparison

The packet delay time of a single source destination node is also shown in Figure 3.18. It is the delay time of packets transmitted from source node 19 to destination node 31.

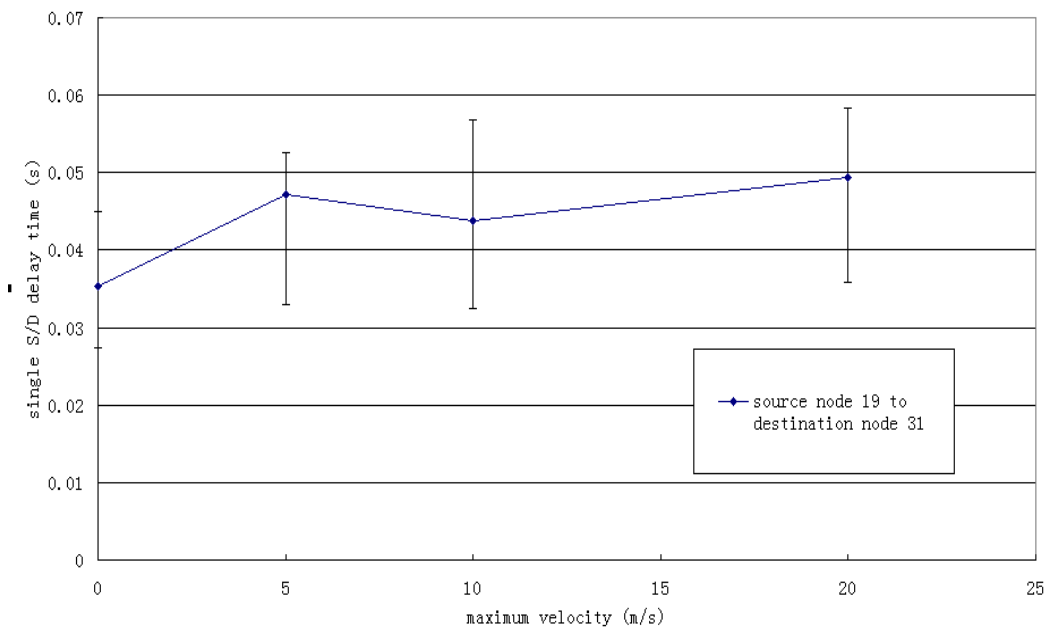


Figure 3.18 Single S/D delay time

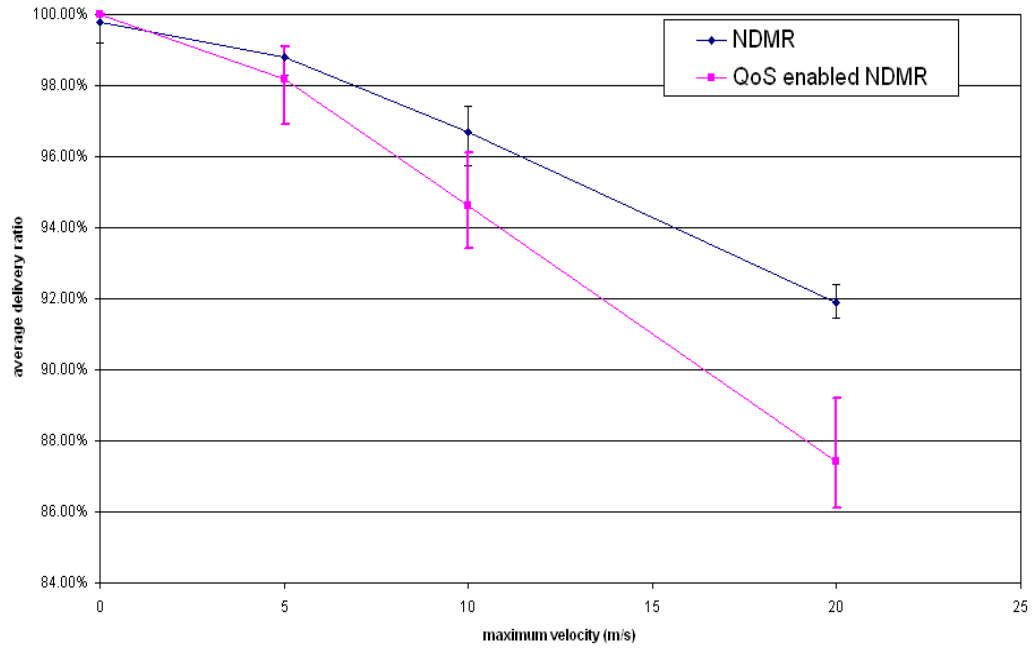


Figure 3.19 Packet delivery ratio (CBR)

Packet delivery ratio is another important metric for ad hoc networks: it can be seen from Figure 3.19 that QoS enabled NDMR has a worse packet delivery ratio (5% lower) than NDMR. This shows that the loss rate is slightly greater for QoS enabled NDMR, which in turn influences the maximum network throughput.

The reason for the lower packet delivery ratio is that the introduction of the periodic (per 5 seconds) route update packets (RUP) slightly increases the load and hence packet collision in the network since they are transmitted back to the source very frequently to report the real-time path information. However, the packet delivery ratio is still above 86% (the lowest error bar) for all mobile velocities, which is a satisfactory packet delivery ratio (in [83], a packet delivery ratio over 80% is deemed acceptable).

The packet delivery ratios of AODV, DSR, AOMDV and QoS enabled NDMR are also compared in Figure 3.20 (normalized values from [14]).

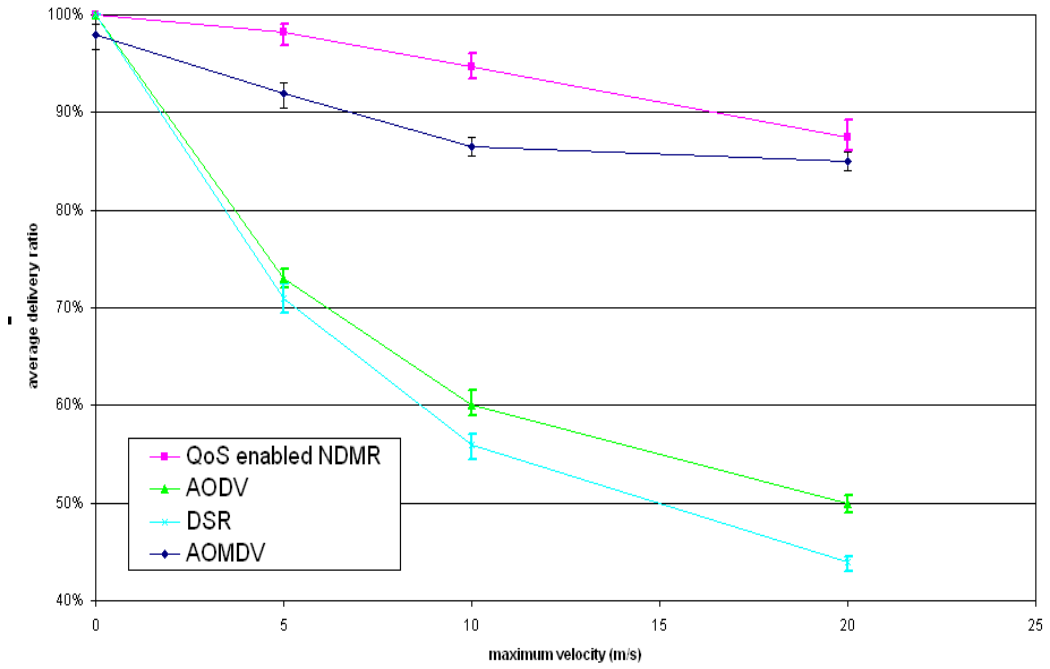


Figure 3.20 Packet delivery ratio comparison

The results show QoS enabled NDMR has a higher delivery ratio than AOMDV and much better packet delivery ratio than both DSR and AODV. More than 86% of the data packets of QoS enabled NDMR reach destinations at all mobility velocities. AOMDV has higher delivery ratio than DSR and AODV, both of which have a relatively low delivery ratio: at higher moving velocity only around 50% packets can reach destination.

The reason that QoS enabled NDMR shows better performance than AODV and DSR is that it contains multiple node-disjoint routing paths. If a routing path is broken because of mobility of nodes or has high value of queue length, the source node of the data flow will select another available node-disjoint routing path from the routing table to keep transmitting packets from source to destination node without any interruption.

QoS enabled NDMR has better packet delivery ratio than that of AOMDV. This is because QoS enabled NDMR is able to find multiple node-disjoint

routing paths but AOMDV is only able to establish link-disjoint paths. That means the movement of nodes has less influence on QoS enabled NDMR, as the source node in QoS enabled NDMR is able to transmit data packets along alternative routing path if a link failure happens or current route has high queue length. These mechanisms of QoS enabled NDMR strengthens its robustness on moving nodes and improves packet delivery ratio.

3.5.2 Exponential source

In this experiment, the source generates packets as exponential distribution. The average sending rate is still set to the same as CBR - 10 packets per second. It is more realistic than CBR as the real traffic for ad hoc networks as packets do not always arrive at a fixed rate. 512 byte Data packets are generated at the source node, the same as CBR. There are also 20 nodes which generate data packets in 50 nodes networks.

Figure 3.21 describes the average end-to-end delay time in terms of nodes velocity. It can be seen that QoS enabled NDMR gives better transmission delay time when the source generates packets exponentially (as well as in the CBR scenario) when there is QoS support. A source node can always choose the path with the lowest queue length to transmit data according the information that is transmitted back in the RREP and RUP packets.

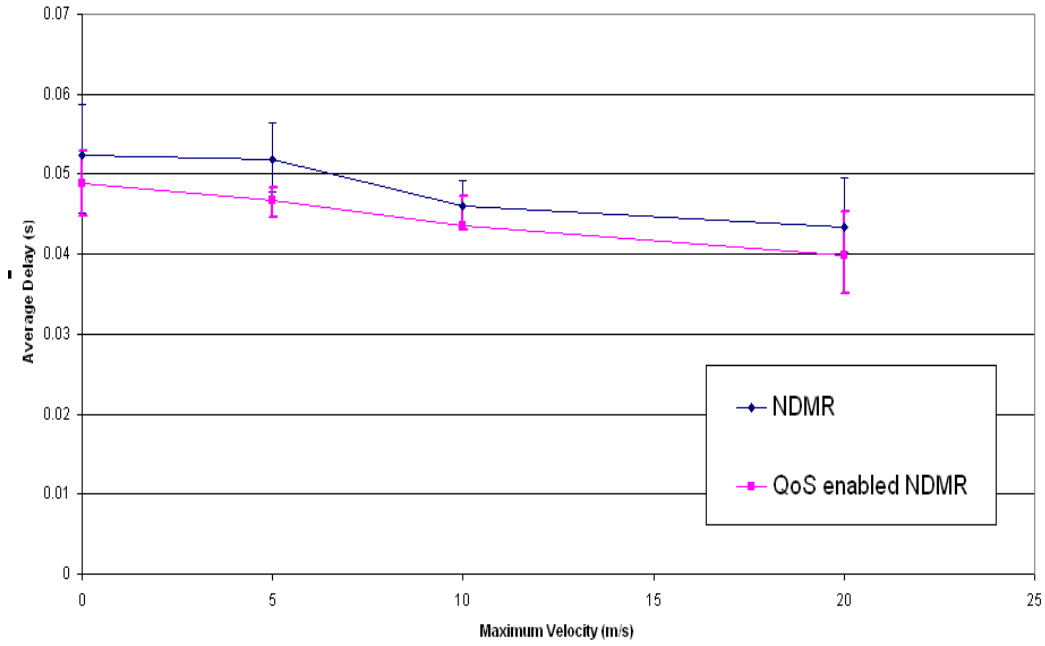


Figure 3.21 Average delay time (exponential source)

In order to show that QoS enabled NDMR achieves a better transmission delay time than standard NDMR in spite of the overlapping of error bars, the highest value of QoS enabled NDMR (■) is shown with the results from corresponding seed (◆) of the NDMR value.

All the highest values of the QoS enabled NDMR and the corresponding seed values of NDMR are shown in Figure 3.22. The delay time of QoS enabled NDMR is lower than that of NDMR for all mobile velocities which shows that QoS enabled NDMR is again achieving better performance than NDMR.

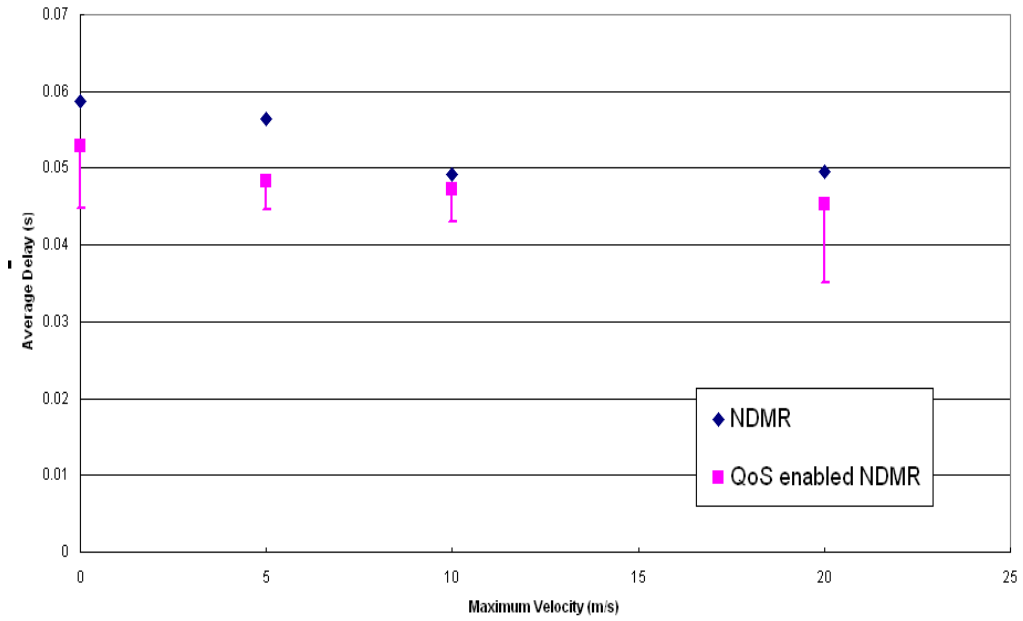


Figure 3.22 Error bar comparison of delay time (exponential)

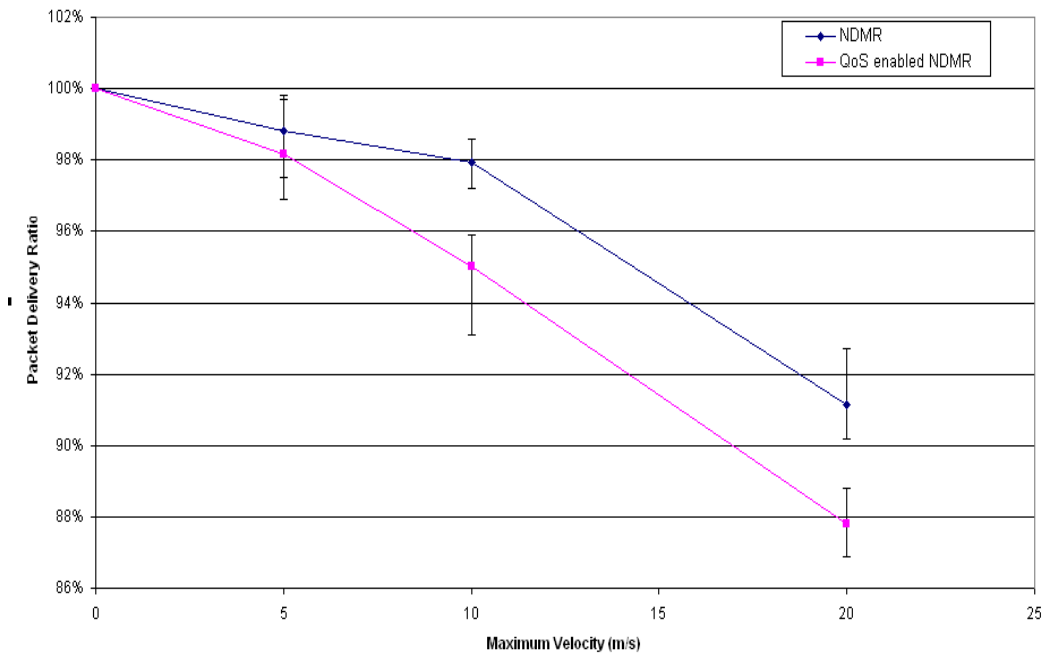


Figure 3.23 Packet delivery ratio (exponential source)

Figure 3.23 shows the packet delivery ratio in terms of nodes velocity. Packet delivery ratio of QoS enabled NDMR is again a little lower than NDMR as shown in Figure 3.23 for the same reason as before that the

introduction of the new RUP packets generates more traffic collisions in the network, but it is still acceptable as the delivery ratio values are all above 88% for all mobile velocities.

