

PERFORMANCE EVALUATION OF MANET ROUTING PROTOCOLS DSDV, TORA AND ENHANCED-TORA

Dissertation Submitted to the Central University of Punjab

For the award of

Master of Technology

In

Computer Science & Technology

By

Amandeep Kaur

Supervisor: Er. Meenakshi

Centre for Computer Science & Technology

School of Engineering & Technology

Central University of Punjab, Bathinda

August, 2014

DECLARATION

It is hereby declare that the dissertation entitled “**PERFORMANCE EVALUATION OF MANET ROUTING PROTOCOLS DSDV, TORA AND ENHANCED-TORA**” has been prepared by me under the expert guidance of Er. Meenakshi, Assistant Professor (Dissertation Coordinator) and Prof. Dr. A. K Jain (Administrative Supervisor), Centre for Computer Science & Technology, School of Engineering & Technology, Central University of Punjab. No part of this work has formed the basis for the award of any degree or fellowship previously.

Amandeep Kaur
Centre for Computer Science & Engineering,
School of Engineering & Technology,
Central University of Punjab,
Bathinda- 151001.

Date: August, 2014.

CERTIFICATE

It is hereby certified that AMANDEEP KAUR has prepared her dissertation entitled **“PERFORMANCE EVALUATION OF MANET ROUTING PROTOCOLS DSDV, TORA AND ENHANCED-TORA”**, for the award of M.Tech. Degree of the Central University of Punjab, under my guidance. She has carried out her work at the Centre for Computer Science and Technology, School of Engineering and Technology, Central University of Punjab, Bathinda.

Er. Meenakshi,
Assistant Professor,
Centre for Computer Science & Technology,
School of Engineering & Technology,
Central University of Punjab,
Bathinda- 151001

Date: August, 2014.

ABSTRACT

Performance Evaluation of MANET Routing Protocols DSDV, TORA and Enhanced-TORA

Name of student: Amandeep Kaur
Registration number: CUPB/MTECH/SET/CST/2012-13/01
Degree for which submitted: M.Tech- Computer Science & Technology
Name of supervisor: Er. Meenakshi
Name of centre: Computer Science & Technology
Name of school: School of Engineering & Technology
Key words: MANET, DSDV, TORA, Mobility Models, IMEP, FTP, TELNET, HTTP.

Wireless Networking has become the phrase du jour these days because of an attractive number of benefits it offers to the end users, by enabling them to access and share a wide pool of resources and information across the globe. Mobile ad hoc networks (MANETs) are a kind of infrastructure less wireless networks in which all the nodes act as peers and themselves configure the network.

Mobile ad hoc networks are an open area of research because of its applicability in a number of areas like tactical networks, emergency services and education. One of the major challenges in networking is the efficient, accurate, reliable, secure and immediate delivery of data from source to destination. Therefore, the efficient routing of data across the mobile ad hoc network is a major concern of researchers all over the world. Several routing protocols have been developed to send data efficiently across mobile ad hoc networks. These protocols have been divided into three categories- Proactive, Reactive and Hybrid. The performance evaluation of these protocols has been going on since a long time. Most of the current research focuses on the pause times, simulation times and network size to measure the performance of various mobile ad hoc network routing protocols. A very few work has been done on the performance analysis of protocols by varying the underlying mobility models and traffic patterns.

This dissertation work is focused on the improvement of the poor performance of TORA with the rise in number of traffic connections. This behavior has been credited to the link sensing mechanism of Internet MANET Encapsulation protocol (IMEP). IMEP is a multi-purpose network-layer protocol that supports the operation of many routing protocols including TORA. It provides mechanism for sensing the status of links in TORA. This link sensing mechanism has been investigated and proposed modification by K. H. Lim and A. Datta has been implemented to observe improvement in the overall performance of Enhanced-TORA protocol in comparison with original TORA.

Through this research, investigations have been made into the behavior of DSDV, and TORA mobile ad hoc routing protocols by varying the underlying mobility models (Random Waypoint, Random Direction and Manhattan Grid model) and Traffic patterns (FTP, TELNET, HTTP). The metrics used to analyze the performance are Throughput, Average End to End delay, Routing overhead, Packet Delivery Ratio and Packet Loss. This research draws an analysis whether under which mobility model and traffic pattern combination these protocols give the best performance. Results show that DSDV gives best performance under Manhattan Grid Mobility model and FTP traffic pattern. Whereas, TORA and Enhanced-TORA give best performance under Random Direction Mobility model and FTP traffic. The performance of Enhanced-TORA is then compared with original TORA using 10 traffic connections under Random Direction mobility model and FTP Traffic and it was observed that Enhanced-TORA outperformed original TORA thereby eliminating the problem of performance degradation with rise in number of connections.

(Amandeep Kaur)

(Supervisor- Er. Meenakshi)

ACKNOWLEDGEMENTS

Foremost, I would like to articulate my heartfelt gratitude to all those who have given me an opportunity to undertake research work in the versatile field of Mobile Ad-Hoc Networking. I am particularly thankful to my guide Er. Meenakshi for bestowing her precious time reviewing my practical as well as documentation work. She provided excellent and timely feedback, support and suggestions that alleviated me to conduct my research in a well-organized manner. She always encouraged critical thinking and guided me attentively throughout the course of the work. Above all, she helped me improve my presentation skills and taught me the accurate way to utilise the available resources optimally, which I heartily appreciate. I have gained a lot of knowledge through this research.

I would also like to extend my sincere and extreme gratefulness to Prof. Dr. A.K Jain, Coordinator of Centre of Computer Science & Technology, the staff of Computer Centre and Centre for Computer Science & Technology for providing an exceptionally excellent academic environment.

This acknowledgement would be deficient if I skip mentioning the emotional support, love, blessings and inspiration provided by my family. I would also like to appreciate my friends Kamalpreet Singh Dhillon, Sapna Thakur and Tanya Garg for their support and critical peer reviewing of my dissertation report. Finally I bow to the almighty, who gave me strength to accomplish this work with honesty, authenticity, sincerity and dedication.

(Amandeep Kaur)

TABLE OF CONTENTS

Sr. No.	Content	Page No.
1.	INTRODUCTION	1-7
1.1.	Mobile Ad Hoc Networks	1
1.2.	Characteristics	2
1.3.	Applications	2
1.4	Challenges	3
1.5	Mobile Ad-Hoc Network Routing Protocols	4
1.6.	Problem Statement	5
1.7.	Objectives of Research	6
1.8.	Organization of the Dissertation	6
2.	REVIEW OF LITERATURE	8-13
3.	DSDV, TORA AND IMEP PROTOCOLS	14-23
3.1.	DSDV Mobile Ad-Hoc Routing Protocol (Proactive Protocol)	14
3.2.	TORA -Temporally Ordered Routing Algorithm (Hybrid Protocol)	17
3.3.	Internet MANET Encapsulation Protocol	21
3.4.	Chapter Summary	23
4.	TOOLS USED	24-31
4.1.	Simulation Tool Used: NS2 (ns-allinone-2.35 Simulation package)	24
4.2.	BonnMotion Tool to Generate Movement Scenarios using Various Mobility Models	29
4.3.	Chapter Summary	31
5.	PROPOSED WORK AND SIMULATION ENVIRONMENT	32-40
5.1.	Problems in original TORA Protocol	32
5.2.	Proposed Technique to Enhance the Performance of TORA	34
5.3.	Simulation Environment for Performance Evaluation of DSDV, TORA and Enhanced-TORA	35

5.4.	Performance Metrics for Evaluation of Protocols	39
5.5.	Chapter Summary	40
6.	RESULTS AND DISCUSSIONS	41-51
6.1.	Performance Evaluation of DSDV and TORA Protocol Under Different Mobility Models and Traffic Patterns	41
6.2.	Overall Results and Discussions for Evaluation of TORA and DSDV under Different Mobility Models And Traffic Patterns	45
6.3	Investigation Into Poor Performance of Original TORA with rise in Number of Connections	46
6.4.	Performance Evaluation of Enhanced TORA under different Mobility models and FTP Traffic	47
6.5.	Performance Comparison of Enhanced-TORA with Original TORA with Respect to Number of Connections	49
6.6.	Chapter Summary	51
7.	CONCLUSIONS AND FUTURE WORK	52
	REFERENCES	53-59

LIST OF TABLES

Table No.	Table Description	Page No.
1.1	Challenges in Mobile Ad Hoc Networks	3
5.1.	IMEP constant values for Original TORA	34
5.2.	Modified IMEP constant values for Enhanced TORA	34
5.3.	Parameters used for simulation of DSDV and TORA	36
6.1.	Best performing mobility models under different traffic patterns for DSDV and TORA	46

LIST OF FIGURES

Figure No.	Figure Description	Page No.
1.1.	Classification of Wireless networks	1
1.2.	Classification of Mobile Ad-Hoc Routing Protocols	4
3.1.	Transmission of a packet from source node 1 to destination node 7.	14
3.2.	Routing table management using UPDATE packet	15
3.3.	Smaller sequence number in update packet	16
3.4.	Equal sequence number in Update packet	16
3.5.	Greater Sequence number in Update packet	17
3.6.	Propagation of QRY message through the network (Route Creation)	18
3.7.	UPD reply from Destination and formation of DAG (Directed Acyclic Graph)	19
3.8.	Route Re-establishment using Link Reversal (Route Maintenance)	20
3.9.	Route Erasure in TORA using CLR packet	21
3.10.	IMEP running at Network layer	21
3.11.	Detection of "UP" link	23
3.12.	Detection of "DOWN" link	23
4.1.	Basic architecture of NS2	25
4.2.	Role of AWK Scripts	25
4.3.	Running dsdv.tcl Script	28
4.4.	Using AWK script to calculate performance metrics	29
5.1.	Movements using Random Waypoint Model	38
5.2.	Movements using Random Direction Model	38
5.3.	Map used in Manhattan Grid Mobility model	39
6.1.	Throughput of DSDV using FTP Traffic	42
6.2.	Average End to End Delay of DSDV using FTP Traffic	42

6.3.	Routing Overhead of DSDV using FTP Traffic	42
6.4.	Packet Drop of DSDV using FTP Traffic	43
6.5.	Packet Delivery Ratio of DSDV using FTP Traffic	43
6.6.	Average End to End Delay of TORA using FTP Traffic	44
6.7.	Packet Delivery Ratio of TORA using FTP Traffic	44
6.8.	Routing Overhead of TORA using FTP Traffic	45
6.9.	Packet Delivery Ratio of Original TORA protocol	46
6.10.	Routing Overhead of Original TORA protocol	47
6.11.	Average End to End Delay of Original TORA protocol	47
6.12.	Packet Delivery Ratio of Enhanced TORA under different mobility models	48
6.13.	Average End to End Delay of Enhanced TORA under different mobility models	48
6.14.	Routing Overhead of Enhanced TORA under different mobility models	49
6.15.	Packet Delivery Ratio Comparison of Original and Enhanced-TORA	50
6.16.	Routing Overhead Comparison of Original and Enhanced-TORA	50
6.17.	Average End to End Delay Comparison of Original and Enhanced-TORA	50

LIST OF APPENDICES

Sr. No.	Description of Appendix	Page No.
A.	Simulation of DSDV, TORA and Enhanced TORA	60-61
B.	Traffic Sources	62-63
C.	Scenarios for Different Mobility Models	64-65
D.	AWK scripts	66-67
E.	Graphical Results for DSDV and TORA under TELNET and HTTP Traffic	68-74

LIST OF ABBREVIATIONS

Sr. No.	Full Form	Abbreviation
1.	Acknowledgement	ACK
2.	Ad Hoc On Demand Distance Vector	AODV
3.	Carrier Sense Multiple Access	CSMA
4.	Clear	CLR
5.	Constant Bit Rate	CBR
6.	Destination Sequenced Distance Vector	DSDV
7.	Directed Acyclic Graph	DAG
8.	Distributed Coordination Function	DCF
9.	Dynamic Source Routing	DSR
10.	File Transfer Protocol	FTP
11.	Fisheye State Routing	FSR
12.	Floor acquisition multiple Access	FAMA
13.	Hyper Text Transfer Protocol	HTTP
14.	Impact of Mobility Patterns on Routing in Ad-hoc Network	IMPORTANT
15.	Institute of Electrical and Electronics Engineers	IEEE
16.	Interface Queue Type	IFQ
17.	Internet MANET Encapsulation Protocol	IMEP
18.	Medium Access Control	MAC
19.	Mobile Ad Hoc Networks	MANET
20.	Multiple Access with Collision Avoidance	MACA
21.	Network Animation	NAM
22.	Network Simulator-2	NS-2
23.	Object Block Message	OBM
24.	Object Oriented Tcl	OTcl
25.	Optimized Link State Routing	OLSR
26.	Packet Delivery Function	PDF
27.	Packet Delivery Ratio	PDR
28.	Peer to Peer	P2P

29.	Personal Digital Assistant	PDA
30.	Query	QRY
31.	Reference Point Group Mobility model	RPGM
32.	Routing Information Protocol	RIP
33.	Secure Efficient Distance Vector	SEAD
34.	Temporally Ordered Routing Algorithm	TORA
35.	Terminal Network	TELNET
36.	Tool Command Language	Tcl
37.	Transmission Control Protocol	TCP
38.	Update	UPD
39.	Upper Layer Protocol	ULP
40.	Virtual Internet Testbed	VINT
41.	Wireless Routing Protocol	WRP
42.	Zone Routing Protocol	ZRP

CHAPTER 1

INTRODUCTION

1.1. Mobile Ad Hoc Networks

Wireless networks have become an epitome of revolution in communication industry because of a number of benefits it offers to the end users like convenience, mobility, ease of deployment, scalability and cost effectiveness (Bang & Ramteke, 2013). These networks can be Infrastructure based or Infrastructure less as shown in Figure 1.1 (Loo et al, 2012).

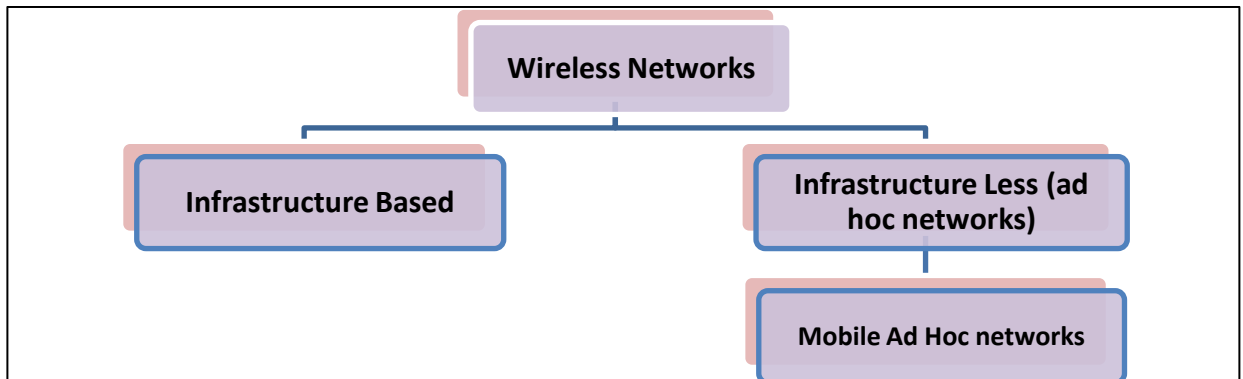


Figure 1.1. Classification of Wireless networks

1.1.1. Infrastructure-based Wireless Networks:

The nodes in infrastructure based wireless networks rely on a centralized organized point usually an Access point to communicate with each other (Dhenakaran & Parvathavarthini, 2013).

1.1.2. Infrastructure-less Wireless Networks (Ad hoc Networks):

The phrase “Ad hoc networks” has become the raison d'être of communication in geographically remote and emergency areas in form of mobile, underwater and flying ad hoc networks. Mobile ad hoc networks (MANETs) are a kind of infrastructure less wireless networks that do not rely on an organized central point and is a sovereign collection of mobile users that communicate with each other as peers in a dynamically changing network topology. This kind of network is decentralized which means that the nodes discover the topology to execute the delivery of messages themselves, which implies that the routing functionality is inbuilt or incorporated inside the mobile

nodes (Dhenakaran & Parvathavarthini, 2013). Therefore there is a strong need of efficient routing protocols to carry out the communication between the mobile nodes inside MANET in reliable, efficient and confidential manner.

1.2. Characteristics:

The various characteristics (Hoebeke et al.) of Mobile ad Hoc network have been discussed below:

- **Autonomy and Infrastructureless:** There is no centrally deployed entity to coordinate or authorize the communication between the various mobile devices.
- **Multi-hop routing:** The packets sent by a source node may travel through a number of intermediate nodes on its journey towards the intended destination node.
- **Dynamic network topology:** The topology of the network is not static i.e. the nodes keep on changing their locations thereby leading to a dynamically changing topology.
- **Heterogenous devices:** There may be devices having different functionalities communicating with each other. For example, a mobile phone, a PDA or a laptop.
- **Scalability:** The nodes can move away and join some other network at any point of time. Similarly, the addition of new nodes into the network is also possible at any time.
- **Self creation, self organization, self administration:** The network can be created at any time by the nodes themselves and is organized and administered by the nodes only.

1.3. Applications:

Mobile ad hoc networks are applicable in a number of areas (Sarkar et al, 2007) (Hoebeke et al.) as described ahead.

- **Tactical Networks:** Various military combat operations in which military personnel's need secure ad hoc communication and automated battlefields.

- **Emergency Services:** Various Rescue operations in disaster prone areas, hospitals for better services in situations of environmental tragedies, police and fire fighting operations.
- **Education:** Virtual classrooms, online tutorials & lectures, worldwide conferences and meetings.
- **Commercial and Civilian Situations:** Ecommerce, business applications, vehicular services, airports, shopping centers, sports stadiums.
- **Sensor Networks:** Smart homes, data tracking of animal movements, chemical and biological monitoring.

1.4. Challenges:

Mobile ad hoc networks face some challenges which need to be addressed as listed in Table 1.1.

Table 1.1. Challenges in Mobile Ad Hoc Networks (Cordeiro & Agarwal)

Challenge	Explanation
Energy Constrained Operation	The nodes operate on batteries; therefore energy conservation is an important system design optimization criterion for these nodes.
Bandwidth Constraints	The throughput of wireless links is usually lesser than wired links. Therefore, the efficiency of links need to be improved by limiting the effects of noise, interference, multiple access etc.
Security	These networks are more prone to security threats like eavesdropping, spoofing, denial of service attacks etc.
Efficient Routing capabilities	There is a need of efficient routing protocols to address the routing and security concerns of mobile ad hoc networks.

1.5. Mobile Ad-Hoc Network Routing Protocols

A Routing protocol is basically a set of rules which administer the transmission of packets from one node to another. To send packets containing useful information from source to destination node in MANETs, a number of routing protocols have been proposed and implemented which are applied according to their suitability in a particular scenario. The protocols can be classified into three categories: Proactive Reactive and Hybrid (Dhenakaran & Parvathavarthini, 2013).

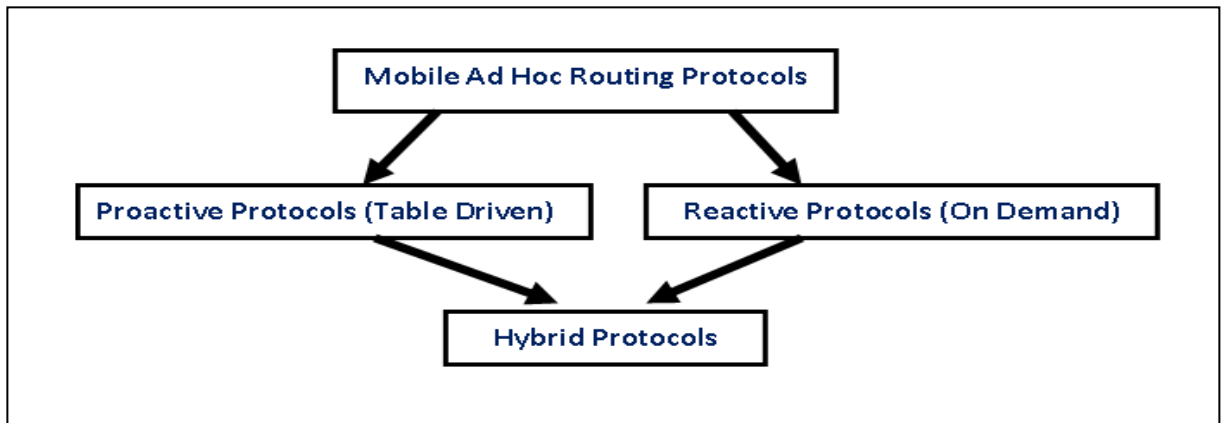


Figure 1.2. Classification of Mobile Ad-Hoc Routing Protocols

1.5.1. Proactive (Table-driven) Protocols: These (Ahmed et. al, 2012) are those protocols in which every node maintains one or more routing tables which contain information about the entire network topology. In these protocols, the transmission of packets is usually faster but exhibit greater routing overhead because all the routes have to be defined before sending the packets. Proactive protocols have lower latency because routes to all nodes are maintained at all the times.

Example: DSDV, OLSR (Optimized Link State Routing).

1.5.2. Reactive (On-Demand) Protocols: These (Ahmed et. al, 2012) are those protocols in which routes are searched only when needed. A route discovery process is initiated which is terminated in case the route has been found or the route is not available. Route maintenance is an important operation of these protocols. Reactive techniques have smaller routing overheads but higher latency.

Example: DSR, AODV.

1.5.3. Hybrid Protocols: Those (Gupta & Kaushik, 2012) protocols which can be operated in both proactive and reactive modes are termed as Hybrid protocols. These behave like proactive protocols at first as the nodes look up the routes from their routing tables. But when a node does not find a route to a particular destination, a route discovery process is started thereby transitioning the protocol from proactive to reactive mode.

Example: TORA, ZRP.

1.5.4. Protocols Chosen for Research Work: DSDV and TORA

This work is focused on a proactive protocol- Destination Sequenced Distance Vector (DSDV) and a hybrid protocol- Temporally Ordered Routing Algorithm (TORA).

- DSDV protocol was chosen for this study because it is a traditional proactive routing protocol and understanding the behavior of this protocol paves way towards extensive analysis of a number of other protocols which are based on its concepts like AODV, SEAD (Secure Efficient Distance Vector) (Perkins & Bhagwat, 1994).
- TORA protocol was chosen because in many studies it has been mentioned as a complex protocol to be implemented because of its looping problem and bugs in NS-2 simulator (Patil et al, 2009). So taking up the challenge to simulate its behavior and implementing the proposed modification is one of the objectives of this research.

1.6. Problem statement

- In the currently available research, the choice of mobility model and traffic patterns for the simulation study of routing protocols has not been considered much. Major focus had been on the pause times, simulation times and network size. In the proposed work, the ad hoc routing protocols DSDV and TORA will be evaluated under various different mobility models and traffic patterns to find out the best performing mobility model and traffic pattern combination.
- TORA faces problems like increased routing overhead, degrading packet delivery ratio and more delay with rise in number of connections. This problem has been credited to link sensing mechanism of IMEP protocol, on which TORA relies for its various functions. Therefore, this mechanism needs to be investigated to enhance

TORA's performance. So in this research, improvement in TORA protocol will be done by the modifying the link status sensing mechanism of IMEP protocol by implementing the proposed modification.

1.7. Objectives of Research

The main objective of this research is the performance enhancement of TORA with rise in number of traffic connections. The overall objectives of this research have been listed below:

- (i) To set up the simulation environment.
- (ii) Then evaluate and graphically analyze the performance of DSDV and TORA under different mobility models (Random Waypoint, Random Direction, Manhattan Grid) and traffic patterns (FTP, TELNET, HTTP) to analyze whether which combination of mobility model and traffic pattern gives best performance for these protocols.
- (iii) Analyzing the behavior of original TORA protocol with rise in number of traffic connections.
- (iv) To implement the proposed modification of link sensing mechanism of IMEP to enhance the performance of TORA protocol with rise in number of connections.
- (v) Performance evaluation of Enhanced-TORA under different mobility models to find out the best performing mobility model.
- (vi) Then compare the performance of Enhanced-TORA protocol with original TORA under the best performing mobility model.

1.8. Organization of the Dissertation

This dissertation has been organized into 7 chapters. Chapter 1 introduces the research area MANET, its characteristics, applications and challenges. A walkthrough into the need of routing protocols and their classification is also given. Further the reason behind the protocols chosen for this research is discussed followed by the problem statement and objectives of research.

In Chapter 2, extensive review of available literature on the selected problem has been given. Chapter 3 details the functioning of DSDV and TORA MANET routing

protocols to understand their routing behavior. IMEP protocol, on which TORA relies for link detection, is also discussed in detail.

Chapter 4 discusses the tools used, their installation and use in this research work.

In Chapter 5, the proposed work and simulation environment has been detailed. In this chapter, the modification in IMEP and various simulation parameters have been described. In Chapter 6, the results obtained have been analyzed graphically and discussions have been made for each simulation scenario leading to a detailed conclusion and future work in Chapter 7.

CHAPTER 2

REVIEW OF LITERATURE

Although, no study seems to have been undertaken that specifically focuses on the variations of mobility models and traffic patterns to evaluate the performance of DSDV, TORA and Enhanced-TORA, but a number of studies have been carried out that relate with the problem. Some of them relate to the modification of TORA protocol but they have not evaluated the performance of their enhanced version of TORA under different mobility models as done in this research work.

Royer et al, 1999 have described the operation of various routing protocols and the comparison of their characteristics in this review paper. The numerous applications of Ad hoc networks and the challenges faced by these networks have been discussed. It has been concluded that no particular protocol is best suited for all the scenarios. Every protocol has certain advantages and disadvantages. They have further highlighted that various routing protocols are suitable for certain situations.

Royer et al, 2000 have described various MAC protocols and performed investigation over what could be the effect of various MAC protocols over the performance of ad hoc network protocols. They have described the MAC protocols viz FAMA, MACA, CSMA and IEEE 802.11 DCF. Conclusions show that the performance of the table driven ad hoc routing protocols remains approximately the same by varying the underlying MAC protocols.