3.5.3 Different number of sources

In order to investigate the performance of different traffic load across the network, this set of experiments changes the number of sources from 10 to 50 in the 50 node network with the maximum moving speed of 10m/s. The packet sending rate is set at 10 packets per second CBR rate. The network load is changed according to a varying number of sources in the network.

Figure 3.24 shows the average end-to-end delay time in terms of the number of sources. From this figure, QoS enabled NDMR has a lower average delay time than standard NDMR even with the increasing of number of sources, although the values do get closer together as the number increases.

Again the reason for the better performance is that the RREP and RUP packets carry the real-time queue length to the source node so the source can choose the lowest-delay path. Although QoS enabled NDMR can always do this, as the number of nodes increases, congestion starts to appear so that the delay on all routes builds up and QoS enabled NDMR loses its advantage over standard NDMR and the results converge.

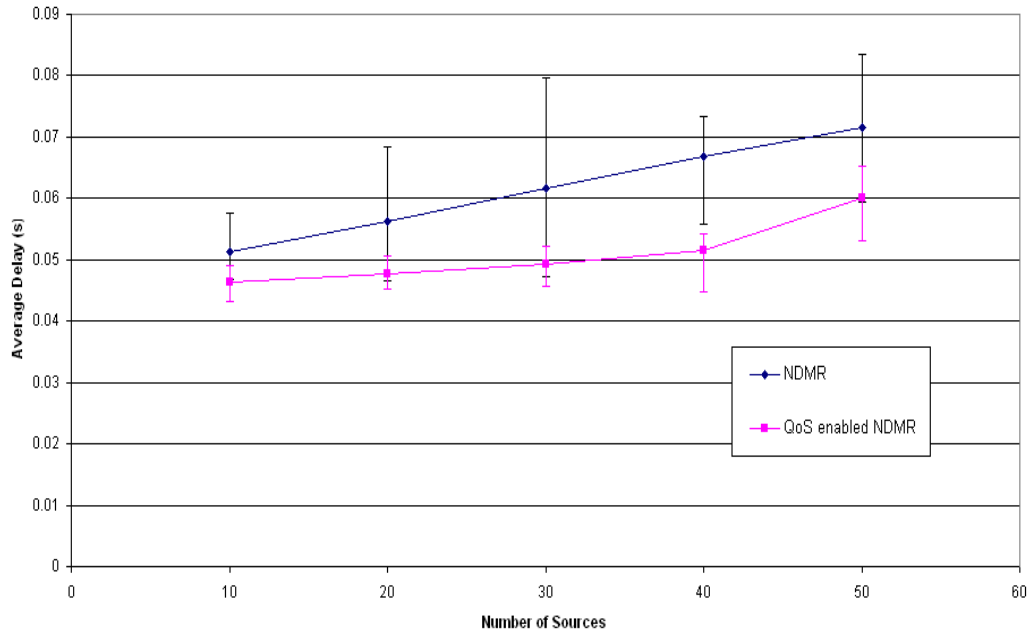


Figure 3.24 Average delay (varying number of sources)

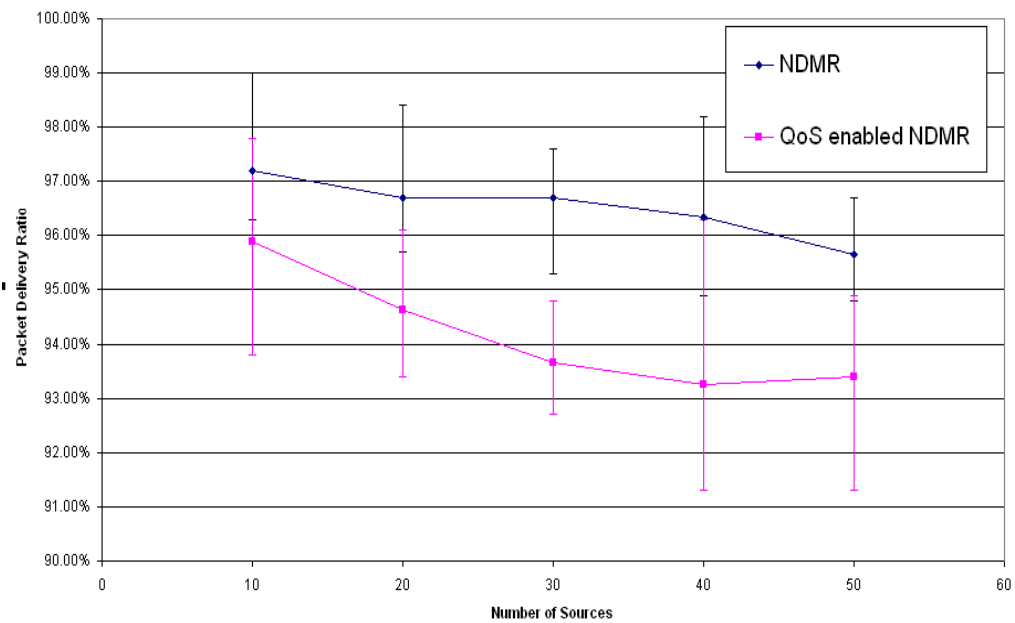


Figure 3.25 Packet delivery ratio (varying number of sources)

The packet delivery ratio of the two protocols as the number of sources increases is shown in Figure 3.25. It shows that the packet delivery ratio for QoS enabled NDMR is again a bit lower than standard NDMR as the introduction of RUP packets causes more collisions. The delivery ratio of

QoS enabled NDMR is more than 93%, but it reduces with the growing numbers of sources as there are more packet collisions across the network with more number of sources.

3.5.4 Different priority

As stated in section 2.7, DiffServ is a standard approach to achieve QoS in any IP network and could potentially be used to provide QoS in MANETs. In this experiment, it is implemented on the network layer for testing rather than on the MAC layer in [14]. There are mainly two types of traffic in a network: Expedited Forwarding (EF) and Best Effort (BE) traffic.

- EF traffic [68] [69]: guarantees minimal delay and low loss. A queue that for EF traffic has greater service rate and lower congestion possibility, which will result in delay and loss in the network. Typical EF traffic examples are voice, video and other multimedia streams, which are regularly produced and need lower delay time and loss ratio.
- BE traffic: there is no QoS guarantee for BE traffic. All users of BE traffic obtain best effort service that they obtain unspecified delivery time and delivery ratio.

It is necessary to let the EF traffic (which requires low packet delay time) to be transmitted on lower delay time path and BE traffic on other node-disjoint paths for the same source destination pair. With QoS enabled NDMR, the source is able to choose the best path for EF traffic and BE packets are supposed to be transmitted on a higher delay time path to meet the different requirement.

In this experiment, the traffic source is CBR with 512 byte data packets. In order to investigate the network performance, 20 sources transmitted with 10 packets per second and velocity is changed from 0 to 20 m/s with intervals of 5. Background BE traffic are generated by 20 other nodes at 2 packets per second.

When BE traffic suddenly increases at a node, the queue length of that node will increase. With RUP packets transmitted back to the source node periodically, the source node is able to know the queue length of the path containing that node and always chooses the path with the lowest queue length transmitting EF traffic. This, therefore, guarantees EF traffic has a better transmission delay time than BE traffic all the time.

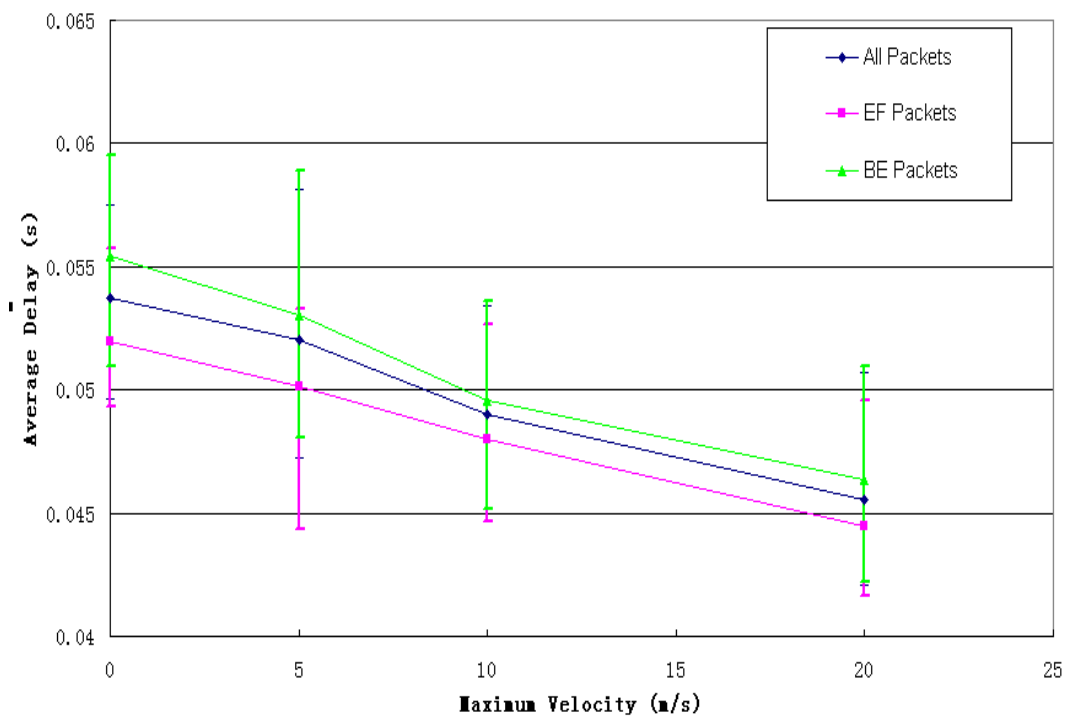


Figure 3.26 Average delay of different priorities

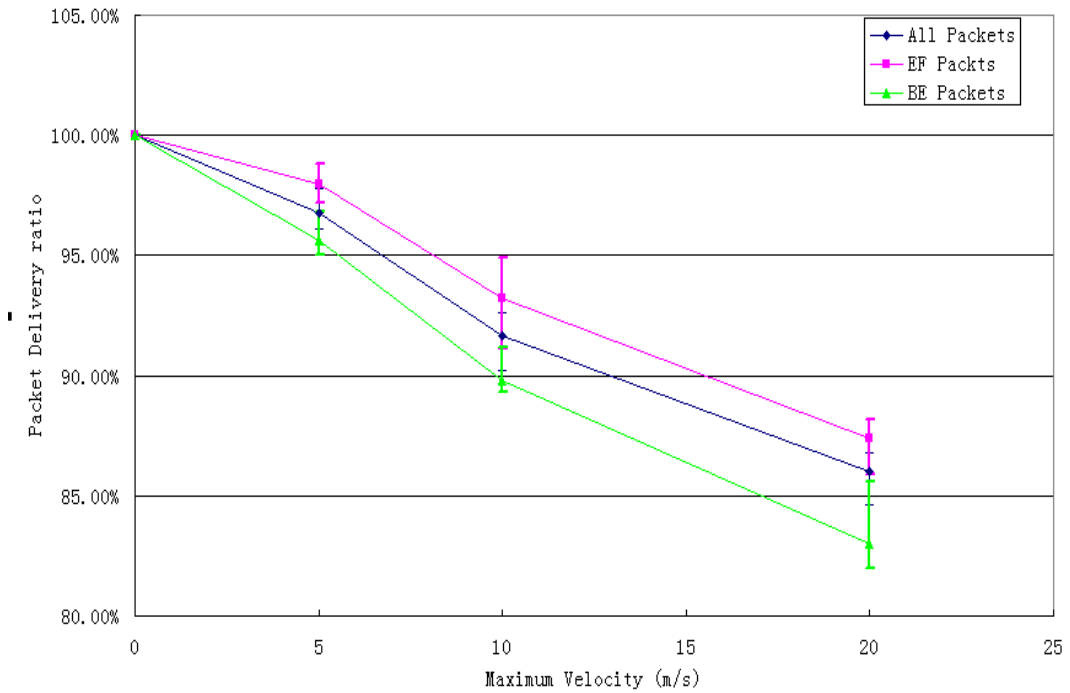


Figure 3.27 Packet delivery ratio of different priorities

Figure 3.26 and Figure 3.27 show the average delay time and delivery ratio in terms of velocity of nodes of different priority traffic. EF traffic gets the lower delay and better delivery ratio than BE traffic. When the maximum velocity increases, the protocol discards more packets than that with low speeds. This is because the rapidly changing network and more frequent path breaks cause a greater number of overhead packets to be transmitted – and the protocol puts the BE traffic on the more heavily congested routes so leading to a higher drop rate.

The error bars of different priority traffic overlap each other, but in all cases (even with individual seeds shown in Figure 3.28 and Figure 3.29). EF traffic always does better than BE traffic – as is expected.