Camp et al, 2002 surveyed various mobility models with a goal of providing detailed descriptions of these models to researchers so that they can choose models for their simulation studies. The drastic changes in the performance of various ad hoc routing protocols with the change in the underlying mobility models have been illustrated. The authors have suggested that the similarities and differences between mobility models that randomly select directions and specific locations should be analyzed in future.

Chlamtac et al, 2003 have summarised the characteristics, capabilities, applications and design constraints of MANETs. The growing interest in research and

development of MANET has been discussed. The need for simulation and performance evaluation of protocols and the use of mobility models, network simulators to analyze the behaviour of MANETs have been described by the authors in a comprehensive manner. The research directions in the field of routing, security and Quality of service in MANETs have been given.

Bai & Helmy, 2004 have performed a survey on various mobility models. They have focussed on the fact that the mobility models play an important role in determining the performance of protocols in Mobile ad hoc networks, So it is necessary to study them and evaluate their effect on the performance of mobile ad hoc routing protocols. The unique mobility characteristics of mobility models were analysed, which were found to be different for different mobility model. So the models are expected to behave differently and affect the performance of various protocols differently. According to their conclusion, researchers are advised to use rich set of mobility models to carry out comprehensive performance analysis.

Bettstetter & Hartenstein, 2004 have thrown some light on the fundamental stochastic properties of the Random Waypoint mobility model. They have provided an in-depth understanding of the behaviour of this model in simulation based analysis of mobile ad hoc networks. They have encouraged that the study of stochastic properties of various mobility models is quite important to understand the functioning of the model properly.

Pal et al, 2011 have tried to investigate the behavior of various mobile Ad-hoc network routing protocols under Exponential and Pareto traffic sources for nodes using mobility model Reference Point Group Mobility model (RPGM). They studied the effect of traffic patterns with Reference Point Group Mobility (RPGM) model on mobile ad hoc network routing by making use of performance metrics Normalized routing load, Packet Delivery Fraction and throughput. In this study only reactive routing protocols were taken under consideration and study of proactive routing protocols in the assumed scenarios has been left for future. They also suggested analysing the performance of protocols by considering multimedia application using Exponential and Pareto traffic sources.

Kumari et al, 2011 have analyzed the performance of some mobile ad hoc routing protocols by varying load on the network, mobility of nodes and type of traffic patterns used (CBR and TCP). The mobility model used for generation of movements was Freeway Mobility model. Scenarios for this model has been generated by IMPORTANT (Impact of Mobility Patterns on Routing in Ad-hoc NeTwork) tool. A detailed simulation analysis was carried out in NS2. The performance metrics used for comprehensive performance analysis of the protocols are Packet Delivery Fraction, Average end-to-end Delay, Packet Loss, Routing Overhead, Normalized Routing Load and throughput.

Radwan et al, 2011 have analysed and compared the performance of some mobile ad hoc networking protocols using different simulation areas. Parameters used for this study are Routing message overhead, Throughput and End to End delay. They presented an idea of analysing the performance of routing protocols in future using different mobility models, MAC protocols and Traffic patterns to yield variations in results.

Al-Maashri & Ould-Khaoua, 2006 have investigated the performance of various mobile ad hoc networks based on various traffic models viz CBR, pareto and exponential with respect to performance parameters like delivery ratio, routing overhead, throughput and end-to-end delay. They suggested the performance evaluation of more protocols under the scenarios used by them.

Pucha et al, 2007 have studied the impact of traffic patterns on routing protocols DSDV, DSR and AODV via detailed simulations. They performed the simulation by varying the number of connections and mobility of nodes. They also discussed the effects of traffic volume and network size on the performance of these protocols. The goal of their research was to motivate the need for comprehensive performance evaluation of ad hoc networks to not only include rich and diverse set of mobility models but also to include diverse traffic patterns that govern changes in the performance of the protocols.

Rajeswari & Venkataramani, 2011 carried out the performance analysis of a proposed MANET routing protocol with respect to Packet Delivery Ratio, Average

End-to-End Delay and Throughput by using TCP and CBR traffic models. Simulation time and number of nodes were varied to see the effects on these protocols under different scenarios. They have suggested the optimization of the drop and queues involved in the protocol as future enhancement.

Patil, 2012 has investigated the effect of change in number of nodes, speed of node and pause times on packet delivery ratio (PDR) of MANET reactive routing protocols. Here, it has been analyzed and compared the performance of MANET routing protocols AODV and DSR based on both CBR and TCP based traffic patterns. The results were discussed by drawing out a comparison between performance bases on CBR as well as TCP based traffic.

Singh et al, 2013 have evaluated the performance of three well known mobile ad hoc routing protocols AODV, DSR and WRP with two traffic generators CBR and FTP. The performance metrics used in this study were packet delivery ratio and average throughput. They have varied the number of nodes as well as the node density to carry out the simulations effectively thereby drawing out conclusions whether which protocol is suitable for a particular scenario.

He, 2002 reviewed the DSDV protocol extensively along with the problems faced by it. The algorithms used by DSDV for its functioning have been discussed through this research. Then the author has analysed the performance of DSDV in terms of packet delivery ratio, end-to-end delay and normalized routing load. Finally the QoS routing performance have been calculated by carrying out simulation analysis.

Patil et al, 2009 carried out the performance analysis of DSDV and AODV MANET routing protocols. However they were not able to analyse the performance of TORA routing protocol because it contained bugs in NS-2. They measured the convergence time of both the protocols and tried to find their strengths and weaknesses. The number of nodes and pause times were varied. The simulation of TORA has been left for future work.

Kumar et al, 2009 have carried out the performance analysis and comparison of various mobile ad hoc routing protocols viz DSDV, DSR and AODV. The performance has been measured in terms of number of retransmission attempts, Control traffic

sent, Control traffic received, Data Traffic sent, Data Traffic received and throughput. According to the results DSDV performs well in terms of data traffic received and throughput. DSR and DSDV perform inadequately as compared to AODV for the control traffic sent and throughput. AODV and DSDV demonstrate an average level of performance for the data traffic received and data traffic sent, respectively.

Tyagi & Chauhan, 2010 presented a simulation based comparison and performance analysis of protocols DSR, AODV (Reactive) and DSDV (Proactive) with respect to different parameters like PDF, Average end to end delay, Routing Overheads and Packet Loss. They discussed that each protocol is suited for different scenarios and it is not clear whether which protocol is suitable for all the scenarios.

Gupta et al, 2010 have termed simulation as an important method for evaluating the performance of mobile ad hoc routing protocols. They analyzed the performance of protocols AODV, DSR and TORA using performance parameters average end to end delay and packet delivery ratio. They have given detailed description of these protocols and NS-2 is used to carry out the simulation using CBR traffic sources and random waypoint mobility model. According to their conclusion, TORA is well suited for dense mobile ad hoc networks because of its support for multiple routes and multicasting. However, this study was made by varying the pause times and keeping the number of nodes and other parameters constant.

Soujanya et al, 2011 have provided a detailed study of various mobile ad hoc routing protocols like DSR, AODV and TORA, FSR, ZPR, WRP and have discussed their pros and cons briefly thereby highlighting why it is necessary to carry out the performance investigations of these protocols in order to rectify the flaws present in their implementations. Various performance parameters like routing overhead, average delay, throughput, media access delay, packet delivery ratio, and path optimality have also been mentioned, which can be used to carry out comprehensive performance analysis.

Narra et al, 2011 have performed extensive analysis of DSDV mobile ad hoc routing protocol under various scenarios like varying the number of nodes and pause times. The implementation of DSDV on ns-3 simulator has been explained thoroughly

in this paper. The simulation analysis has been done with respect to performance parameters like packet delivery ratio, routing overhead and delay. Further, the performance of DSDV was compared with OSLR and AODV protocols. Results show that the overhead and packet delivery ratio of DSDV is affected by number of nodes.

Kumar et al, 2012 have analyzed the performance differentials to compare AODV, DSR and TORA ad hoc network routing protocols. The simulation results with different scenarios having thirty nodes have been reported through this study. The performance evaluation has been done in terms of number of hops per route, retransmission attempts, traffic sent, traffic received and throughput. The overall performance of TORA in the assumed scenarios was better than the other protocols.

Lim & Datta, 2012 have explained the TORA routing protocol in detail. It has been mentioned that TORA relies on IMEP (Internet MANET Encapsulation Protocol) to create, maintain and erase routes. They have studied the behaviour of IMEP that is expected to be the reason of poor performance of TORA. So the authors have listed down the flaws of IMEP and suggested two modifications that resulted in performance improvement of TORA in terms of packet delivery, routing overhead and average packet latency.

Rajeshwari et al, 2012 have talked about secure group communication in MANETs. The simulation analysis has been made with respect to the throughput, packet transmission between the source and destination node. They performed the study on four MANET protocols viz AODV, DSR, TORA and OLSR. It was concluded that in the assumed simulation environment AODV performed better than the other protocols and mobility of nodes played an important role in the simulation.

Khiavi et al, 2012 analyzed the performance of DSDV, AODV, DSR and TORA mobile ad hoc routing protocols by extensive simulations. The performance metrics used were Packet Delivery Ratio, Network Life Time, System Life Time, End-to-End Delay and Routing Overhead. According to their conclusion, DSDV performed better for a wide range of scenarios by varying the pause times and number of nodes.

CHAPTER 3

DSDV, TORA AND IMEP PROTOCOLS

Efficient routing is a major challenge in the field of networking. Routing protocols play a crucial role and thus research investigations into the behavior of protocols is an on-going process. The detailed functionality of DSDV and TORA protocols have been explained in this chapter along with IMEP protocol on which TORA is based.

3.1. DSDV Mobile Ad-Hoc Routing Protocol (Proactive Protocol)

DSDV, short for Destination Sequenced Distance Vector, is based on the idea of Routing Information protocol (RIP) that uses classical Bellman Ford routing algorithm (He, 2002).

3.1.1. Packet Transmission using DSDV Protocol:

According to (Lesiuk, 1998), DSDV is a proactive routing protocol, hence the routes are predefined. Whenever a node wants to send packets to a particular destination, routing table lookup is performed and the packet is transmitted to the next hop on the route towards destination. Packet transmission using DSDV protocol involves a series of steps. This has been explained through an example. Suppose node 1 wants to send packets to node 7. Node 1 will look for the next hop in its routing table towards destination node i.e. node 2. It will then transfer the packet to the next hop node 2 as shown in Figure 3.1. Same process will be repeated at all the nodes, until the packet reaches its intended destination.

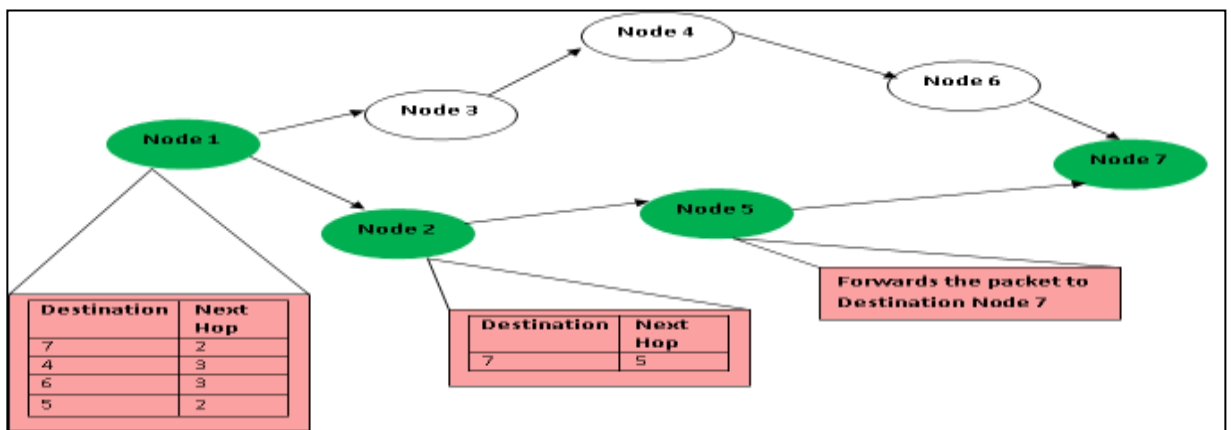


Figure 3.1. Transmission of a packet from source node 1 to destination node 7

3.1.2. Managing the Routing Table

The pivotal point of DSDV (He, 2002) (Lesiuk, 1998) is the generation and maintenance of the routing tables. Everytime the network topology changes (for example, the link between node 4 and 5 is broken in Figure 3.2.), the routing table needs updation and when routing tables are not updated, loops may be formed. To carry out routing table maintenance, some additional information is also stored inside the routing table i.e. Destination Address, Next Hop Address, Route Metric, Route Sequence Number. Each node will broadcast a routing table update packet periodically as well as immediately when there is a topology change. Update packet starts out with a metric of 1. Each receiving neighbour node is one hop away from node that sends the Update packet. The neighbours will increment this metric and then retransmit the update packet. Process is repeated round the clock until every node in the network has received a copy of the update packet with a corresponding metric. For example in Figure 3.2., Node 5 broadcasts an UPDATE packet which is received and incremented by the rest of the nodes. If node receives duplicate update packets, it will only consider the packet with smallest metric and ignore the rest.