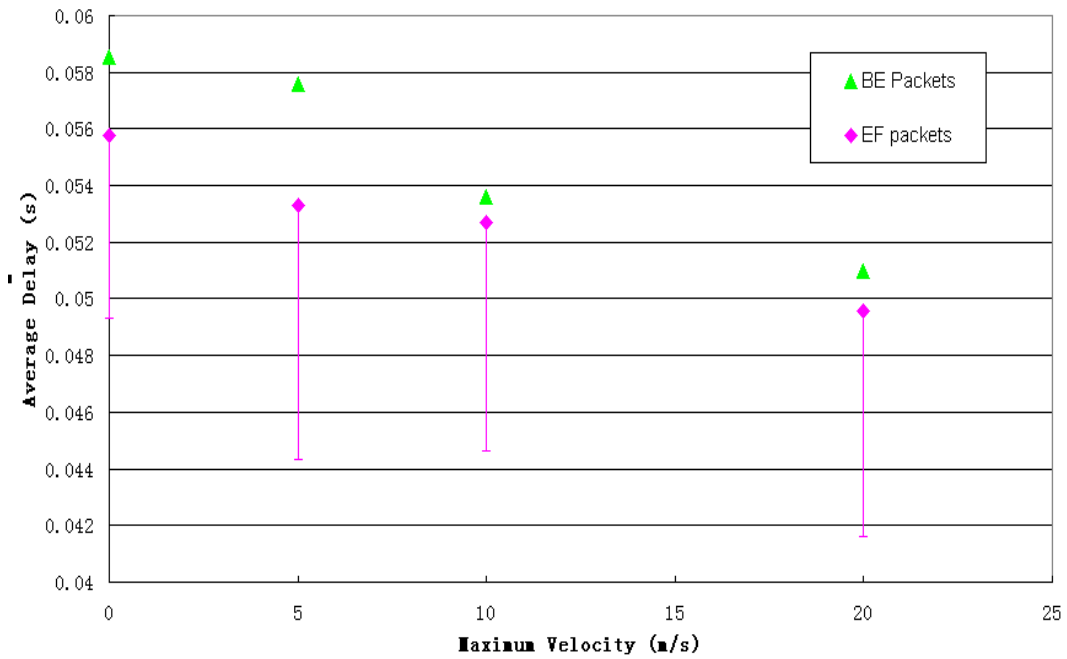


Figure 3.28 Error bar comparison of average time (different priority)

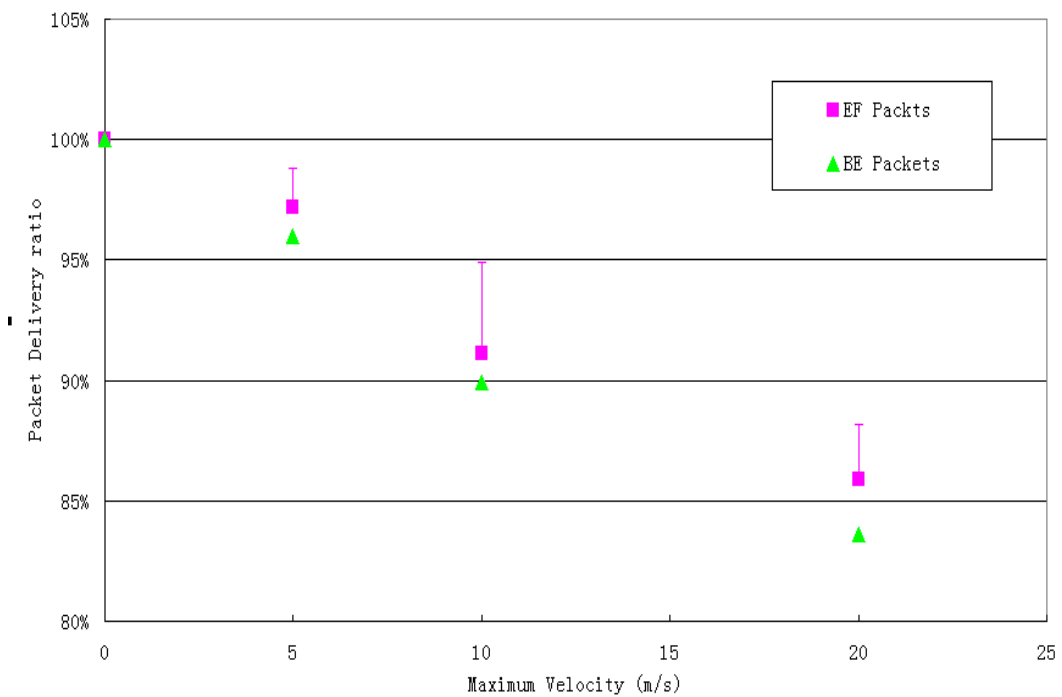


Figure 3.29 Error bar comparison of delivery ratio (different priority)

3.5.5 Packet sequence integrity

To efficiently utilize multiple access lines for single file transfer, multipath TCP communication methods have attracted interested [give refs]. A multipath TCP enables simultaneous distributed data transfer between two end-points on multiple TCP connections. However, these methods cannot fully utilize the available bandwidth of multiple paths because they do not properly consider the end-to-end delay of packet transmission, so out-of-order data arrival at the destination node causes a bottleneck in data sort operations. This problem is more severe in environments where the quality of each path is different or unstable, such as in MANETs.

Arrival-Time matching Load-Balancing (ATLB) [84] continuously calculates the delay of each path, including the TCP queuing delay at a source node and the network delay, and then sends a data packet through the TCP connection with the lowest end-to-end delay. ATLB consists of a data distribution method to reduce the cost of data alignment in a destination node and a path-failure detection and recovery mechanism to prevent delay of the data transfer.

With QoS enabled NDMR, the source node will change path to transmit packets automatically according to the real time path delay, which may cause the “out-of-order” problem: will the packets arrive at the destination in sequence? To answer that a sequence number is put into each packet in the simulation, the results show that with EF packets, all packets arrive at destination in sequence and only 1 in 4000 BE packets do not arrive at the destination in sequence. The upper layers can completely deal with such a low out-of-sequence rate (e.g. ATLB) rather than trying to solve the problem in the routing layer.

3.6 Summary

In this chapter, a novel and useful QoS enabled NDMR in mobile ad hoc networks is proposed. With the new RUP packets carrying real time “queue_length” field, the source node is able to transmit packets along the optimum path.

A simulation model of QoS enabled NDMR has also been implemented by the author and performance results for routing protocols of NDMR and QoS NDMR are shown and compared in different simulation scenarios.

Simulation results show that the performance of QoS enabled NDMR is better than the standard NDMR.

The next chapter will present an improved routing protocol based on QoS enabled NDMR.

4 QoS ENABLED NDMR IN A MULTI-RATE ENVIRONMENT

4.1 Introduction

Multi-rate feature is an important aspect of wireless communications. Wireless devices work on different transmission rates depending on channel quality and so can operate at a high data rate when the channel quality is sufficiently good, rather than staying at the lowest possible rate that will cover all eventualities. IEEE 802.11b provides the multi-rate capability for wireless LANs and ad hoc networks; it specifies four modulation formats and data rates: the basic access rate based on 1Mbps DBPSK modulation, the enhanced access rate based on 2Mbps DQPSK, and two High Rate access rates based on the CCK modulation scheme for 5.5Mbps and 11Mbps [43]. 802.11g is the third modulation standard for Wireless LAN. It works in the 2.4 GHz band (like 802.11b) but operates at a maximum raw data rate of 54Mbps. 802.11g also provides multi-rate capability. The modulation scheme used in 802.11g is OFDM copied from 802.11a with data rates of 6, 9, 12, 18, 24, 36, 48, and 54Mbps, and reverts to CCK (like the 802.11b standard) for 5.5 and 11Mbps and DBPSK/DQPSK for 1 and 2Mbps[78].

Higher data rates are commonly achieved by more efficient modulation schemes and the ability to transmit at multiple data rates is available at the physical layer for many wireless local area networking devices today.

Rate adaptation dynamically switches data rates to match the channel conditions, in order to select the rate that will give the highest throughput for the channel conditions.

Channel quality is measured by SNR (Signal Noise Ratio) at the receiver, which depends on the transmission power, the interference and the

distance between the source and destination nodes. Rate selection uses the channel quality to select an appropriate rate. A variety of techniques ([44], [45] and [46]) have been reported but a common technique is threshold selection [44], where the value of a parameter is compared against a set of values and the appropriate for that value is chosen.

As is well known, a higher data rate requires a better channel quality. When other factors are the same, the SNR is higher when the distance between the source and destination nodes is shorter. In this chapter, distance will be used as a measure of signal quality – however, this is only to demonstrate the principle and more accurate representations could be used.

In this chapter, an overview of topology for multi-rate in ad hoc networks is presented. Then an adaptation is proposed to implement multi-rate transmission on the physical layer based on the QoS enabled NDMR to minimize overall delays.

4.2 Overview of Multi-Rate ad hoc networks

4.2.1 Receiver-Based Autorate (RBAR) protocol

The concept of RBAR [46] is to allow the receiver to select the appropriate rate for the data packet during the RTS/CTS packet exchange. Advantages of this approach include:

- Both channel quality estimation and rate selection mechanisms are on the receiver to give more accurate rate selection as the process that estimates the channel quality can thus directly access all of the information made available, such as the number of multipath components, the symbol error rate and the received signal strength.

- The rate selection is done during the RTS/CTS exchange so that the channel quality estimates are performed nearer the time the data packet is to be sent, rather than during previous packet transmission schemes as is the case with other schemes ([70] and [71]).
- It can be implemented into IEEE 802.11 with minor changes.

4.2.2 A Relay-Based MAC (RMAC) to support multi-rate feature

In RMAC [45], the packet transmission process is split into two phases: handshaking and data transmission.

- Handshaking is used to reserve the wireless channel, to exchange the channel quality information and to decide what data rates will be used in the data transmission phase.
- The data transmission phase can be performed either in direct mode (where the transmitter sends the data packet to the receiver directly) and relay transmission (where the transmitter transmits the data packet to the relay and the relay forwards the packet to the receiver). The decision as to which mode is taken is based on the information exchanged at the handshaking phase.

Although relay uses one intermediate node, it is not ad hoc and also the aim is to select the best data rate and mode of transmission rather than to make use of a multi-rate environment – so it will not be considered further.

4.2.3 High throughput reactive routing in multi-rate ad hoc networks

This routing protocol [44] is probably the closest existing protocol to the concept presented here, although it is still very different. The approach is

to use the MAC delay as a metric when choosing high throughput routes in AODV. This is a very useful metric to identify congestion hot spots and measure link interference in an ad hoc network. For 802.11 MAC, under certain assumptions such as zero bit error rate and no collision losses, the MAC delay time per frame is in the form of

$$ax + b \mu s$$

where x is the frame length in bytes, and a and b are parameters of data rates and modulation schemes [72]. When using MAC delay as the routing metric, we assign each link a cost proportional to its theoretical MAC delay given above. Table 4.1 gives the MAC delays and link costs for different 802.11 b rates. The aim of the routing protocol is to find the least cost path from the source to the destination.

Table 4.1 MAC delay and link cost ⁹

Link rate, Mbits/s	MAC delay, μs	Link cost
1.0	9814	4.28
2.0	5678	2.48
5.5	3046	1.33
11.0	2294	1.0

In this protocol, the basic AODV protocol is modified by changing the hop count metric used in AODV to the new cost metric based on MAC delay. A new “path cost” field is appended to the RREQ and RREP packets. When an intermediate node receives an RREQ, it increases the cost value in the RREQ by the “cost” of the link over which it received the request, and then re-broadcasts the RREQ packet. During the RREQ

⁹ Table 1 from [44]

forwarding process, intermediate nodes update their route table only when the new path cost is smaller than the previous one.

Multi-path routing is not supported in standard AODV protocol as it always chooses the minimum hop count path. However, discovering multiple paths is achieved in this protocol and better (lower-delay and higher throughput) paths (not necessarily the minimum hop count route) are acquired for transmitting data.

This is done by requiring intermediate nodes to retransmit a duplicate RREQ if the new path has a smaller path cost than the previous one. After receiving the first RREQ packet, the destination waits for a timeout period so that it can learn all the possible routes, which it does by accepting more than one RREQ from different neighbours. The destination chooses the route with the least path cost then generates and transmits an RREP packet back to the source node along the reverse path.

Figure 4.1 shows a simple topology (taken from [44]) of an ad hoc network with node connectivity and available data rate of each link. The traditional shortest-path routing protocols will select route A-B-C for the source destination pair of A-C as this has the minimum number of hops. However, route A-D-B-E-C consists of higher data rate links (although with increased hop count), and could achieve higher throughput if chosen.

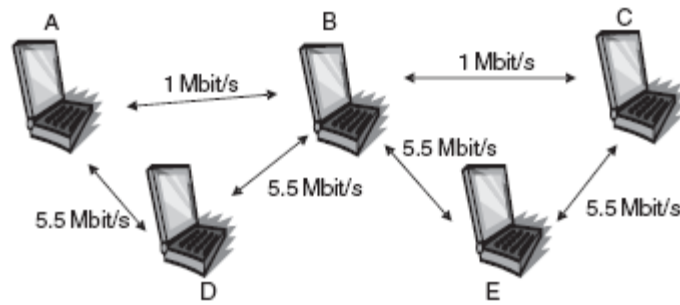


Figure 4.1 Simple topology with Multi-rate support ¹⁰

All the multi-rate routing protocols mentioned above do not use the node-disjoint routing supporting multi-rate, which cannot achieve the best multi-rate performance along multipath. In the next section, a routing protocol combining the node-disjoint and multi-rate is proposed.

4.3 Multi-rate QoS enabled NDMR

4.3.1 Introduction

There has been no previous work in the literature on including the advantages of multi-rate transmission on NDMR. This section describes how that capability can be added to QoS enabled NDMR.

In the previous chapter, it was shown that QoS enabled NDMR achieves better transmission delay by selecting the path with lowest queue length, but all packets are transmitted with the same transmission rate on the physical layer so that the approach does not take advantage of the multi-rate feature in wireless communications.

The delay at a node depends on the transmission rate as well as queue length, so that, if advantage is taken of multi-rate capability when

¹⁰ Figure 1 from [44]

implementing the QoS enabled NDMR, the overall delay time can be lowered because it is the delay time that is now being considered directly.

Link rate, as explained earlier, depends on the SNR value – and received signal power decreases with distance, so that generally the SNR decreases as the distance between source and destination increases.

In this thesis, distance is used as a first approximation to SNR so that packets can be transmitted with a higher rate when the distance between nodes is short and with lower rate when the distance is long.

This simplification is made because the point here is to demonstrate the ability of QoS enabled NDMR to take additional factors, like distance into consideration. It would be perfectly possible to replace the measure “distance” with the algorithm that implements 802.11 rate variation, but the principle would remain the same.