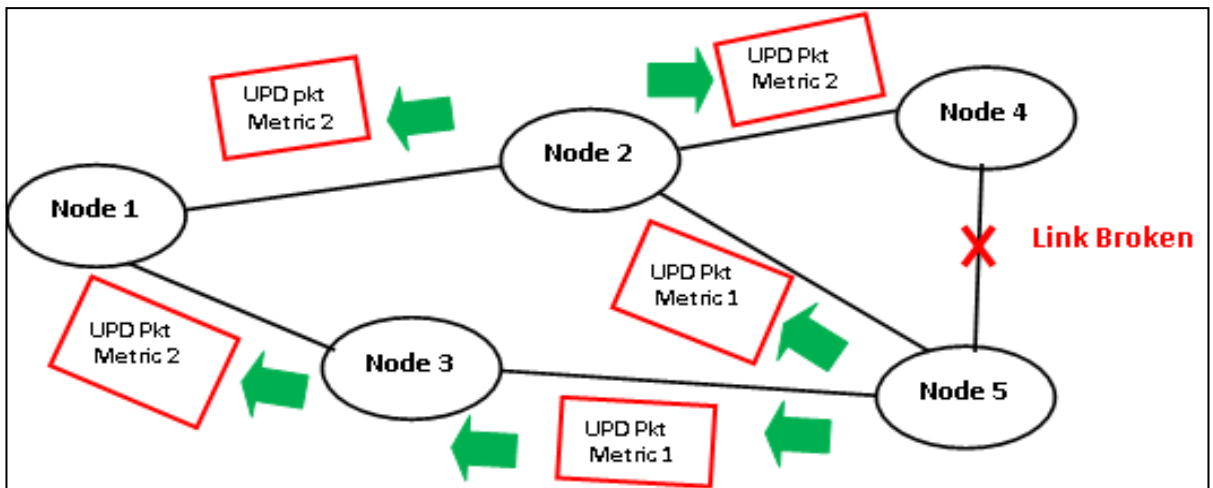


Figure 3.2. Routing table management using UPDATE packet

3.1.3. Distinguishing the Stale Packets

According to (Lesiuk, 1998), each update packet is tagged by the original node with a Sequence number (monotonically increasing number which uniquely identifies each update packet from the given node) to distinguish stale packets. Each time an

update packet is forwarded; the packet not only contains the address of eventual destination, but also contains address of transmitting node. The three cases of updation have been described below:

Case I: When Sequence number in update packet is lesser than the Sequence number already in routing table, then UPDATE is ignored.

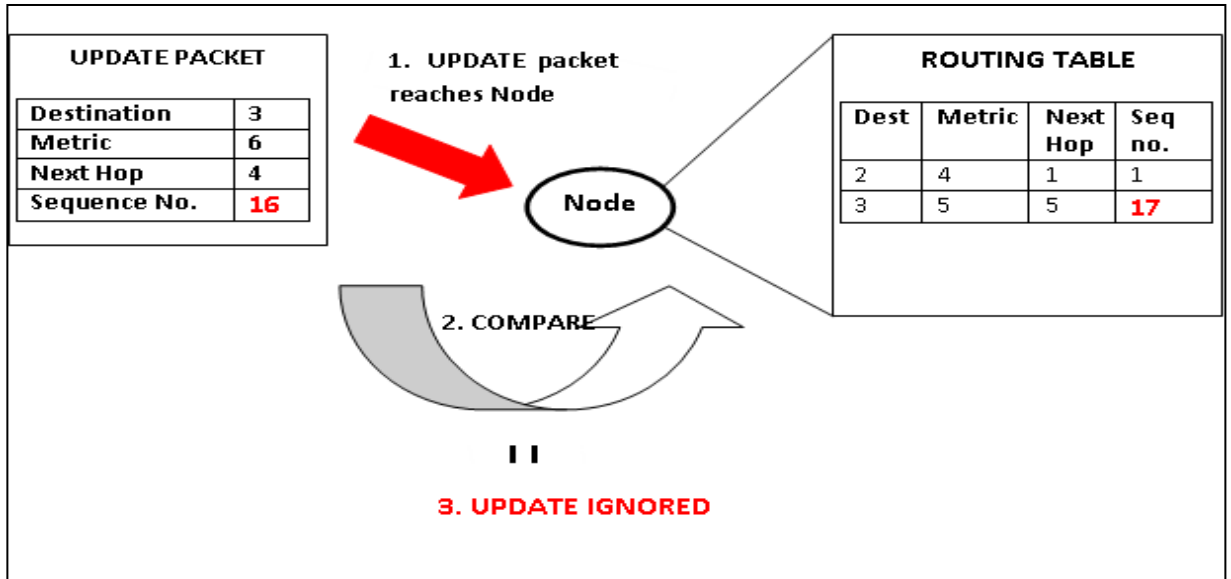


Figure 3.3. Smaller sequence number in update packet

Case II: When Sequence number in update packet is equal to Sequence number already in routing table, then the metric fields are compared. If metric of Update packet is greater than UPDATE is ignored else it is performed.

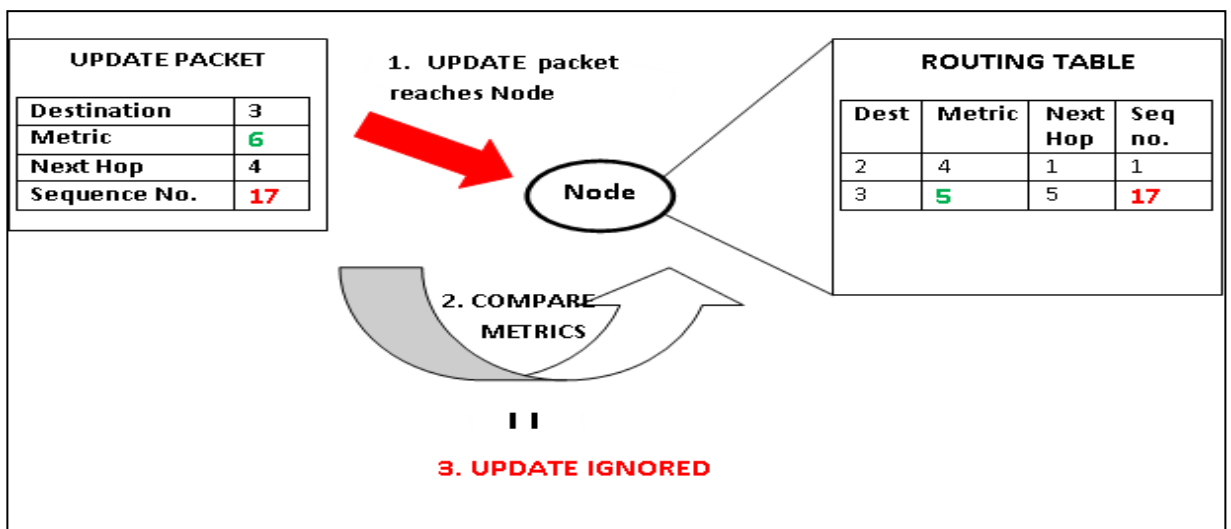


Figure 3.4. Equal sequence number in Update packet

Case III: When Sequence number in update packet is greater than Sequence number already in routing table, then UPDATE is performed.

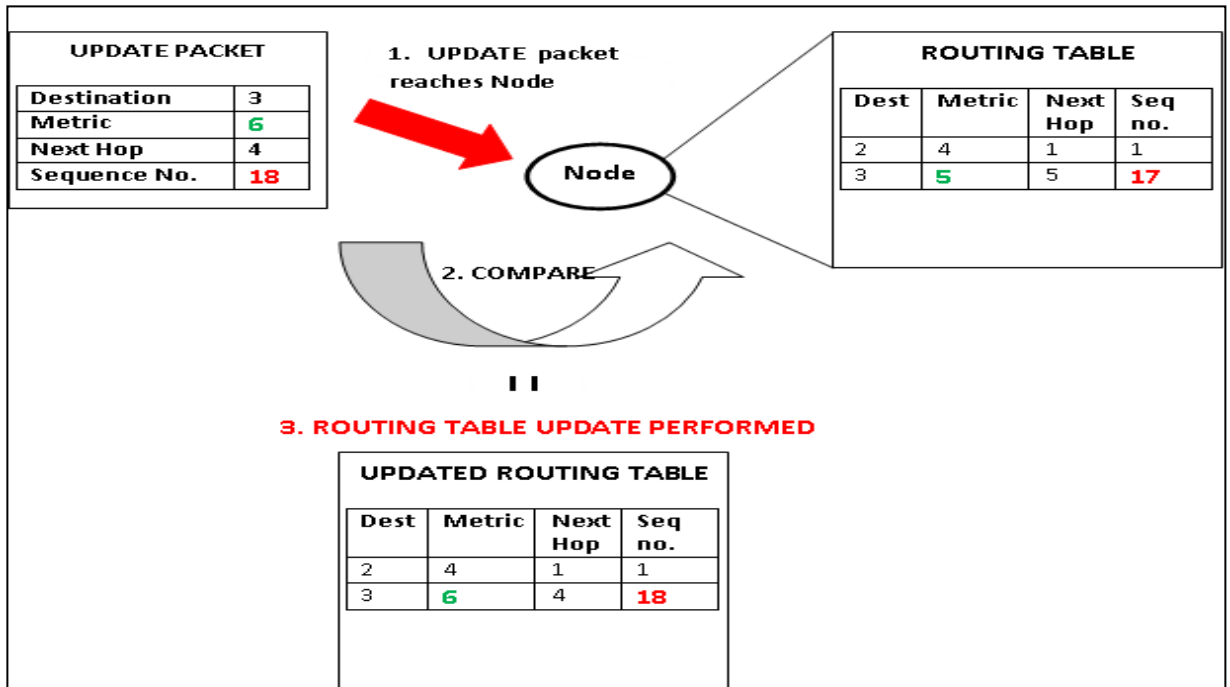


Figure 3.5. Greater Sequence number in Update packet

Each node periodically transmits its entire routing table to its neighbors using update packets. Neighbors will update their tables based on this information, if required (Lesiuk, 1998).

3.2. TORA -Temporally Ordered Routing Algorithm (Hybrid Protocol)

TORA (Park & Corson, 1997) is a source initiated on-demand routing protocol that is highly adaptive, proficient and scalable distributed routing algorithm based on the concept of Link Reversal and proposed for highly dynamic mobile and multi hop wireless networks. Link reversal algorithms (Gafni & Bertsekas, 1981) provides a simple mechanism for routing in mobile ad hoc networks and maintain routes to a particular destination node even during frequent changes in topology of the network (Busch et al., 2003). These are adaptive, self-stabilizing, distributed mobile ad hoc routing algorithms (Perkins, 2000) which form basis of TORA routing algorithm.

TORA finds out multiple routes from a source node to a destination node. Each node maintains routing information about adjacent nodes. During any topological

change, the control messages are exchanged among the set of nodes which are near the occurrence region of that topological change (Lesiuk, 1998). This has been marked as an important and main feature of TORA protocol. It basically involves three functions:

- Route Creation
- Route Maintenance
- Route Erasure

3.2.1. TORA Route Creation: According to (Lesiuk, 1998), It is done by using QRY (Query) and UPD (Update) packets.

Step 1: Set the height and reference level of destination node to 0 and that of all other nodes to NULL (Not Defined).

Step 2: Source will broadcast the QRY packet with destination node ID included in it.

Steps 1 and 2 have been described through Figure 3.6.

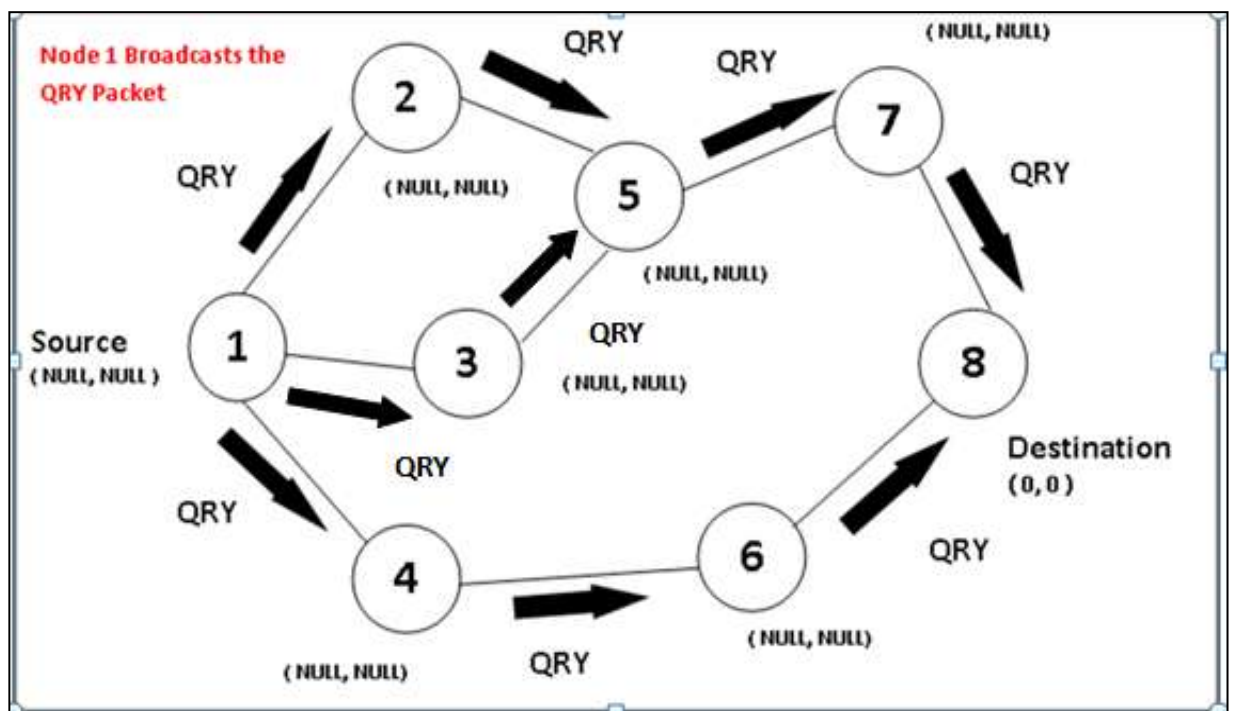


Figure 3.6. Propagation of QRY message through the network (Route Creation)

Step 3: A node with non-NULL height will respond with UPD packet which contains the height of that node in it.