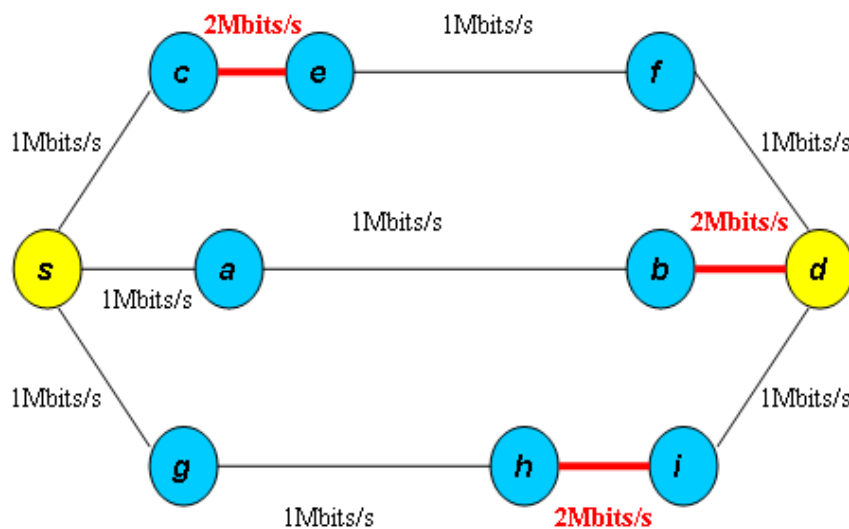


Figure 4.2 Simple topology of Multi-rate

As an example, Figure 4.2 shows a simple topology of a node-disjoint ad hoc network together with node connectivity and available data rate of each link. In the standard NDMR or QoS enabled NDMR routing protocols, all nodes transmit packets at the same rate, but in this figure, higher rates can be used where the nodes are closer together. Of course, nodes move and the distance between them changes, so that this figure represents just a snapshot.

In this multi-rate featured protocol, a node is enabled to transmit packets at higher rate if the link distance is below a threshold. This is called *multi-rate QoS enabled NDMR*.

In the examples here, the distance threshold is set to 100m (remember the maximum transmission range is 250m) and below that distance the higher rate can be used.

In the snapshot in Figure 4.2, the node distance between node *c* and *e*, node *b* and *d*, and node *h* and *i* are all less than that threshold so that packets between these paths will be transmitted with a higher rate to and hence the transmission delay time will be reduced.

4.3.2 Multi-rate Simulation Model

To evaluate the effectiveness of QoS enabled NDMR with Multi-rate support; its performance is compared with that of standard QoS enabled using a simulation built in OPNET.

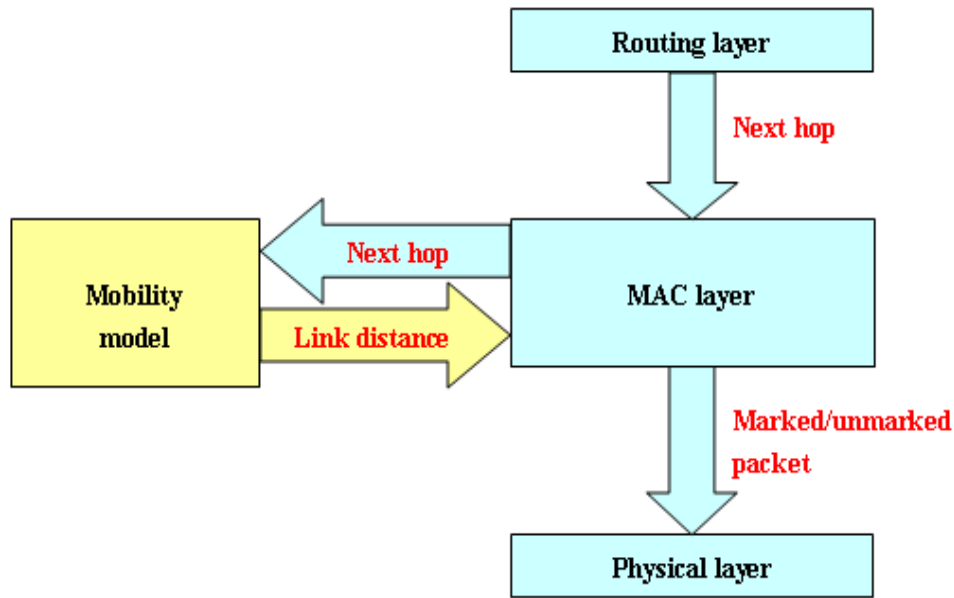


Figure 4.3 Flow chart of multi-rate implementation

A flow chart illustrating the implementation of multi-rate QoS enabled NDMR is shown in Figure 4.3. In order to adjust transmission rate at the physical layer according to the link distance, global variables are used to pass parameters between layers. When a packet gets all the routing information it needs, it will be passed to the MAC layer. Before passing the packet to physical layer, the MAC layer looks up the link distance in the mobility model through a global variable using the “next_hop” field in the packet. If the distance is lower than the threshold (100m), it marks the packet and passes it to the physical layer.

If the physical layer receives a marked packet, it will adjust the transmission rate and forward the packet at the higher rate of 2Mbits/s; otherwise, the packet will be transmitted at the basic rate of 1Mbits/s.

4.3.3 Mobility and Traffic Model

Random waypoint model [29] is applied to simulate nodes movement. Each node moves from a random start position to a random destination at a certain speed. After reaching the destination, another random destination is selected after a pause time (here 30s). 50 nodes move around within a 1000m x 1000m area field. The radio transmission range of each node sets to 250m.

Source nodes generate 512 bytes packets at CBR (constant bit rate) or exponential rate. All nodes are located randomly across the network. The number of sources also could be changed for the experiment of different the traffic load in the network. The higher sending rate is used when the link distance is less than the threshold 100m; in other cases, the basic rate is used.

A simulation runs for 600 simulation seconds. One result point is the average value of five runs with different random seeds. The maximum and minimum values (error bars) are shown in the figures.

4.4 Simulation results

4.4.1 CBR source (constant bit rate)

The first experiment is using CBR sources, each sending 512-byte data packets and there are 20 source nodes within the network of 50 nodes. The packet sending rate at the physical layer is set to 1Mbits/s for link distances more than 100m and 2Mbits/s if the link distance is less than that.

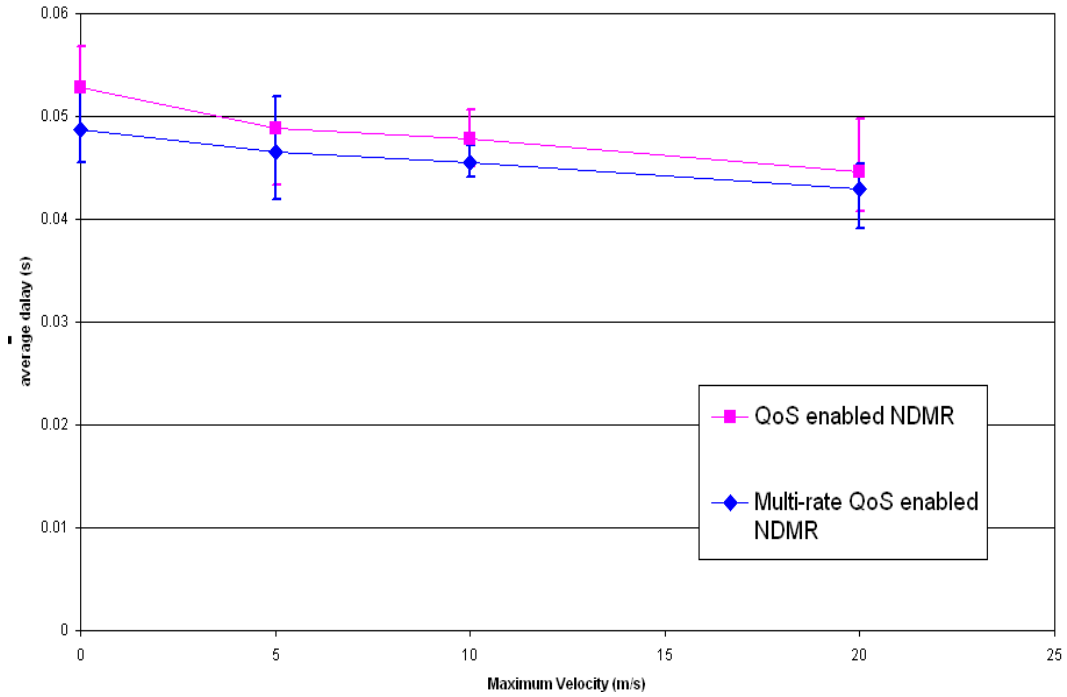


Figure 4.4 Average delay comparison (multi-rate CBR)

The simulation results in Figure 4.4 show that by allowing multi-rate transmission, the overall delay is reduced again at all mobile velocities compared with the standard QoS enabled NDMR. With multi-rate implemented in the QoS enabled NDMR, when packets are transmitting at a higher speed, the queue length of the node along that path will shrink, thus reduce the transmission delay of that path and across the network.

This is expected as the multi-rate approach implemented in the QoS enabled NDMR accelerates the packet forwarding when the link distance is small and hence reduces the overall transmission time. The results confirm that this is indeed happening.

The error bars of two routing protocols do overlap, but again the highest value of QoS enabled NDMR (■) is higher than the corresponding multi-rate run (◆) - Figure 4.5.

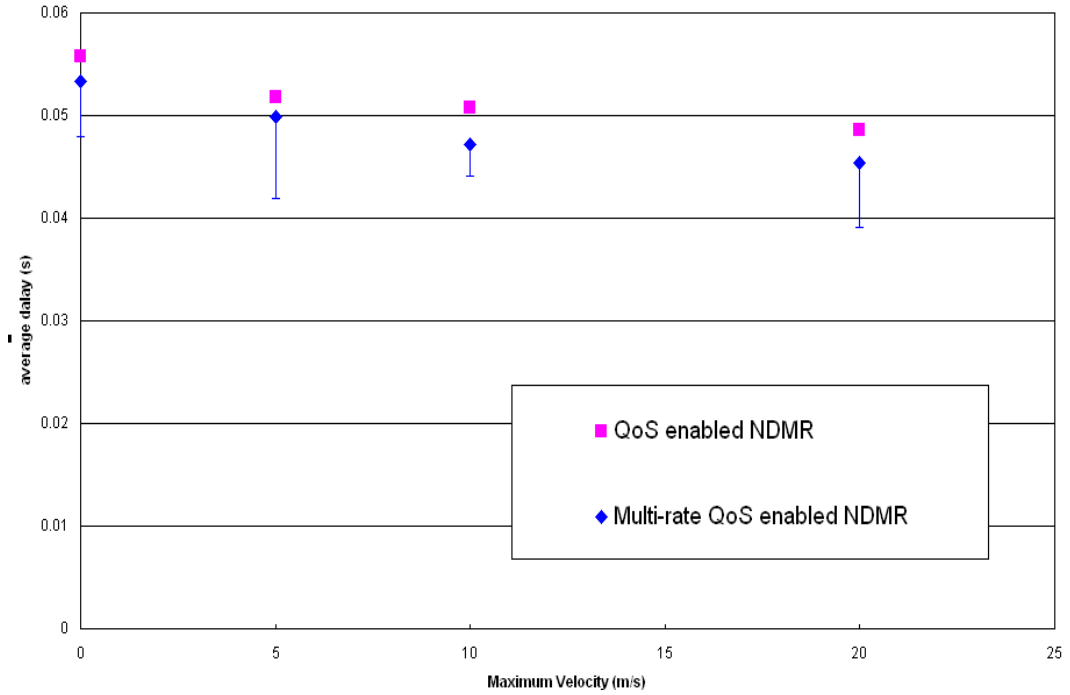


Figure 4.5 Error bar comparison of delay time (CBR)

4.4.2 Exponential source

In this experiment, the real-time source generates packets as an exponential distribution. This is a more realistic approximation than CBR for real mobile ad hoc networks. Traffic sources again have 512 byte data packets and there are 20 source nodes within the 50-nodes network. The packet sending rate at physical layer is set to the same as CBR - 1Mbit/s and 2Mbits/s.

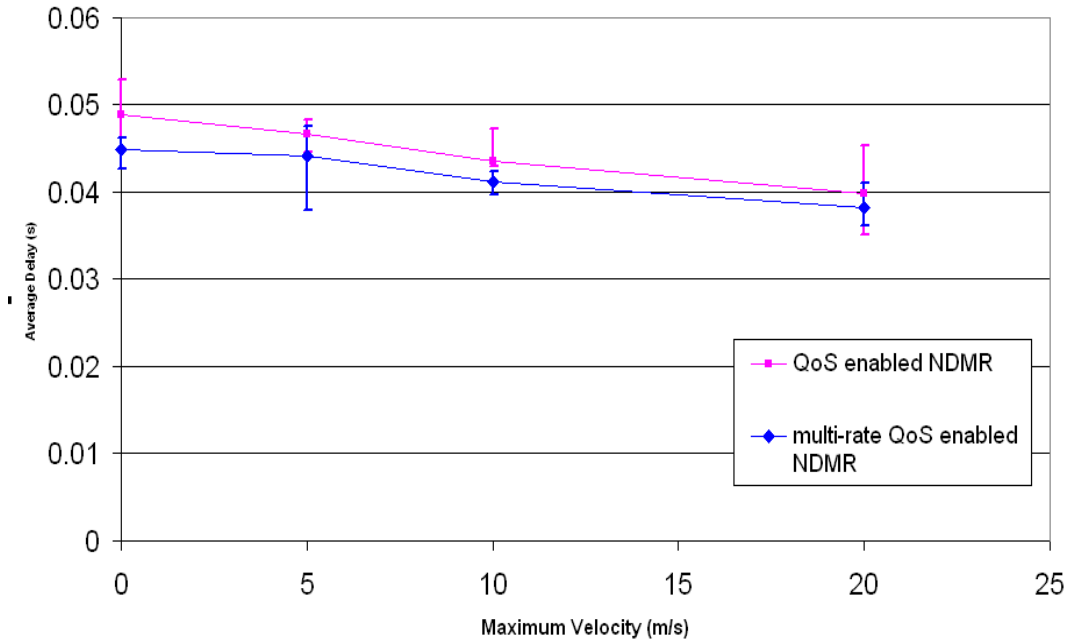


Figure 4.6 Average delay comparison (multi-rate exponential source)

Again, although error bars overlap, the corresponding runs (Figure 4.6) still show that the multi rate approach is better.

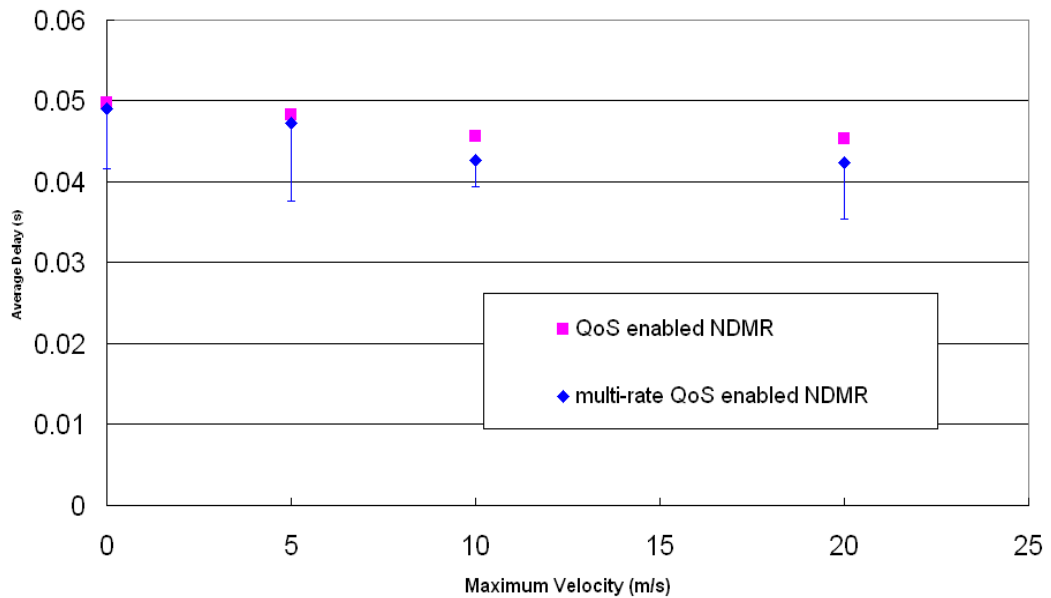


Figure 4.7 Error bar comparison of delay time (exponential)

Values with corresponding runs of these two protocols are also drawn in Figure 4.7 to show the delay time comparison.