Step 4: Every node that receives the UPD packet will set its height to one more than that included in the UPD (i.e one more than the height generated by the node which created the UPD packet).

Steps 3 and 4 have been described in Figure 3.7.

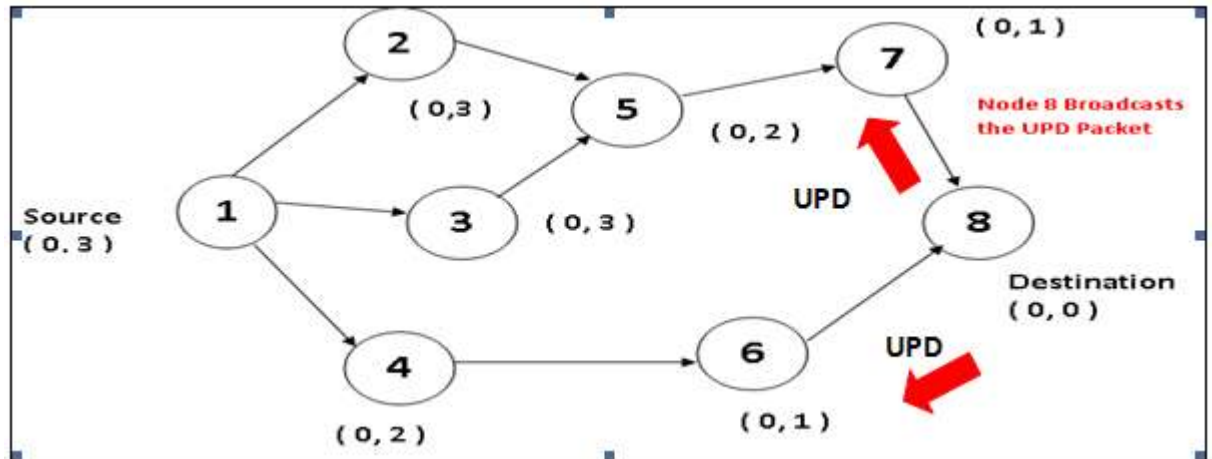


Figure 3.7. UPD reply from Destination and formation of DAG (Directed Acyclic Graph)

In Figure. 3.7, Destination node 8 broadcasts the UPD packet, which is sent to every node. Every node which receives this UPD packet sets its height as one more than that contained in the UPD packet. A DAG (Directed Acyclic Graph) is formed and arrows indicate the direction in which the UPD packet is received. For example, node 6 receives UPD packet from destination node 8 and sets its height to 1 as the height metric obtained in the UPD packet was 0.

3.2.2. TORA Route Maintenance

According to (Shah, 2001), a node with higher height is said to be on upstream link whereas a node with lower height is said to be on downstream link. For example; nodes 6 and 7 are on upstream links for node 8 and node 8 is on downstream link for these nodes. Whenever a node moves from its location, the route of the DAG is broken and route maintenance phase is called to re-establish the DAG for same destination again. When the last downstream link of a node fails, it leads to a new reference level. This result in the propagation of the new reference level by the neighboring nodes and links are then reversed to reflect the changes made while

adapting to the new reference level. Figure 3.8. shows how the link reversal is being used in TORA (Park, 1997). Here node 7 changes its location, hence node 5 forms a new reference level 1 because its downstream link to node 7 has failed i.e. there is now no outgoing link from node 5. The links towards nodes 2 and 3 are now reversed. So in TORA link reversal occurs only when there is no outgoing link.

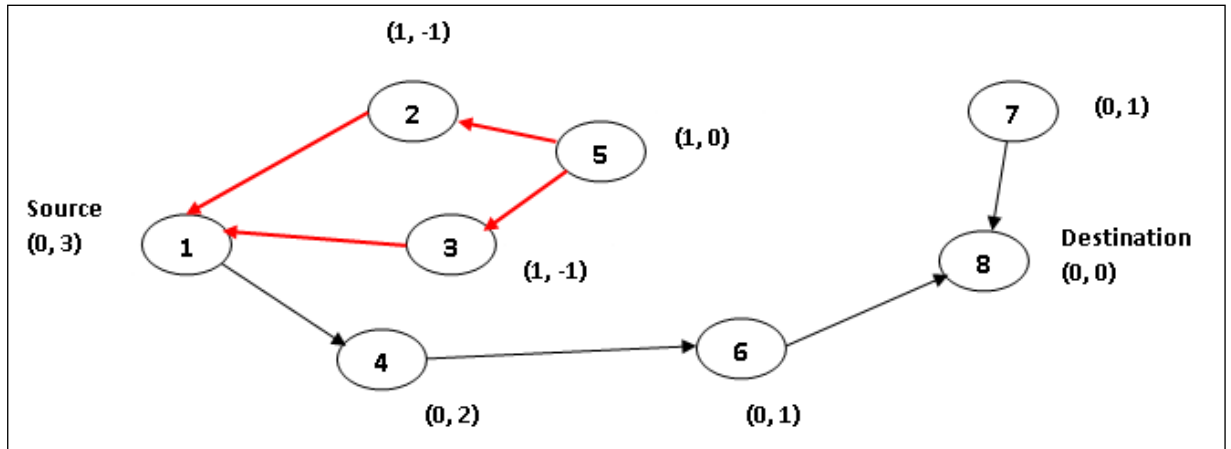


Figure 3.8. Route Re-establishment using Link Reversal (Route Maintenance)

3.2.3. TORA Route Erasure

As described by (Hoffman & Romanoff, 2012), TORA floods a broadcast clear packet CLR throughout the network to initiate the route erasure phase thereby erasing all the invalid routes. Whenever there is a partition in the network, the CLR packet is sent by the node which detects the partition. When a node receives a CLR packet it resets the hierarchy of all neighbor nodes to NULL and the route creation process is started again from the last node which receives the CLR packet. As shown in Figure 3.9., the partition is detected by nodes 5 and 6. Hence these nodes broadcast a CLR packet to initiate route erasure.

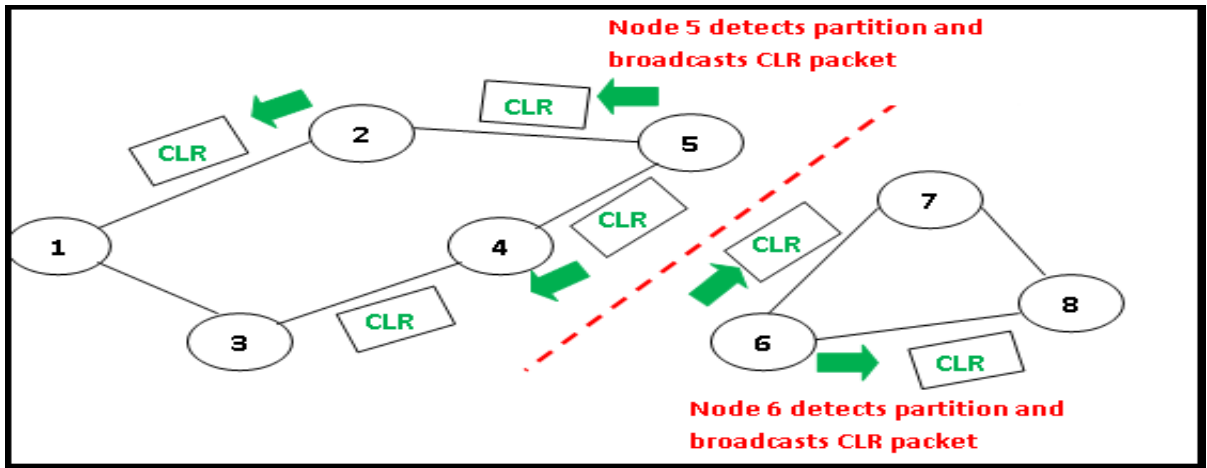


Figure 3.9. Route Erasure in TORA using CLR packet

3.3. Internet MANET Encapsulation Protocol

IMEP is a multipurpose network-layer protocol which is designed to support the operation of many routing algorithms that are used for MANETs. TORA protocol also uses the services of IMEP to facilitate its various functions. IMEP incorporates mechanisms to support link status sensing, control message aggregation and encapsulation, broadcast reliability, network layer address resolution, interrouter security authentication procedures (Corsen & Park, 1997) (Ilyas, 2003). It runs at network layer where Upper Layer Protocol (ULP) packets from above layers are encapsulated in IMEP messages and further encapsulated into IP packets at network layer as shown in Figure 3.10.

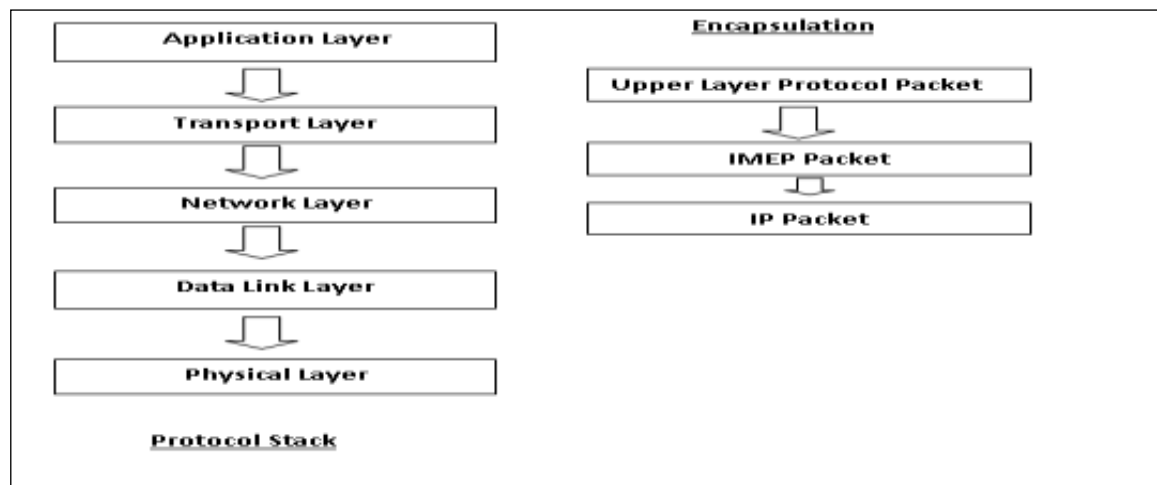


Figure 3.10. IMEP running at Network layer (Corsen & Park, 1997)

IMEP and TORA have been designed to work together synergistically. With the support of IMEP, TORA achieves multi-hop distributed routing, multi-path support and route cycle avoidance (Zubairi, 2009).

3.3.1. Link Status Sensing Mechanism of IMEP: (Corsen et al, 1999) have described that Routing protocols like TORA usually require correct knowledge of the status of links between the neighboring nodes. A link can be unidirectional (incoming or outgoing) or bidirectional. Status of link is depicted by following terms:-

- **UP:** Link is present between the nodes. It means that the status of the link is non-null and it is either unidirectional or bidirectional.
- **DOWN:** Link is not present and therefore has null status.

The UP and DOWN links are depicted ahead in Figure 3.11 and Figure 3.12.

Whenever there is a link change, upper layer protocols are informed through callback functions in MANET IMEP interface. This study is focused on the Link/Connection status sensing mechanism of IMEP protocol. This mechanism can be divided into two categories- Implicit and Explicit (Corsen et al, 1999). The major focus of this research is on Implicit link failure detection method.

(i) Explicit Link Failure Detection Method: (Corsen et al, 1999) (Lim & Datta, 2012) have described that to determine link status information, a node broadcasts BEACON packets to its one-hop neighbors. When node receives a reply in the form of an ECHO packet from a neighbor, it labels that link as bi-directional. Node continues to send BEACON packets every BEACON_PERIOD interval and sets a MAX_BEACON_TIME.

$$\text{MAX_BEACON_TIME} = (\text{Maximum number of BEACON retransmissions} * \text{BEACON_PERIOD})$$

The link is marked as DOWN when node does not receive any ECHO packet after MAX_BEACON_TIME.