4.4.3 Different number of sources

This set of experiments changes the number of sources from 10 to 50 in the 50 nodes network with the maximum moving speed of 10 m/s. The packet sending rate sets at 10 packets per second CBR rate as before. The network load is changed according to different number of sources in the network.

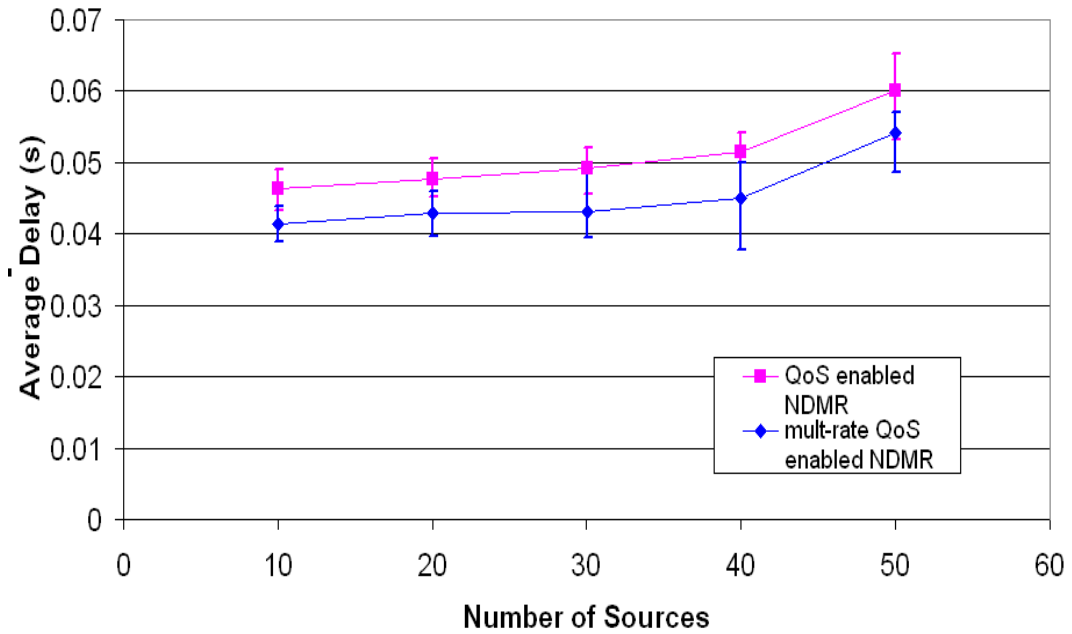


Figure 4.8 Average delay comparison (different number of sources)

The average delay time shrinks for all number of sources with the support of multi-rate in Figure 4.8. With multi-rate support, the data can be transmitted at a higher rate when the distance is short; the pure QoS enabled NDMR allows the data transmitted all at the lower base rate. Thus, multi-rate archives better performance. The trends of the two delay time lines are the same as multi-rate only changes the transmission rate when node distance is low, other factors in the scenario remain the same.

A comparison with corresponding runs is given in Figure 4.9 to show that multi-rate QoS enabled NDMR still gets lower transmission delay with different number of sources.

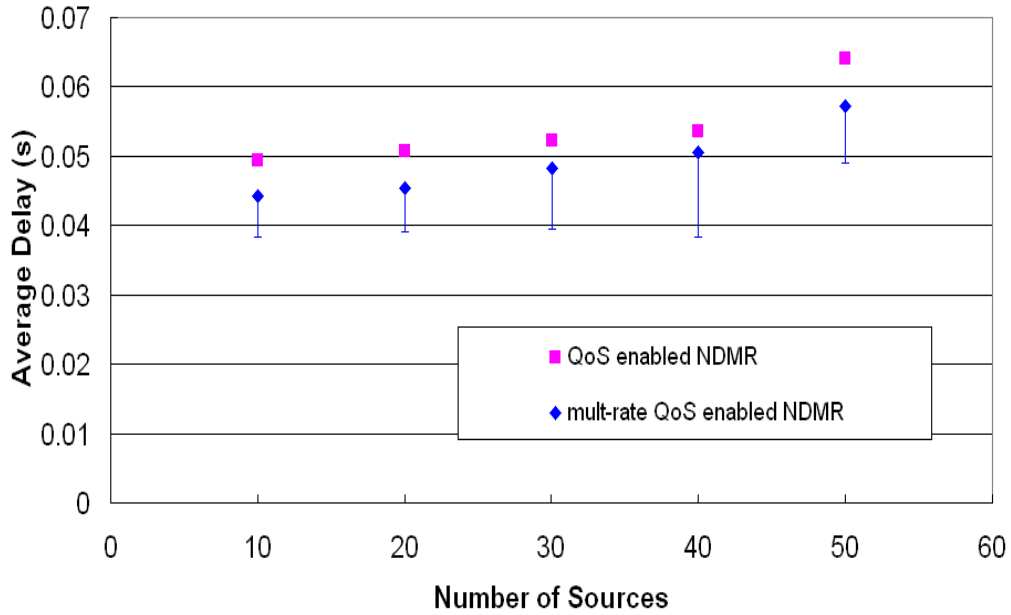


Figure 4.9 Error bar comparison of delay time (different number of sources)

4.5 Summary

In this chapter, a new protocol which combines multi-rate feature and QoS enabled NDMR is proposed. This takes advantage of multi-rate availability in mobile ad hoc networks and allows lower end-to-end transmission delay by using different transmission rates according to different distances between nodes.

The performance evaluation and comparison between QoS enabled NDMR and multi-rate QoS enabled NDMR are studied by extensive simulations that show that QoS enabled NDMR with multi-rate achieves better performance than pure QoS enabled NDMR by taking advantage of the higher rates available on close links.

It can be concluded that in the ever changing topology of ad hoc networks, *multi-rate* QoS enabled NDMR could be served as a better QoS approach to support real-time multimedia applications.

5 NDMR IN SPARSE NETWORKS

5.1 Introduction

In some situations, there are much fewer nodes in the transmission – for example, while there could be 50 nodes in a reasonable size area (1000m * 1000 m) for civilian communications, monitoring applications or medium scale military deployments. For example, a Vehicular Ad Hoc Network (VANET) can be fully connected during the rush hour in a day [75]. In a large conference room, a large number of desktops and laptops need to be connected to each other during the conference.

However, there may be other scenarios where the area is only sparsely populated – and in some cases this may be the very reason ad hoc is deployed since the nodes are so far apart, the only way to communicate outside the area is to relay through other nodes. For example, in a smart house, several sensors could be installed in a large house with a sparse density and they are required to communicate with each other [77]. VANET could be sparsely connected during the non-rush-hour time [75]. We also need communication in the desert area with the settlements are spread over a large scale and few populations [76].

When the node distribution is sparse, there may be problems when applying NMDR and QoS enabled NDMR as these protocols are based on the existing of several paths, but with fewer nodes it is not easy to find several paths between a source-destination pair any more. This is illustrated in Figure 5.1.

As shown in the figure, when there are enough nodes in the field, the distance between each node is within the transmission range and it is very easy to find multiple paths for a source-destination nodes and QoS

enabled NDMR is feasible and effective.

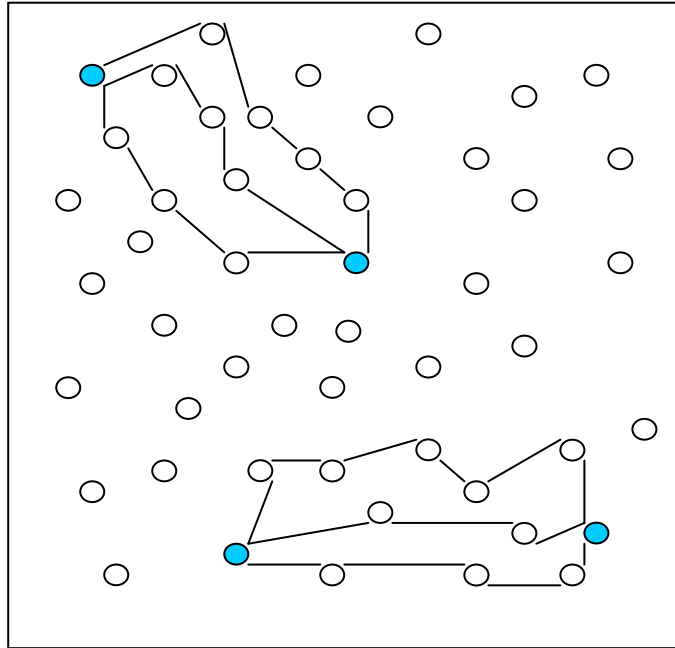


Figure 5.1 Sufficient nodes for NDMR

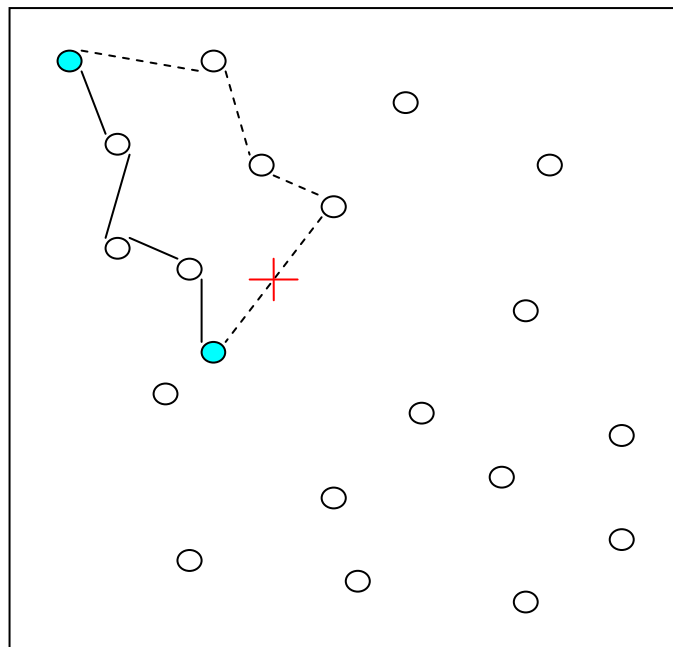


Figure 5.2 Sparse scenario

The situation is different with a sparse scenario. From Figure 5.2, we can see that with only 20 nodes in the same area, the density is much lower and nodes are not within each other's transmission range so often. For a source and destination pair, it is, in this example, impossible to find several paths for supporting NDMR and QoS NDMR so that the routing will revert to a single path, so losing the benefits of having multiple paths.

For instance, if a path breaks there is the time to establish a new one, although in a very sparse network it may actually be impossible to find another route on a path break and communication is lost. This is an inevitable consequence of having very sparse ad hoc networks and there is nothing that can address that extreme condition.

In this chapter, intermediate situations are considered where the density is low but alternative paths can be found, but not necessarily node-disjoint. Such a situation is illustrated in Figure 5.3.

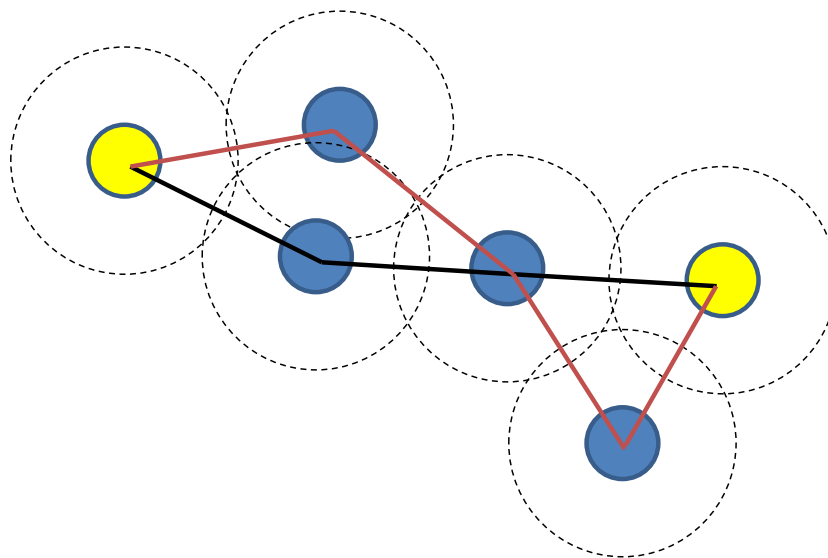


Figure 5.3 Sparse scenario

5.2 Limitations of NDMR

As discussed previously, NDMR and QoS enabled NDMR show better performance than other on-demand ad hoc routing protocols, but they have all been simulated in scenario with a dense distribution of nodes (50 nodes in a 1000m * 1000m field). In the different situation of a sparse network (less than 30 nodes in a 1000m * 1000m field), it may be that node-disjoint paths will not be available and the advantages of the protocol would be lost.

NDMR routing protocol is based on the concept that there is more than one route path (up to three paths in the implementation) between one source and destination pair. If one route path breaks, the source node could use the other path to continue transmitting data without initialising a new route discovery process so that the route discovery time is saved and a lower delay time than other routing protocols can be obtained.

With a sparse network, the possibility of finding a node-disjoint route is lower. For one source and destination pair, there may be only one route path and under such circumstances there are no back-up routes for that source node to continue transmitting data in the event of a path break. NDMR would, therefore, lose the advantage of multiple node disjoint routing protocols and it would have to initiate a route discovery process to find a new route in the event of a path break. This new route discovery would be likely to increase the overall delay.

This would be the same for NDMR and QoS enabled NDMR.

Figure 5.4 shows results from the sparse node situation compared with the dense-node network. The simulation scenarios are all the same except for the number of nodes in the network area. The results clearly show that with a lower number of nodes, the average delay time increases a lot,

the performance is no longer acceptable and NDMR loses its advantage. This increased delay is entirely down to the greater need to rebuild paths on a break as there are fewer back-up paths available.