(ii) Implicit Link Failure Detection Method: According to (Corsen et al., 1999) (Lim & Datta, 2012), IMEP sends out OBM packets. In this method, node 1 sends out

OBM packets to the neighbouring nodes and reply is in form of an ACK packet (Figure 3.11) similar to BEACON and ECHO packets used in the explicit method. There is a $MAX_RETRANS_TIME = (RETRANS_PERIOD \text{ between each transmission} * \text{maximum number of retransmissions})$. After $MAX_RETRANS_TIME$, if a neighbour replies with ACK packet, the link is marked as UP (Figure 3.11) and if a neighbor did not reply with ACK packet then that link is marked DOWN as shown in Figure 3.12.

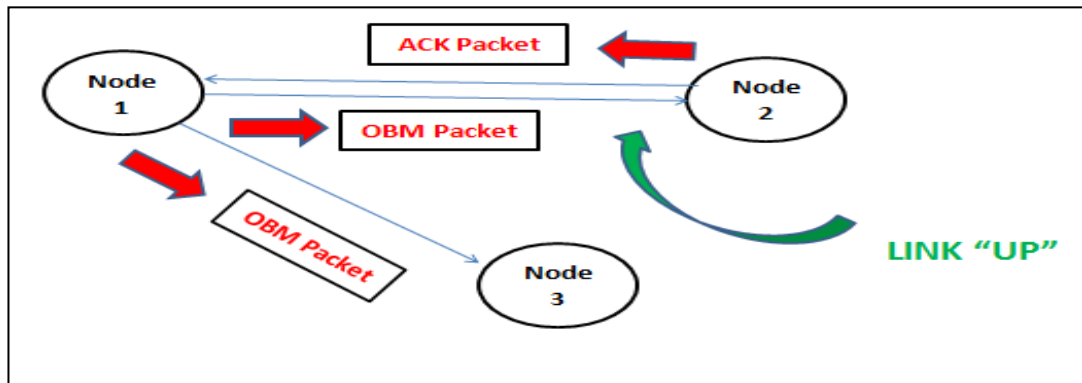


Figure 3.11. Detection of "UP" Link

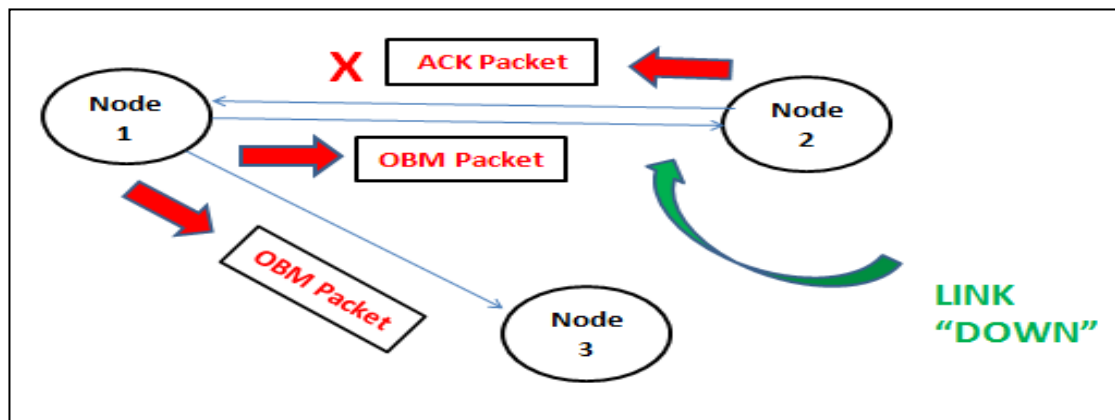


Figure 3.12. Detection of "DOWN" Link

3.4. CHAPTER SUMMARY

In this chapter, the detailed description of protocols used for research work i.e DSDV and TORA is given. The functioning of these protocols has been enlightened along with the IMEP protocol which facilitates the route creation, maintenance and erasure functions of TORA. The implicit and explicit link failure detection mechanism of IMEP has also been discussed.

CHAPTER 4

TOOLS USED

Simulation (Issariyakul & Hossain, 2012). is basically a mixture of science and art. It is widely used in engineering research. There are a number of simulation tools available to simulate the behavior of various networks and routing protocols. Since Mobile ad hoc networks have not been deployed widely, simulation is a good choice to model their behavior and test their suitability under different scenarios. Simulation makes it possible to understand the behavior of the networks and the underlying routing protocols so as to find out their applicability in different situations.

4.1. Simulation Tool Used: NS2 (ns-allinone-2.35 Simulation package)

According to (Issariyakul & Hossain, 2012) NS-2 is an event driven packet level network simulator developed as part of Virtual Internet Testbed (VINT) project. It is an open source software which can be used both in Windows 32 and Linux operating systems. It is an object oriented simulator written in C++, with an OTcl Interpreter as a frontend. It is used to evaluate the performance of existing network protocols as well as new protocols before their deployment in actual scenarios. It can be used to run large scale experiments which are not difficult to implement in reality in order to understand their feasibility and effectiveness (Meenaghan & Delaney, 2004). A variety of networks can be implemented using this software.

(Issariyakul & Hossain, 2012) have mentioned that NS2 uses OTcl to create and configure a network whereas C++ is used to run the simulation. All C++ codes need to be compiled and linked to create an executable file. Whenever any change is made in the .cc or .h files, the “make” command is executed preceded by “make clean” command. This requires time. Whenever any change is made in the “make-file” utility, “./configure” command needs to be executed. Whereas, OTcl is an interpreter. Any changes made in it, need not be compiled. So it is fast to change. The basic architecture of NS2 is shown in Figure 4.1.

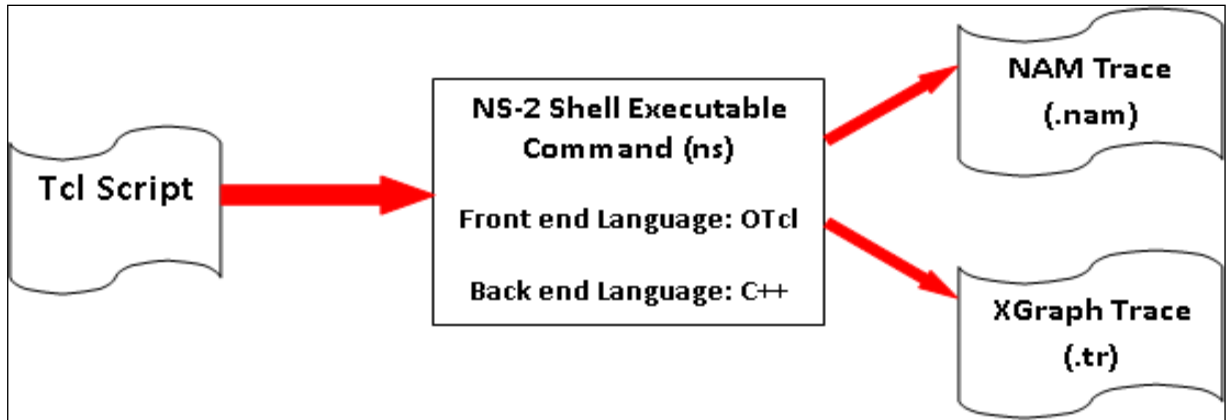


Figure 4.1: Basic architecture of NS2

An NS-2 script in Tcl scripting language is fed as input to the NS-2 shell Executable command (ns). After running the simulation, two files are generated as output: NAM Trace and X-Graph Trace. NAM Trace is used to show the animation of movement and communication among nodes and X-Graph trace is analyzed by the AWK scripts to calculate various performance metrics as shown in Figure 4.2.

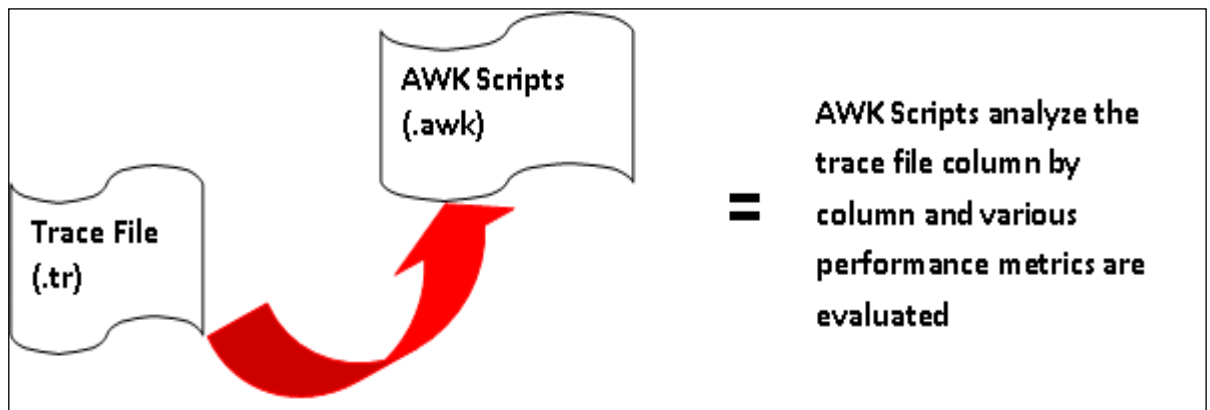


Figure 4.2: Role of AWK Scripts

4.1.1. Installation of ns-allinone-2.35 Package on Linux Ubuntu 12.04 (Dubey, 2013)

There is an “install” script in the allinone package which configures the NS2 environment and uses “make” utility to create NS2 executable file.

- (i) Download the NS-allinone-2.35 package from <http://www.isi.edu/nsnam/ns> and copy it in your HOME Directory.
- (ii) Open Terminal using command ALT+CTRL+T.

(iii) Execute following commands:

- `cd ns-allinone-2.35`
- `sudo apt-get update`
- `sudo apt-get install build-essential autoconf automake libmxu-dev`

(iv) Execute the command : `./install`

(v) Execute following command to install the X-Graph utility:

- `$ sudo apt-get install xgraph`

(vi) Set environmental Variables using command : `gedit ~/.bashrc`

Gedit window will be opened. In the opened file, add following lines at the end
Replace “/your/path” by the folder where you have saved the downloaded and
extracted package in Home Directory.

LD_LIBRARY_PATH

```
OTCL_LIB=/your/path/ns-allinone-2.35/otcl-1.13
```

```
NS2_LIB=/your/path/ns-allinone-2.35/lib
```

```
X11_LIB=/usr/X11R6/lib
```

```
USR_LOCAL_LIB=/usr/local/lib
```

```
export
```

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$OTCL_LIB:$NS2_LIB:$X11_LIB:$U
```

```
SR_LOCAL_LIB
```

TCL_LIBRARY

```
TCL_LIB=/your/path/ns-allinone-2.35/tcl8.4.18/library
```

```
USR_LIB=/usr/lib
```

```
export
```

```
TCL_LIBRARY=$TCL_LIB:$USR_LIB
```

PATH

```
XGRAPH=/your/path/ns-allinone-2.35/bin:/your/path/ns-allinone
```

```
2.35/tcl8.4.18/unix:/your/path/ns-allinone- 2.35/tk8.4.18/unix
```

```
NS=/your/path/ns-allinone-2.35/ns-2.35/
```

```
NAM=/your/path/ns-allinone-2.35/nam-1.14/
```

```
PATH=$PATH:$XGRAPH:$NS:$NAM
```

(vii) Now, run “ns” command after which “%” symbol appears in terminal window, which signifies successful installation.

4.1.2. Writing a Tcl Script in NS2 (Issariyakul & Hossain, 2012)

Class Tcl is a C++ class which acts as an interface to OTcl domain. The Tcl script is fed as input to ns2. Steps for writing a Tcl Script have been discussed below:

(i) Creating a Simulator: The simulator is created by using command “new Simulator”. This returns a handle called “Simulator”, which is stored in ns variable. The command executed for this purpose is:

- `set ns [new Simulator]`

(ii) Creating Trace File: Following line is used to open out.tr file for writing the trace data.

- `set mytrace [open out.tr w]`
- `$ns trace-all $mytrace`

The second line declares the file handle mytrace for file out.tr. All the information regarding regular trace is stored in the file out.tr by execution of the second line.

(iii) Creating a NAM Trace File: Following lines open out.nam file and create myNAM handle for this file. All the information regarding NAM trace is stored in out.nam file.

- `set myNAM [open out.nam w]`
- `$ns namtrace-all $myNAM`

(iv) Defining a “finish” procedure: This procedure is called immediately before the simulation process ends. The keyword global tells the Tcl interpreter that the variables ns, myTrace, myNAM have been defined globally. Then the buffer of packet tracing variable is flushed. The files associated with handles myTrace and myNAM are closed. The statement “nam out.nam &” is executed from the shell environment. And the last line informs the NS2 to exit with code 0.