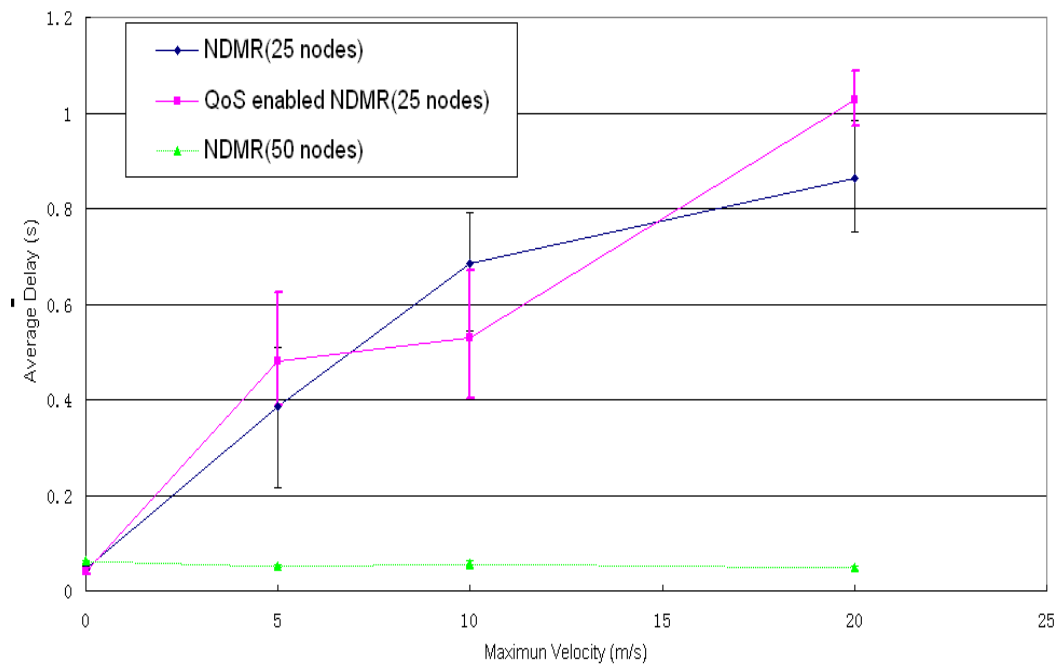


Figure 5.4 Average delay comparison in sparse and dense network

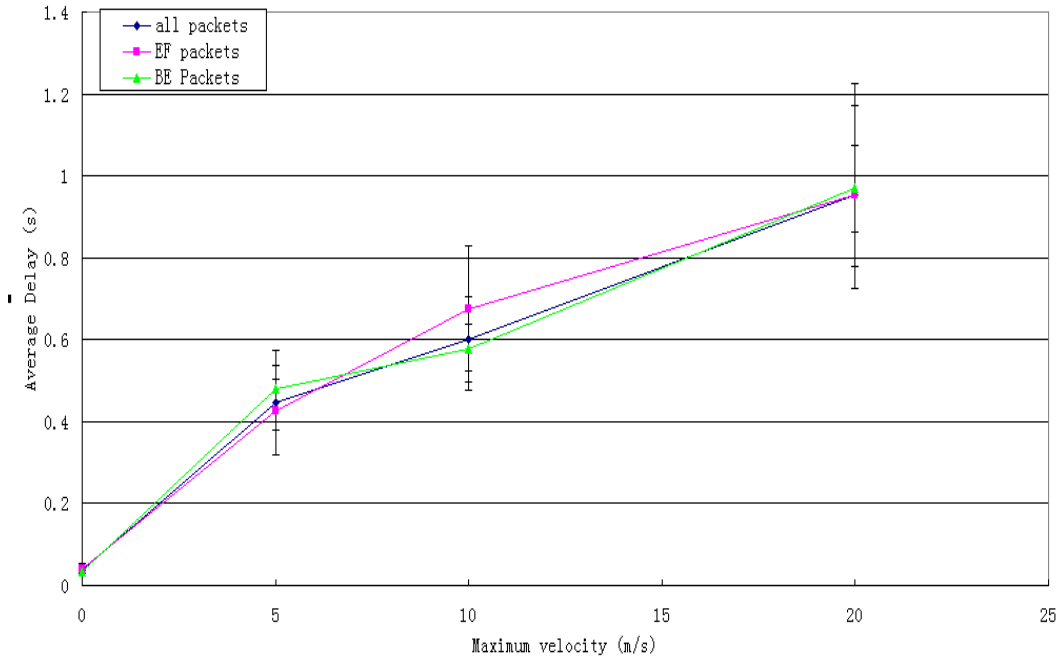


Figure 5.5 Average delay in sparse network – different priority

Figure 5.5 is the average delay time of EF and BE traffic for the sparse network. The priorities no longer work and delay times of the two priorities overlap: EF traffic does not have a lower delay time any longer because there is no better path to transmit higher priority packets in sparse node networks.

The delay time is also longer with the increase of mobile velocity because the higher the speed at which the node moves, the quicker it moves out of the transmission range of the connected node; with only a small number of nodes in the network it is harder to establish a route between a source and destination pair.

From the results above, the performance of NDMR gets worse with the decrease in number of nodes. In order to find out how many nodes in the network are acceptable for NDMR, a simulation with different numbers of nodes has been carried to assess the performance of NDMR.

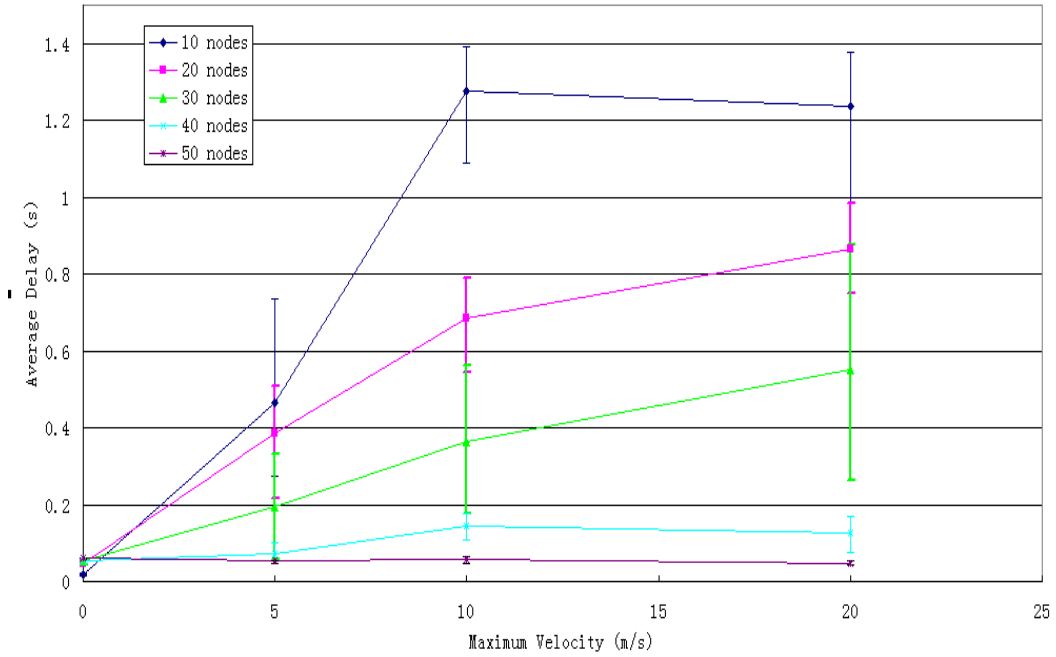


Figure 5.6 Average delay of different number of nodes

It can be seen from Figure 5.6, the average delay time improves dramatically as the number of nodes increases. When there are 40 or 50 nodes in the network, the performance is very good, but as the number of nodes decreases to less than 30, the performance is not acceptable anymore. It has been shown, therefore, that NDMR works very well only in the dense node network as it needs enough nodes to establish two or three node-disjoint routes that can be backups for each other to all data to be transmitted continuously.

It can be seen that the error bars overlap when the node velocity reaches 20m/s. That is because the faster the node moves, the quicker the network topology changes, and uncertainty of finding node-disjoint path is increased in the sparse node network. The error bars are also larger with the sparse node situation as there is more variability in finding paths.

However, despite this large variation in average delay time, it is clear that

as the number of nodes goes down then the performance deteriorates.

5.3 Modifications for NDMR in sparse node networks

Although NDMR will not work properly in sparse node network, it is a very good routing protocol with many advantages. In order to achieve better performance with small number of nodes in network for NDMR, modifications can be carried out to get round some of these limitations.

These modifications have been investigated in a series of experiments.

5.3.1 Loosening of route constraint

In the first experiment, the route constraint is loosened. A route can have one common node with other routes for a source-destination pair – so it is link-disjoint but not node disjoint. An example is shown in Figure 5.7. One common node *C1* is now allowed between the route paths of yellow source-destination pair.

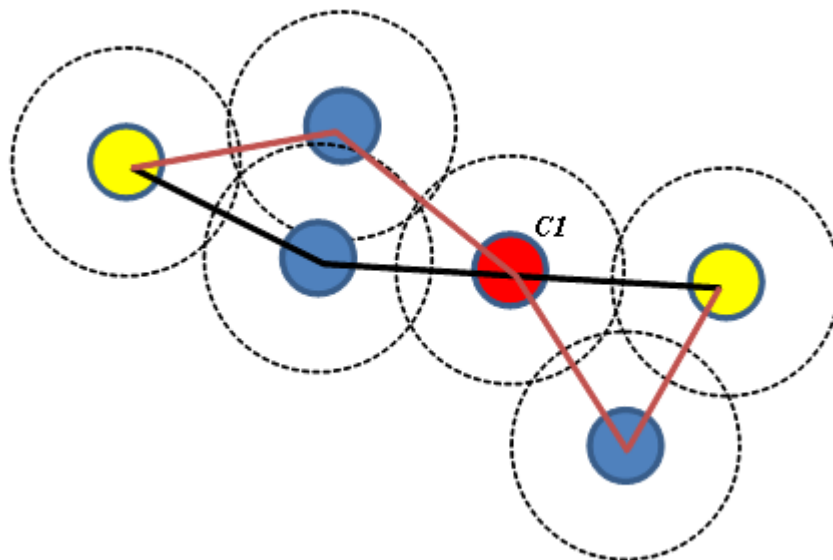


Figure 5.7 One common node

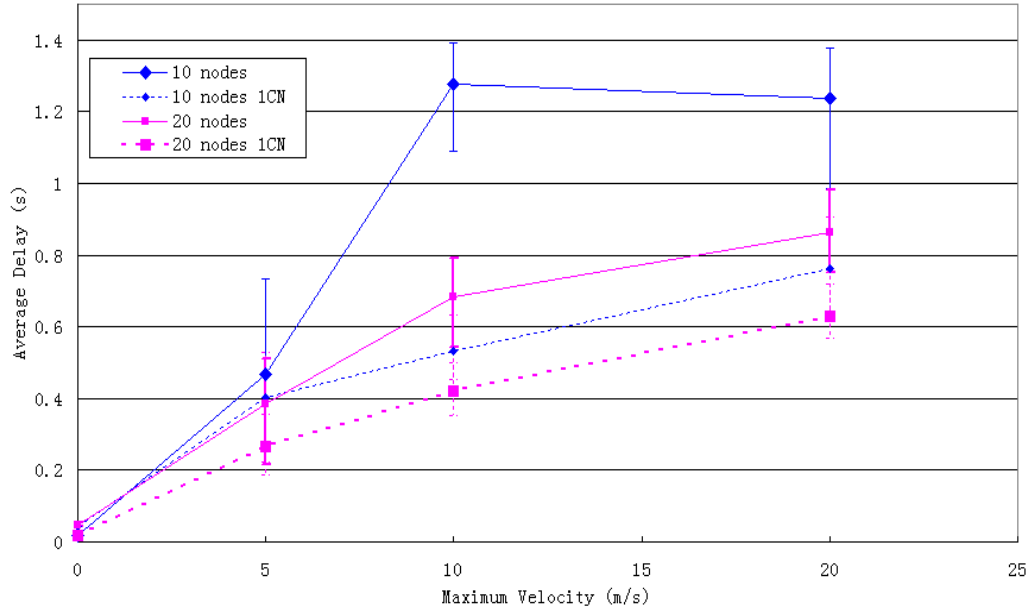


Figure 5.8 Average delay of one common node (CN) path

From Figure 5.8, when NDMR allows one common node in the sparse network, the average delay time (dotted line) decreases. The reason is that after allowing one common node, it is easier to find back-up routes among the small number of nodes to continue transmitting packets if the first route fails, so that saves the route discovery time and thus decreases the delay time dramatically.

The error bars overlaps as the node velocity increases. As discussed before, the uncertainty of the network topology increases as the node speed increases and this gets worse the smaller the number of nodes. The variation in delay still shows that with the decrease of number of nodes in the network, the performance is getting worse.

In order to clearly show the effect of one common node, the number of route breaks and route discovery time are also plotted in Figure 5.9 and Figure 5.10. They are both improved after allowing one common node in sparse node network.