```
proc finish { } {  
global ns mytrace myNAM  
$ns flush-trace  
close $mytrace
```

```
close $myNAM
exec nam out.nam &
exit 0 }
```

(v) Creating the nodes: Nodes can be created as follows:

- set node_(0) [\$ns node]
- set node_(1) [\$ns node]

(vi) Generating Node Movements: The movements can be generated for nodes using setdest tool, which is available in the path ns2-allinone-2.35/ns-2.35/indep-utils/cmu-scen-gen/setdest/. However, in this study the movements have been generated using BonnMotion tool [Refer Appendix C].

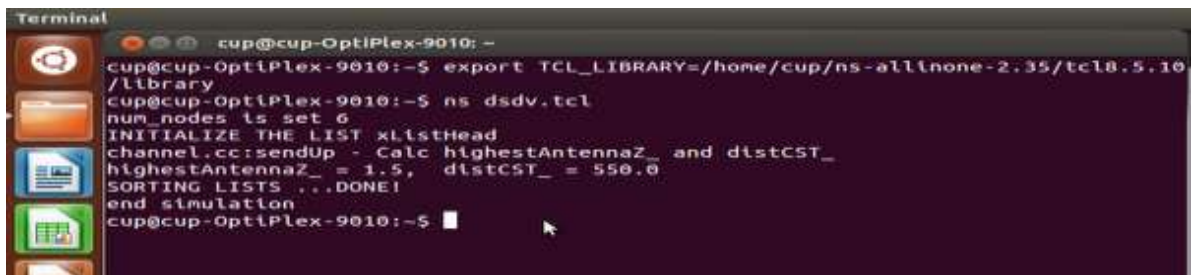
(vii) Traffic Generation: There are different kinds of traffic sources available in NS2. Most widely used traffic is the CBR traffic. Some other traffic patterns like FTP, TELNET, HTTP, Exponential, Pareto are also available in NS2. Traffic sources used in this study have been discussed in Appendix B.

4.1.3. Running the Simulation

The tcl file is saved as file_name.tcl, which is run using following command:

ns file_name.tcl

Suppose the file name is dsdv.tcl. This file is run using command *ns dsdv.tcl*. The simulation is done as shown in Figure 4.3.



```
Terminal
cup@cup-OptiPlex-9010:~$ export TCL_LIBRARY=/home/cup/ns-allinone-2.35/tcl8.5.10/library
cup@cup-OptiPlex-9010:~$ ns dsdv.tcl
num_nodes is set 6
INITIALIZE THE LIST xListHead
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
SORTING LISTS ...DONE!
end simulation
cup@cup-OptiPlex-9010:~$
```

Figure 4.3. Running dsdv.tcl Script

Here ns is an C++ executable file, which is created from compilation. After executing this file, two files are generated- Trace and NAM file. Trace file is used to calculate the performance metrics like throughput, delay, overhead etc and is also used to make graphs using the XGraph utility in ns2. The NAM file is used to run the

animation and understand the behaviour of nodes, their movements, traffic analysis and packet dropping etc.

Suppose the name of the trace file is dsdv.tr. It can be used to calculate the performance metrics with help of AWK scripts which contain the coding for these metrics [Refer Appendix D for AWK scripts].The command used for this purpose is:

awk -f <awk script name> <trace file name>

Figure 4.4. shows the execution of trace file by using awk script and results for various parameters are listed.

```
cup@cup-OptiPlex-9010:~$ awk -f all.awk dsdv.tr
GeneratedPackets = 22764
ReceivedPackets = 22654
Packet Delivery Ratio = 99.5168%
Total Dropped Packets = 1
Average End-to-End Delay = 58.3158 ms
```

Figure 4.4. Using awk script to calculate performance metrics

4.2. BonnMotion Tool to Generate Movement Scenarios using Various Mobility Models

BonnMotion is a Java software which can be installed on Windows as well as Linux (Aschenbruck et al, 2010) (BonnMotion, 2013). In this research, three mobility models have been used viz. Random Waypoint, Random Direction and Manhattan Grid mobility model. These models have been discussed in detail in next chapter.

4.2.1. Installation Steps

- (i) To install this software first make sure java is installed on your system, then double click on install script inside the BonnMotion package, which can be placed in any folder or Desktop.
- (ii) Terminal window will be opened and it will ask you to enter your java path i.e the path where Java has been installed.
- (iii) Next it will ask for BonnMotion path, where you have saved your extracted file.
- (iv) After this 3 files i.e. bm.bat, compile.bat and makedoc.bat will be created in the bin folder. This means that BonnMotion is successfully installed in your system.

4.2.2. Generation of Movements for Nodes Using Different Mobility Models

The commands mentioned below are run in terminal to generate movements for nodes using Random Waypoint, Manhattan Grid and Random Direction mobility models.

(i) For Random Waypoint Model:

```
sh bm -f scenario_name RandomWaypoint -n <number of nodes> -d <simulation time duration in seconds> -i <initial phase cutoff in seconds>
```

Example: *sh bm -f scen1 RandomWaypoint -n 6 -d 900 -i 3600*

This will generate two files: scen1.params and scen1.movements.gz. But these two files are not ns-2 compatible. To make them runnable in ns-2, following command is executed:

```
sh bm NSFile -f scen1
```

This will generate the file: scen1.ns_movements.

(ii) For Manhattan Grid Model:

```
sh bm -f scenario_name ManhattanGrid -n <number of nodes> -d <simulation time duration in seconds> -i <initial phase cutoff in seconds>
```

Example: *sh bm -f scen1 ManhattanGrid -n 6 -d 900 -i 3600*

This will generate two files: scen1.params and scen1.movements.gz.

But these two files are not ns-2 compatible.

To make them runnable in ns-2, following command is executed:

```
sh bm NSFile -f scen1
```

This will generate the file: scen1.ns_movements.

(iii) For Random Direction Model:

```
sh bm -f scenario_name RandomDirection -n <number of nodes> -d <simulation time duration in seconds> -i <initial phase cutoff in seconds>
```

Example: *sh bm -f scen1 RandomDirection -n 6 -d 900 -i 3600*

This will generate two files: scen1.params and scen1.movements.gz.

But these two files are not ns-2 compatible.

To make them runnable in ns-2, following command is executed:

sh bm NSFile -f scen1

This will generate the file: scen1.ns_movements.

The code of scen1.ns_movements files generated by all three models is further incorporated in the Tcl scripts to implement mobility of nodes.

4.3. CHAPTER SUMMARY

This chapter describes the tools used to accomplish this dissertation work. NS-2 simulator has been described along with its installation procedure. The process of creating Tcl script and running the simulation in NS-2 has also been discussed. Further the tool used to generate movements for nodes called BonnMotion has also been explained. Its installation steps have been detailed and the commands used to generate movements have been described.

CHAPTER 5

PROPOSED WORK AND SIMULATION ENVIRONMENT

This research is focused on the performance evaluation of DSDV and TORA protocols by varying the underlying mobility models (Random Waypoint, Manhattan Grid and Random Direction) and traffic patterns (FTP, HTTP, TELNET). Further, the degrading performance of TORA is investigated with rise in number of connections and hence the proposed modification by Lim & Datta was performed in the link sensing mechanism of IMEP protocol. The proposed modification and the simulation environment have been described in this chapter.

5.1. Problems in original TORA Protocol

The code of original TORA protocol is prone to various errors in `tora.cc` and `tora.h` files of NS-2 simulator which need to be mitigated by applying patches, which include performing the changes as described in section 5.1.1. and 5.1.2. Even after applying these changes, its performance degrades with rise in number of connections. This problem lies in the link status sensing mechanism of IMEP protocol on which TORA is based. This mechanism is investigated in this research work and proposed modification has been performed as discussed ahead in section 5.2.

5.1.1. “Invoked from within” Error:

The reason behind this error is the absence of a code for port classifier in `tora.cc` and `tora.h` files present in `ns-allinone-2.35/ns-2.35/tora` path. This error can be rectified by inserting following code in the command function of `tora.cc` file (Sahraei, 2006) (Islam, 2011):-

```
else if (strcmp(argv[1], "port-dmux") == 0) {
    dmux = (PortClassifier *)TclObject::lookup(argv[2]);
    if (dmux == 0) {
        fprintf(stderr, "%s: %s lookup of %s failed\n", __FILE__,
            argv[1], argv[2]);
        return TCL_ERROR;
    }
    return TCL_OK;
}
```

```
}
```

Tora.h file also needs to be modified by declaring a header file for the port classifier as follows:-

```
#include <classifier/classifier-port.h>
```

At the end of Tora.h file add following line:

```
protected:  
    /* for passing packets up to agents */  
    PortClassifier *dmux;  
};
```

5.1.2. Infinite Loop Error

After performing above modification, simulation of TORA started, but entered an infinite loop. This problem was rectified by inserting following code in the Retransmit-timer function of imep.cc file present in ns-allinone-2.35/ns-2.35/imep path (Erlend, 2010).