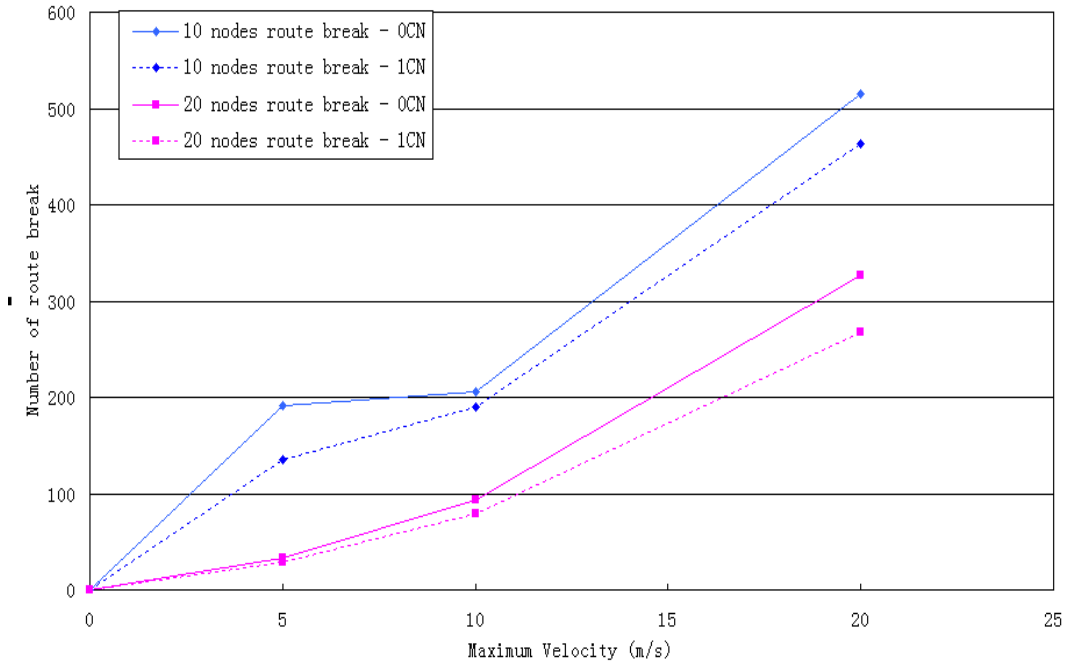


Figure 5.9 Number of route breaks in sparse scenario

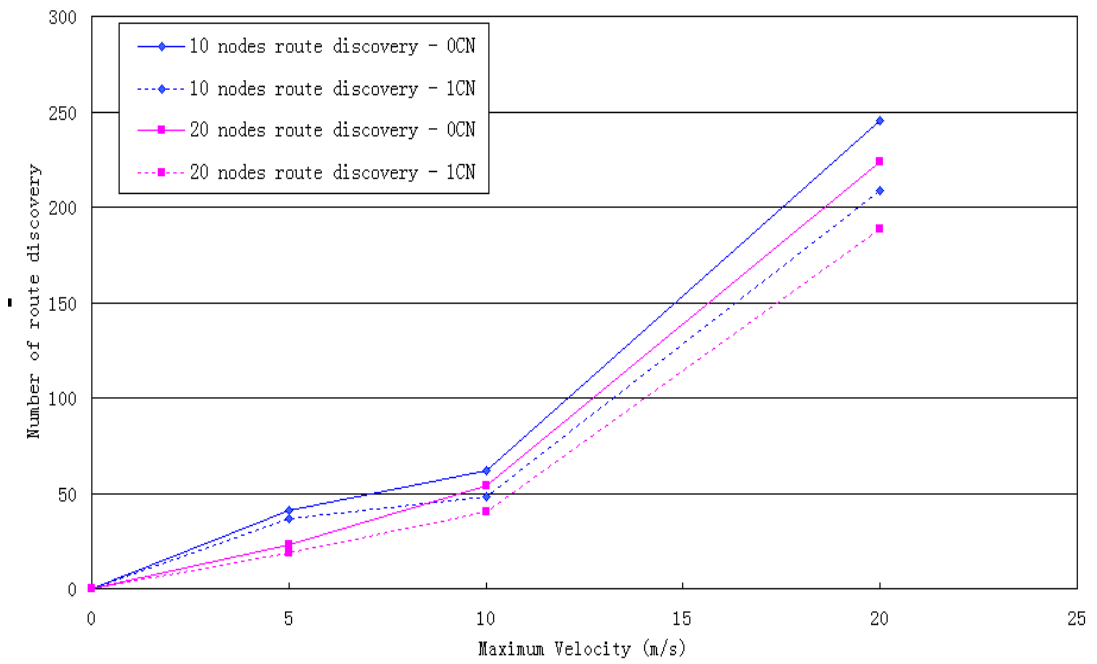


Figure 5.10 Number of route discovery times in sparse scenario

From the figures above, we can see that with one common node allowing the number of route breaks to decrease so the time penalty of route discovery due to route break is also decreased, hence reducing the

average transmission delay time in sparse node networks.

5.3.2 QoS enabled NDMR with one common node

In order to investigate the effect of having one common node, QoS enabled NDMR with one common node has been implemented. Figure 5.4 has shown that QoS enabled NDMR cannot achieve better performance than NDMR in sparse node network as no multipath exists. The results shown in Figure 5.11 also compare the average delay against velocity for QoS enabled NDMR in different density networks. It shows that with the density increasing in the network, QoS enabled NDMR gets better performance.

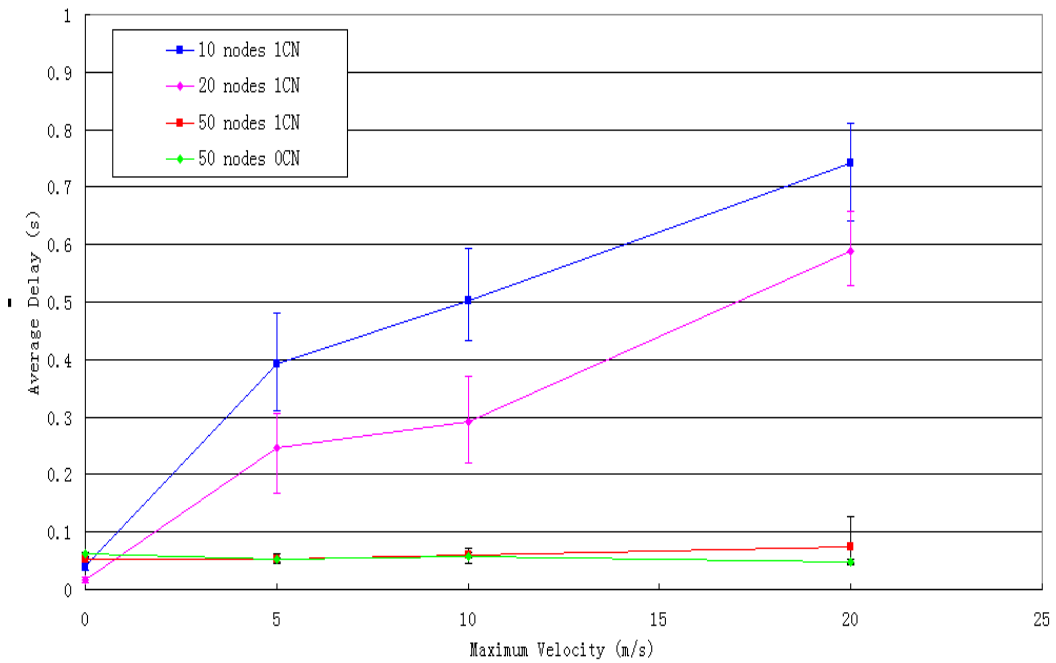


Figure 5.11 QoS enabled NDMR with one common node comparison

From Figure 5.11, the average delay time of QoS enabled NDMR with one common node in a sparse node network (10 and 20 nodes) also increases a lot. A comparison between QoS enabled NDMR and QoS enabled NDMR with one common node in a dense node network is also shown in

this figure.

It can be seen that in the dense node network, the routing protocol with one common node achieves better transmission delay time when the node are static but QoS enabled NDMR without a common node is better when the nodes moving fast (maximum speed is 20m/s).

The reason is that when the network is static, there is no link break happening and allowing one common node in the route path makes it is easier to find multiple *short* routes between source and destination nodes to reduce the transmission delay time.

When the nodes are moving faster, a break involving the common node will have a greater effect overall; the advantage of QoS enabled NDMR is that the back-up routes are always unaffected by a path break and so the advantages of NDMR will take effect and there will be a lower transmission delay time.

5.3.3 Two common nodes

In this experiment the constraint is loosened even further to allow two common nodes (Figure 5.12). Two common nodes *C1* and *C2* are now allowed between the route paths for the yellow source-destination pair. The simulation results are shown in Figure 5.13.

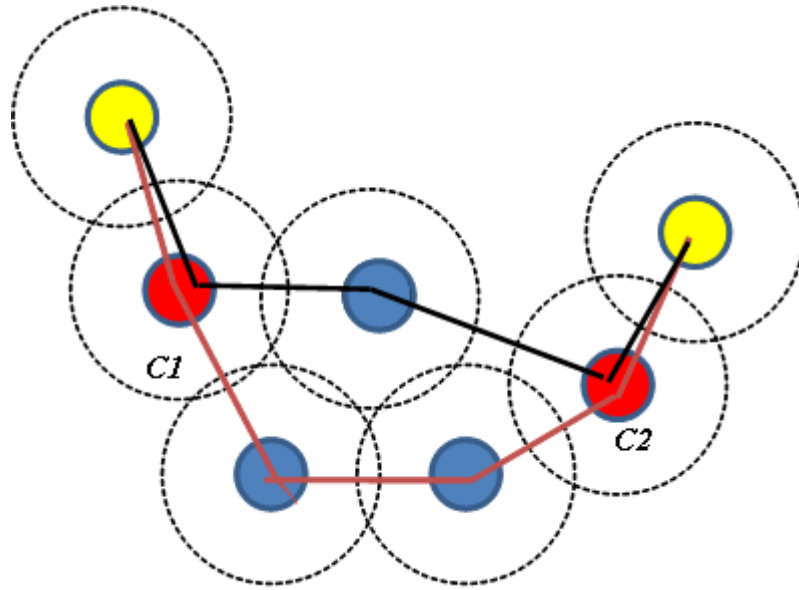


Figure 5.12 Two common nodes

After two common nodes are allowed, the delay time is reduced again compared with the case when only one common node is permitted: the source node has more back-up routes to choose and thus there is a higher probability that there is a back-up route to choose in the event of a path break.

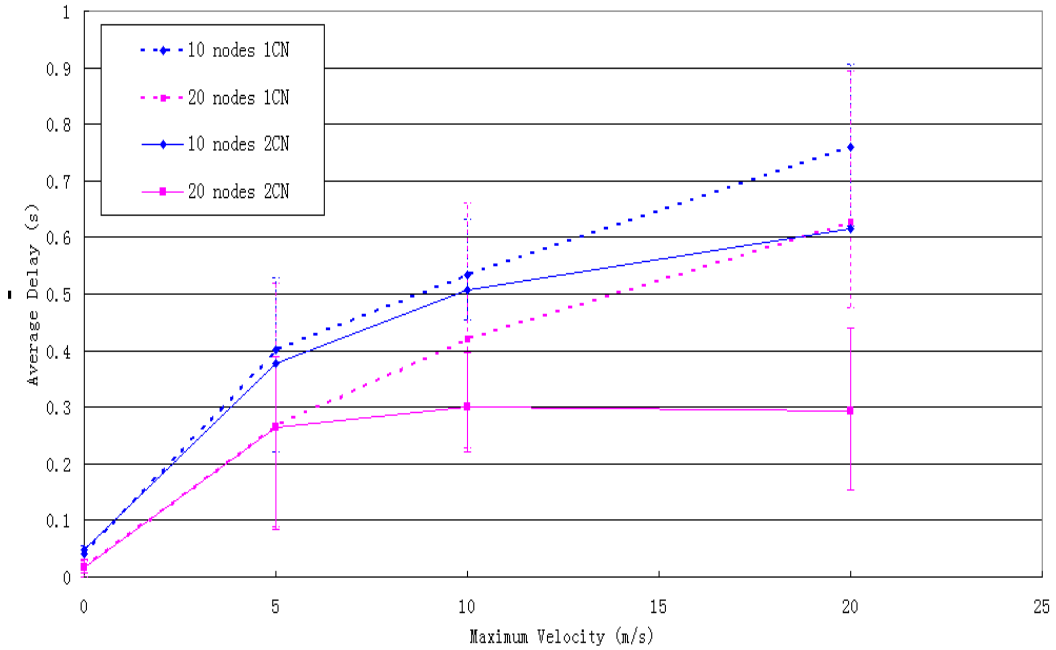


Figure 5.13 Average delay time of one and two CN path

The error bars overlaps very much this time. As two common nodes are allowed, the link breaks happen more often and with the change of network topology the delay time varies a lot so leading to more variation between runs.

We can see from the results of allowing common nodes, although the delay time is lower for sparse networks with one or two common nodes in the route path, it is still higher than the standard NDMR and QoS enabled NDMR in a dense node network.

It must be remembered though that allowing common nodes is losing the advantage of NDMR; without multiple node-disjoint routing paths, it is very easy to have link breaks on several paths due to only a single node failure: this violate the original intention of the NDMR routing protocol and means that one or more of the back-up routes is no longer available.

5.4 Summary

In this chapter, some of the problems with trying to use NDMR in sparse networks have been addressed. NDMR and QoS enabled NDMR are not suitable for low-density mobile ad hoc networks as there are not enough paths to set up several “node-disjoint” routes.

After loosening the restrictions in sparse node networks, a better performance can be obtained, but the problem of not having completely separate back-up paths becomes more acute as the node velocity increases.

6 CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

This thesis has presented an architecture for improving QoS in mobile ad hoc networks by modifying the Node-Disjoint Multipath Routing Protocol.

The introduction of a new type of packet in the network recording the queue length of each node along the paths set up gives the opportunity to allocate packets to paths in an optimum way to meet instantaneous constraints.

This approach allows the protocol to take advantage of the multi-rate capability of IEEE 802.11 and set up paths in such a way that the overall delay is calculated, not just the queue length at nodes. This could obviously be extended so any other measurable parameters could be included in the assessment of the performance.

While the results show considerable benefit, the investigation also considers the effect of sparse networks where the advantages of NDMR cannot be achieved because of the lack of node-disjoint routes.

6.2 Future Work

One aspect of this work that could be extended is to dynamically switch between node-disjoint, link-disjoint or common links and nodes in response to the instantaneous configuration in the network.

The control overhead to discover the options (for example minimum number of common nodes) would be greater and a study would have to

be performed to see if the overhead of this is outweighed by improved performance.

7 REFERENCES

- [1] Charles E. Perkins, Elizabeth M. Royer and Samir R. Das, Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks, *IEEE Personal Communications*, Feb 2001.
- [2] Charles E. Perkins, Elizabeth M. Belding-Royer, Samir R. Das, Ad Hoc On-Demand Distance Vector (AODV) Routing, <http://www.ietf.org/internetdrafts/draft-ietf-manet-aodv-13.txt>, IETF Internet draft, Feb 2003.
- [3] E. Marilly, O. Martinot, H. Papini, D. Goderis, "Service level agreements: a main challenge for next generation networks", Universal Multiservice Networks, 2002. 2nd European Conference on page(s): 297- 304.
- [4] G. Weiss, "Multiagent System: A Modern Approach to Distributed Artificial Intelligence", the MIT Press, Cambridge, Massachusetts, London, England, 1999.
- [5] J. Broch, D. Johnson, and D. Maltz, The Dynamic Source Protocol for Mobile Ad hoc Networks, <http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt>, IETF Internet draft , 19 July 2004.
- [6] L.G. Cuthbert, D. Ryan, L. Tokarchuk, J. Bigam and E. Bodanese, "Using intelligent agents to manage resource in 3G Networks", *Journal of IBTE*, 2001, vol. 2 part 4.
- [7] M. Luck, R. Ashri and M. d'Inverno, "Agent-Based Software Development", Artech House, Inc., 2004.

- [8] S. Floyd and V. Jacobson, Random Early Detection Gateways for Congestion Avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397-413, 1993.
- [9] S. Blake, "An Architecture for Differentiated Services", IETF RFC2475, December 1998.
- [10] S. Chakrabarti, A. Mishra, "QoS Issues in Ad hoc Wireless Networks", *IEEE Communications Magazine*, February 2001.
- [11] Soamsiri Chantaraskul, "An Intelligent-Agent Approach for Managing Congestion in W-CDMA Networks", Submitted for the degree of Doctor of Philosophy, August 2005.
- [12] Xuefei Li and Laurie Cuthbert, On-demand Node-Disjoint Multipath Routing in Wireless Ad hoc Networks, In *Proceedings of the 29th Annual IEEE Conference on Local Computer Networks, LCN 2004*, Tampa, Florida, U.S.A., November 16-18, 2004.
- [13] Xuefei Li and Laurie Cuthbert, Multipath QoS Routing of supporting DiffServ in Mobile Ad hoc Networks, In *Proceedings of the 6th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD2005)*, Towson, Maryland, May 23 ~ 25, 2005.
- [14] Xuefei Li, "Multipath Routing and QoS Provisioning in Mobile Ad hoc Networks", *PhD thesis, University of London*, January 2006.
- [15] Z.J. Haas and S. Tabrizi, "On Some Challenges, and Design Choices in Ad hoc Communications", *Proceedings of the IEEE Military Communications Conference (MILCOM)*, Bedford, MA, October 1998, pp.187-192.

- [16] T. Chen and M. Gerla, "Global state routing: A new routing scheme for ad hoc wireless networks", in *Proceedings of IEEE ICC'98*.
- [17] C. E. Perkins and P. Bhagwat, Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, In *Proceedings of ACM SIGCOMM*, pages 234-244, 1994.
- [18] S. Murthy and J.J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks", *ACM Mobile Networks and Applications Journal*, 1996.
- [19] Mahesh K. Marina, and Samir R. Das, On-demand Multipath Distance Vector Routing in Ad Hoc Networks. In *Proceedings of the 9th IEEE International Conference on Network Protocols (ICNP)*, November 2001.
- [20] A. Iwata, C.-C. Chiang, G. Pei, M. Gerla, and T.-W. Chen, "Scalable Routing Strategies for Ad Hoc Wireless Networks" *IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks*, Aug. 1999, pp.1369- 79.
- [21] Chai-Keong Toh, "A novel distributed routing protocol to support Ad hoc mobile computing", in *Proceedings of 1996 IEEE 15th Annual Int'l. Phoenix Conf. Comp. and Commun.*, Mar. 1996, pp. 480-486.
- [22] L. Wang, Y. Shu, M. Dong, L. Zhang and O. Yang, "Adaptive Multipath Source Routing in Ad hoc Networks", *IEEE ICC 2001*, Page: 867-871 vol.3, June 2001.

- [23] Ye, Z., Krishnamurthy, S. V., Tripathi, S.K., A Framework for Reliable Routing in Mobile Ad Hoc Networks. *IEEE INFOCOM* (2003).
- [24] L. Wang, Y. Shu, Z. Zhao, L. Zhang and O. Yang, "Load Balancing of Multipath Source Routing in Ad hoc Networks", in *Proceedings of IEEE ICC'02*, April 2002.
- [25] S. Blake, "An Architecture for Differentiated Services", IETF RFC2475, December 1998.
- [26] K. K. Ramakrishnan, Sally Floyd, and D. Black. The addition of explicit congestion notification (ECN) to IP. Request for Comments (Standards Track) 3168, Internet Engineering Task Force, September 2001.
- [27] IEEE Computer Society LAN MAN Standards Committee, *Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, IEEE standard 802.11, 1997. The Institute of Electrical and Electronics Engineers, New York, NY, 1997.
- [28] A. Chandra, V. Gummalla, and J. O. Limb, "Wireless Medium Access Control Protocols", in *IEEE communications Surveys*, 2000. <http://www.com-soc.org/pubs/surveys>. Second Quarter.
- [29] J. Broch, D.A. Maltz, D.B. Johnson, Y.C. Hu, and J. Jetcheva, A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols, In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking* (MobiCom'98), October 25-30, 1998, Dallas, Texas, USA.

- [30] Alcatel, "Service Level Agreement Solutions for Backbone Carriers", Technical paper in the *backbone network and business solutions series*, 2003.
- [31] J. Ferber, "Multi-Agent Systems an Introduction to Distributed Artificial Intelligence", Addison-Wesley, 1999.
- [32] M.J. Wooldridge, "An introduction to Multiagent Systems", *John Wiley & Sons*, 2002.
- [33] OPNET Technologies, online help, Inc. <http://www.opnet.com/>.
- [34] Soamsiri Chantaraskul, An Intelligent-Agent Approach for Managing Congestion in W-CDMA Networks, *PhD thesis, University of London*, August 2005.
- [35] S. Chakrabarti, A. Mishra, QoS Issues in Ad hoc Wireless Networks, *IEEE Communications Magazine*, February 2001.
- [36] M.Wooldridge & N.R. Jennings, Agent Theories, Architectures and Languages: a Survey in Wooldridge & Jennings eds. *Intelligent Agents*, Springer-Verlag, Berlin, 1995.
- [37] Alex Hayzelden, John Bigham, Heterogeneous Multi-Agent Architecture for ATM Virtual Path Network Resource Configuration, in *Intelligent Agents for Telecommunications Applications (IATA '98)*, LANAI 1437, S.Albayrak & F.J.Garijo (eds), Springer-Verlag, 1998, ISBN 3-540-64720-1.
- [38] L. Wang, Y. Shu, Z. Zhao, L. Zhang and O. Yang, "Load Balancing of Multipath Source Routing in Ad hoc Networks", in *Proceedings of IEEE ICC'02*, Page(s):3197 - 3201 vol.5 April 2002.

- [39] D. D. Perkins and H. D. Hughes, "A survey on quality-of-service routing in ad hoc networks", *IEEE Journal on Selected Areas in Communication*, vol.17, no.8, 1999.
- [40] D.chalmers and M. Sloman, "A survey of QoS in mobile computing environments", *IEEE Communications Surveys*, 1999.
- [41] Q. Xue and A. Ganz, "Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks", *Journal of Parallel and Distributed Computing*, (63), pp.154- 165, 2003.
- [42] H. Xiao, W. K. G. Seah, A. Lo, and K. C. Chua. "A Flexible Quality of Service Model for Mobile Ad hoc networks", In *Proceedings of IEEE Vehicular Technology Conference, Tokyo, Japan, May 2000*, pages 445-449.
- [43] IEEE 802.11 WG, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band", standard, IEEE, Sep 1999.
- [44] Z. Fan, "High throughput reactive routing in multi-rate ad hoc networks", *Electronics Letters*, 9th December 2004, vol. 40 no. 25.
- [45] Weilin Zeng, Haixia Tan, and Tatsuya Suda, "A Relay Based MAC Protocol to Support Multi-rate Feature in Mobile Ad Hoc Networks", In the *Proceedings of the Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous'05)*, 17-21 July 2005.

- [46] Gavin Holland, Nitin Vaidya, Paramvir Bahl, "A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks", In the proceedings of the *ACM/IEEE Int. Conf. on Mobile Computing and Networking (MOBICOM'01)*, Rome, Italy, July 2001.
- [47] Yongho Seok, Jaewoo Park and Yanghee Choi, "Multi-rate Aware Routing Protocol for Mobile Ad Hoc Networks", In the proceedings of *Vehicular Technology Conference, 2003. VTC, spring, 2003*. The 57th IEEE Semiannual Volume 3, 22-25 April 2003 Page(s):1749 - 1752 vol.3.
- [48] Beongku An, Dohyeon Kim and Innho Jee, "A Modeling Framework for Supporting QoS in Mobile Ad-hoc Networks", In the proceedings of *Vehicular Technology Conference, 2003. VTC, Spring, 2003*. The 57th IEEE Semiannual Volume 2, 22-25 April 2003 Page(s):935 - 939 vol.2.
- [49] Wei Liu and Yuguang Fang, "Courtesy Piggybacking: Supporting Differentiated Services in Multihop Mobile Ad Hoc Networks", *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, Volume 2, 7-11 March 2004 Page(s):1273 - 1283 vol.2.
- [50] Michael Gerharz, Christian Vogt, Christian de Waal, "Current Approaches towards Improved Quality-of-Service Provision in Mobile Ad-hoc Networks", Computer Science Department, Communications Systems, March, 2003.
- [51] M. Mirhakkak, N. Schult, D. Thomson, "Dynamic Quality-of-Service for Mobile Ad Hoc Networks", *Mobile and Ad Hoc Networking and Computing, 2000. MobiHOC. 2000 First Annual*

- [52] Mohammed Saghir Tat Chee Wan Rahmat Budiarto, "Multicast Routing with Quality of Service in Mobile Ad hoc Networks", in the proceedings of *National Computer Science Postgraduate Colloquium 2005 (NaCSPC'05)*, June 2005, USM.
- [53] Antonis Dimakis, Linhai He, John Musacchio, Hoi-Sheung Wilson So, Teresa Tung, Jean Walrand, "Adaptive Quality of Service for a Mobile Ad Hoc Network", *Department of Electrical Engineering and Computer Sciences*.
- [54] Venus S. Y. To, Brahim Bensaou and Sammy M. K. Chau, "Quality of Service Framework in MANETs Using Differentiated Services", in the proceedings of *Vehicular Technology Conference, 2003. VTC, Fall, 2003*. 2003 IEEE 58th, Volume 5, 6-9 Oct. 2003 Page(s):3463 - 3467 Vol.5.
- [55] R. Guerin and V. Peris, "Quality-of-Service in Packet Networks Basic Mechanisms and Directions", *Computer Networks*, vol. 31, pp. 169--189, Feb. 1999.
- [56] Marc R. Pearlman, Zygmunt J. Haas, Peter Sholander, and Siamak S. Tabrizi, "On the Impact of Alternate Path Routing for Load Balancing in Mobile ad hoc Networks." In *Proceedings of the IEEE/ACM International Symposium on Mobile Ad hoc Networking and computing*, Boston, Massachusetts, USA, August 2000, pp.3-10.
- [57] N. Nikaein and C. Bonnet, "A Glance at Quality of Services in Mobile Ad hoc Networks", in *Proceedings of DNAC 2002: glance 16th conference of New Architectures for Communications*, pp. 22-29, Paris 2002.

- [58] Y. Ge, T. Kunz, L. Lamont, "Quality of Service Routing in Ad hoc Networks Using OLSR", *HICSS 2003 presentation*, Hawaii, USA, January 2003.
- [59] S. Chen and K. Nahrstedt, "Distributed Quality-of-Service routing in Ad hoc Networks", *IEEE JSAC*, vol.17, no.8, August, 1999, pp.1488-1505.
- [60] T.-W. Chen, J. T. Tsai, and M. Gerla, "QoS Routing Performance in Multihop Multimedia Wireless Networks", In *Proceedings of IEEE International Conference on Universal Personal Communications (ICUPC)*, Volume 2, 12-16 Oct. 1997 Page(s):557 - 561 vol.2.
- [61] C. R. Lin and Chung-Ching Liu, "An On-demand QoS Routing Protocol for Mobile Ad hoc Networks", In *IEEE Global Telecommunications Conference (GLOBECOM)*, Volume 3, 27 Nov.-1 Dec. 2000 Page(s):1783 - 1787 vol.3.
- [62] S. Choi, J. del Pedro, N.S. Shankar, S. Mangold, "IEEE 802.11e contention based channel access (EDCF) performance evaluation", in *Proceedings of the IEEE ICC*, pp. 16-21, May 2003.
- [63] Sunil Kumar, Vineet S. Raghavan and Jing Deng, "Medium Access Control protocols for ad hoc wireless networks: a survey", Elsevier B.V., doi: 10.1016/j.adhoc.2004.10.001.
- [64] Na Yao, "A CBR Approach for Radiation Pattern Control in WCDMA Networks", PhD thesis from University of London, January 2007.
- [65] Yapeng Wang, "Intelligent Radio Resource Management for WLANs", PhD thesis from University of London, June 2007.

- [66] E. M. Royer and C. K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks", IEEE Personal Communications, vol.6 issue.2, pp.46-55, April 1999.
- [67] P. Mohapatra, J. Li and C. Gui, "QoS in mobile ad hoc networks". IEEE Wireless Communications Magazine, 10(3): pp.44-52, June, 2003.
- [68] Anna Charny, Jon Bennett, Kent Benson, Jean-Yves Le Boudec, Angela Chiu, William Courtney, Sahhram Davari, Victor Firoiu, Charles Kalmanek, and K. K. Ramakrishnan. Supplemental information for the new definition of the EF PHB. Request for Comments (Informational) 3247, Internet Engineering Task Force, March 2002.
- [69] Bruce Davie, Anna Charny, Jon Bennet, Kent Benson, Jean-Yves Le Boudec, William Courtney, Shahram Davari, Victor Firoiu, and Dimitrios Stiliadis. An expedited forwarding PHB. Request for Comments (Standards Track) 3246, Internet Engineering Task Force, March 2002.
- [70] A. Kamerman and L. Monteban. WaveLAN-II: A high-performance wireless LAN for the unlicensed band. Bell Labs Technical Journal, pages 118-133, summer 1997.
- [71] R. Ramanathan and M. Steenstrup. Hierarchically-organized, multihop mobile wireless networks for quality-of-service support. Mobile Networks and Applications, 3(1):101-119, June 1998.
- [72] Jun, J., et al."Theoretical maximum throughput of IEEE 802.11 and its applications". Proc. IEEE Int. Symp. On Network

Computing and Applications, Cambridge, MA, USA, 2003, pp. 249–256.

- [73] Olivier Dousse and Patrick Thiran, "Connectivity vs Capacity in Dense Ad Hoc Networks", IEEE INFOCOM 2004, School of Computer and Communication Sciences.
- [74] Philippe Jacquet, "Geometry of Information Propagation in Massively Dense Ad hoc Networks", *MobiHoc'04*, May 24–26, 2004, Roppongi, Japan.
- [75] Nawaporn Wisitpongphan, Fan Bai, Priyantha Mudalige, Varsha Sadekar, and Ozan Tonguz, "Routing in Sparse Vehicular Ad Hoc Wireless Networks", IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 25, NO. 8, OCTOBER 2007.
- [76] Abolhasan, M 2007, Sparse ad hoc networks for the desert, Desert Knowledge Cooperative Research Centre, Alice Springs.
- [77] Geoff West, Stewart Greenhill & Svetha Venkatesh, "A Probabilistic Approach to the Anxious Home for Activity Monitoring", Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05).
- [78] IEEE 802.11g-2003: Further Higher Data Rate Extension in the 2.4 GHz Band". IEEE (2003-10-20).
- [79] Enhanced Interior Gateway Routing Protocol, http://www.cisco.com/en/US/tech/tk365/technologies_white_paper09186a0080094cb7.shtml.
- [80] Z. Ye, S.V. Krishnamurthy, S.K. Tripathi, "A Framework for Reliable Routing in Mobile Ad hoc Networks," In Proceedings of

IEEE Conference on Computer and Communication (INFOCOM),
270-280, 2003.

- [81] A.M. Abbas, B.N. Jain, "Multiple Attempt Node-Disjoint Multipath Routing for Mobile Ad hoc Networks", Proceedings of 1st IEEE Conference on Communication Software and Middleware (COMSWARE), January 2006.
- [82] Christian Lazo Ramírez and Manuel Fernández Veiga, "QoS in Vehicular and Intelligent Transport Networks Using Multipath Routing", 1-4244-0755-9/07/\$20.00 '2007 IEEE, page 2556-2661.
- [83] Yunqian Ma, Yinzhe Yu, Guor-Huar Lu, and Zhi-Li Zhang, "Improving Wireless Link Delivery Ratio Classification with Packet SNR", *Electro Information Technology, 2005 IEEE International Conference*, 22-25 May 2005.
- [84] Yohei Hasegawa, Ichiro Yamaguchi, Takayuki Hama, Hideyuki Shimonishi and Tutomu Murase, "Improved Data Distribution for Multipath TCP Communication", *in the proceeding of IEEE Globecom 2005*.