```
If (rexat-CURRENT_TIME < 0.000001) //Preventing eternal loop  
    rexmitTimer.start (0.000001);  
else  
    rexmitTimer.start (rexat - CURRENT_TIME);
```

After performing all these modifications, simulation of TORA was successful.

5.1.3. Poor Performance of TORA with Rise in Number of Connections

According to (Lim, K. H., & Datta, A., 2012), incorrect detection of link failures causes TORA to perform unnecessary route maintenance, resulting in a congested network and data packets not reaching their destinations. The implicit method of link failure detection is the main cause of incorrect detection of link failures. The main problem was due to ACK packets failing to reach the sender of OBM packets causing the senders to assume that their neighbouring links are down. The suggested modification was to increase the maximum number of OBM retransmissions, which increases the probability that the OBM packets sent by a node reaches all of its neighbours and nodes that fail to reply with ACK packets within

MAX_RETRANS_TIME have their links labeled as DOWN. The detailed description about Implicit and Explicit Link status sensing mechanism is given in section 3.3.

5.2. Proposed Modification to Enhance the Performance of TORA

The problem of poor performance of TORA with rise in number of traffic connections was rectified by implementing the modification proposed by K. H Lim and A. Datta. This was done by changing the IMEP constants as shown in Table 5.1 and Table 5.2.

Table 5.1. IMEP constant values for Original TORA

IMEP Constant	Value
BEACON_PERIOD	1.0 seconds
BEACON_JITTER	0.010 seconds
MAX_BEACON_TIME	(BEACON_PERIOD * 3)
Min Message Aggregation Delay	150 ms
Max Message Aggregation Delay	250 ms
OBM RETRANS_PERIOD	0.500
MAX_REXMITTS (OBM)	3
MAX_REXMITTS_TIME	(RETRANS_PERIOD * (MAX_REXMITTS +1))

Table 5.2. Modified IMEP constant values for Enhanced TORA

Modified IMEP Constant	Value
BEACON_PERIOD	3.0 seconds
BEACON_JITTER	0.010 seconds
MAX_BEACON_TIME	(BEACON_PERIOD * 3)
Min Message Aggregation Delay	150 ms
Max Message Aggregation Delay	250 ms
OBM RETRANS_PERIOD	0.500-0.800
MAX_REXMITTS (OBM)	5
MAX_REXMITTS_TIME	(RETRANS_PERIOD * (MAX_REXMITTS +1))

These changes have been introduced in the `imep_spec.h` in which the parametric constants related with Link/connection status sensing, Control message aggregation, broadcast reliability have been given.

Summary of all Modifications: The modifications were done in following files:-

- **Tora.cc:** Code for port-dmux was incorporated in the command function of this file as discussed in section 5.1.1.
- **Tora.h:** Header files related to port-dmux code were inserted in this file as discussed in section 5.1.1.
- **Imep.h:** Code for Retransmit timer was added in this file to avoid looping problem in TORA as discussed in section 5.1.2. The changes related to OBM retransmissions were also made.
- **Imep.spec.h:** The number of OBM retransmissions were increased thereby increasing the `MAX_REXMIT_TIME`. The `OBM_RETRANS_PERIOD` is also kept variable.

After making the changes, “make clean” command was run to destroy all previous object files and then “make” command was run to reflect the changes made and creation of new object files. After making these changes, the performance of enhanced TORA was further evaluated over 10 connections under various mobility models and FTP traffic. The performance of Enhanced-TORA under the best performing mobility model was then compared with Original-TORA protocol.

5.3.Simulation Environment for Performance Evaluation of DSDV, TORA and Enhanced-TORA

Tcl scripts for DSDV and TORA were written to set the topology of the network by declaring various parameters as shown in Table 5.3 (Refer Appendix A for list of experiments performed). However for Enhanced TORA, only FTP traffic has been used. The simulation was performed using NS-2.35 simulator on Linux Ubuntu 12.04 operating system under Intel Core-i7 processor.

Table 5.3. Parameters Used for Simulation of DSDV, TORA and Enhanced-TORA

Simulation Parameter	Value
Propagation Model	Two Ray Ground
MAC	IEEE 802.11
Interface Queue Type (IFQ)	Droptail/PriQueue
Antenna	OmniAntenna
Routing Protocol	DSDV, TORA
Dimensions of Topography	1600x1600
Simulation Time	200 ms
Traffic Source	FTP TELNET HTTP
Number of nodes	10, 30, 50, 70, 100
Maximum packet in IFQ	100 bytes
Mobility Models	Random Waypoint mobility model Random Direction mobility model Manhattan Grid mobility model

The parameters used in Table 5.3. have been described below:

5.3.1. Propagation Model

According to (Henderson, T. 2011), Radio propagation models are used to predict the received signal power of each packet. At the physical layer of each mobile wireless node, there is a receiving threshold value. Whenever a packet is received, if its received signal power is below the receiving threshold value, it is marked as erroneous packet and is therefore dropped by the MAC layer. Two Ray Ground Reflection Model is based on the assumption of both the direct path and a ground reflection path.

5.3.2. Medium Access Control (MAC) Protocol

As described by (Xu, S. & Saadawi. T., 2001), in mobile ad hoc network various mobile nodes share a medium whose access is facilitated by using a MAC protocol. In this work, standard IEEE 802.11 MAC protocol (Foh, C. H. & Zukerman, M., 2002)

has been used to control the access to the shared medium. This protocol covers the MAC and physical layer and makes use of Distribution Coordination Function (DCF). Here DCF is a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism (Kulenica, E. & Pham, C., 2002).

5.3.3. Interface Queue Type (IFQ)

IFQ (Henderson, T. 2011) is a FIFO queue that contains the packets of the routing protocols. In this study, priority Queue has been used which gives priority to routing protocol packets by inserting them at the head of queue.

5.3.4. Antenna Type

As described by (Balanis, C. A., 1997), “An Antenna is metallic device which radiates or receives radio waves”. It converts electronic signals to electromagnetic waves with minimum loss of signals. Omni-directional antennas mount vertically and transmit and receive equally in all directions within the horizontal plane (Antenna Basics).

5.3.5. Mobility Models Chosen for Study

The choice of mobility model plays crucial role in simulations of mobile ad hoc networks as it creates movement patterns for various nodes (Panda & Mohapatra, 2009). This study focuses on the use of three mobility models to generate movements for mobile ad hoc network simulation using DSDV and TORA protocols. These models are random based models- Random Waypoint, Random Direction and one city based model- Manhattan Grid model. These models have been discussed below.

(i) Random Based Mobility Models: In these models, the mobile nodes can move in random directions and freely without any constraints. The destination, speed and direction are chosen randomly and independent of other nodes (Gauthier, 2009).

- **Random Waypoint Mobility Model:** It is commonly used benchmark synthetic model for mobility which is simple and widely available. It can be generated directly using sedtest tool that is included in the ns2 itself. Each node moves along a zigzag line from one waypoint W_i to the next W_{i+1} . A Mobile node stays in one location for some time interval called pause time and when this time expires, it chooses a random destination in the simulation area that is distributed uniformly between [min-speed,

max-speed]. The node then travels towards its newly chosen destination at the selected speed. This process is repeated again when the node reaches the destination and pauses for a specified time as shown in Figure 5.1 (Hyytiä, 2005). If pause time is zero, mobility is continuous (Gauthier, 2009).

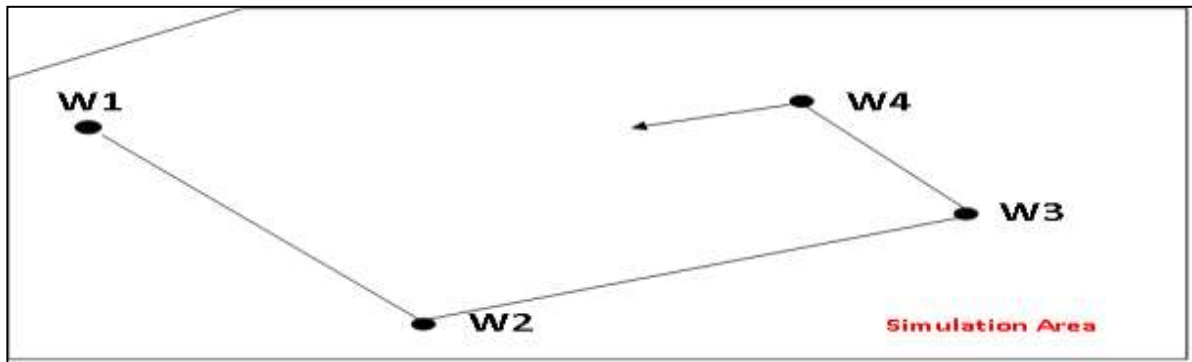


Figure 5.1. Movements using Random Waypoint Model

- **Random Direction Model:** It is a variant of random waypoint model. Mobile nodes choose a random direction in which to travel and the travels to the border of simulation area in that direction. Once the simulation boundary is reached, the node pauses for a specified time and chooses another angular direction (between 0 and 180 degrees) and continues the process again as shown in Figure 5.2. Since nodes reach the border and then take pause and choose new direction, thus average hop count in this model is greater than that in Random Waypoint Model (Roy, 2011).

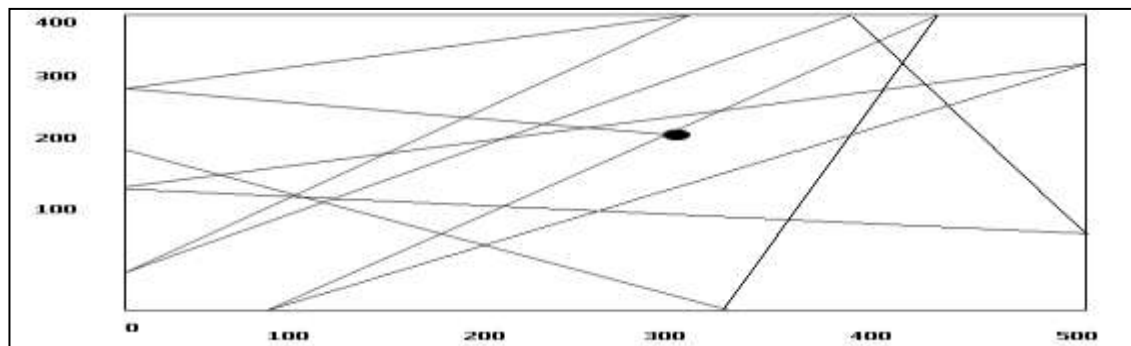


Figure 5.2. Movements using Random Direction Model

(ii) City Section Based Manhattan Mobility Model

Manhattan Grid Mobility model is used to imitate the movement pattern of mobile nodes on horizontal and vertical streets defined by maps and is useful for movement generation in an urban area where various mobile devices are accessing a ubiquitous

computing service. The mobile node is encouraged to move along the grid of horizontal and vertical streets on the map whereby this model got its name “Manhattan Grid” as shown in Figure 5.3. At the intersection of a horizontal and a vertical street, the mobile node can turn left, right or head straight. The choice of movement at the intersection is probabilistic: the probability of moving on the same street is 0.5, the probability of turning left is 0.25 and the probability of turning right is 0.25. (Javadi, 2007).

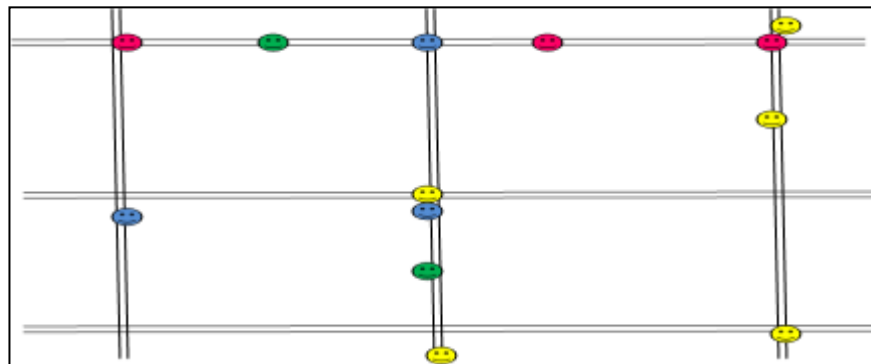


Figure 5.3. Map used in Manhattan Grid Mobility model

5.3.6. Traffic Patterns

Mobile Ad hoc routing protocols can be evaluated on basis of different application layer traffic patterns like HTTP, FTP and TELNET as discussed below (Thompson et al, 1997).

- **FTP Traffic:** It consists of two types of sessions: control sessions and data transfer session. The control session consists of a client TCP connection and its return traffic and data session consists of the actual data transfer.
- **TELNET Traffic:** It provides bidirectional text-based communication using a virtual terminal connection.
- **HTTP Traffic:** The HTTP Traffic is the data transmitted and received over the protocol between your computer and the web server.

5.4. Performance Metrics for Evaluation of Protocols

These metrics are evaluated by making use of AWK scripts in which the formulae for these metrics are coded. Various metrics considered for this study have been listed below:

- **Throughput:** The maximum allowable rate at which data can be transmitted over a link is termed as Throughput. Protocols having better throughput or packet delivery ratio are considered to be efficient protocols (Mishra et al, 2012).

- **Average End to End delay:** It is the average time required by packets to reach the destination. It considers all kinds of delay such as queuing delay, route discovery delay, interface delay, etc. It is also known as the average time between sending and successfully receiving a packet. Sometimes it is also known as path optimality (Mishra et al, 2012). Simulations are carried out to evaluate performance with respect to average end-to-end delay.

- **Packet Loss:** It is the number of packets lost per transmission from sender to receiver node. It may also be called the degree of loss of packets or datagrams while these are in transit.

- **Routing Message Overhead:** Routing message overhead is defined as the total number of control packets transmitted from source to destination. It may also be called as Control message overhead. The increase in the routing message overhead reduces the performance of the mobile ad-hoc network as it consumes some part of bandwidth available for transmission of data between the nodes (El-Nabi, 2005).

- **Packet Delivery ratio:** It is the ratio of number of transmitted packets to the number of received packets.

5.5. Chapter Summary

This chapter discusses the removal of bugs from NS-2 implementation of TORA and reasons behind poor performance of TORA protocol with rise in number of connections. The proposed technique to enhance the performance of TORA has been described and the simulation environment in which the performance will be evaluated is detailed. The various components of the simulation environment like

propagation model, MAC protocol, Interface Queue Type, mobility models, traffic patterns and performance parameters have been discussed.

CHAPTER 6

RESULTS AND DISCUSSIONS

In this chapter, the performance of DSDV and TORA protocol is evaluated under different mobility models (Random Waypoint, Manhattan Grid and Random Direction) and traffic patterns (FTP, TELNET, HTTP). The NS-2 implementation of TORA contains bugs which have been rectified to evaluate it under chosen simulation environment as discussed in section 5.1. It was then observed that the performance of TORA protocol degrades with rise in number of traffic connections. Therefore the proposed modification is performed and the performance of Enhanced-TORA is also evaluated under three mobility models and FTP traffic pattern to find out which mobility model gives best performance. The performance of Enhanced-TORA is then compared with performance of Original TORA under single and 10 traffic connections using FTP traffic under best performing mobility model.

6.1. Performance Evaluation of DSDV and TORA Protocol Under Different Mobility Models and Traffic Patterns

The performance of DSDV and TORA protocols was evaluated under the three mobility models (Random Waypoint, Random Direction and Manhattan Grid) and traffic patterns (FTP, TELNET, HTTP). To evaluate TORA, firstly the changes discussed in section 5.1 needs to be performed. The graphs of TORA and DSDV using TELNET and HTTP traffic patterns have been given in Appendix E.

6.1.1. Performance Evaluation of DSDV Protocol Under FTP Traffic and different mobility models

Experiment: Comparison of Throughput, Average end to end Delay, Routing Overhead, Packet Drop and Packet Delivery Ratio of DSDV protocol under different mobility models and FTP Traffic.

Objective: To analyze which model gives the best performance under FTP Traffic.

Results: The results have been shown in Figure 6.1. to Figure 6.5.

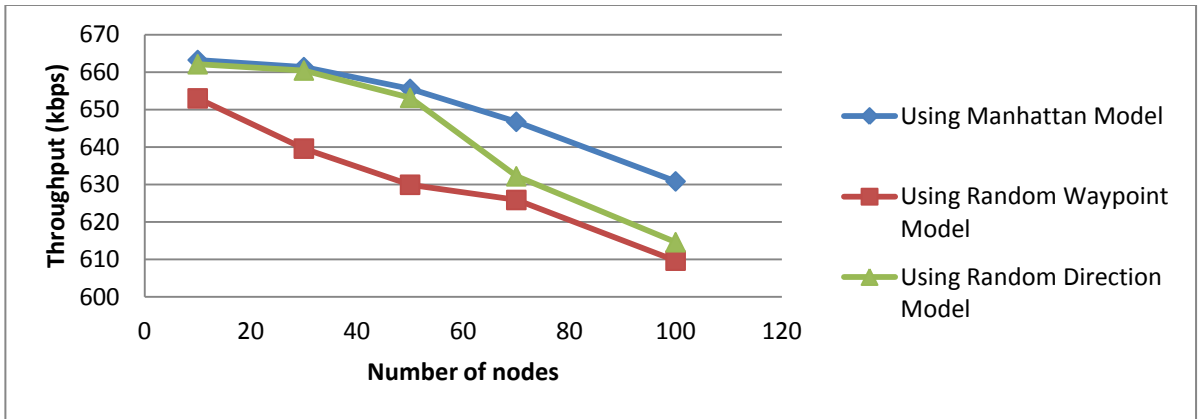


Figure 6.1. Throughput of DSDV using FTP Traffic

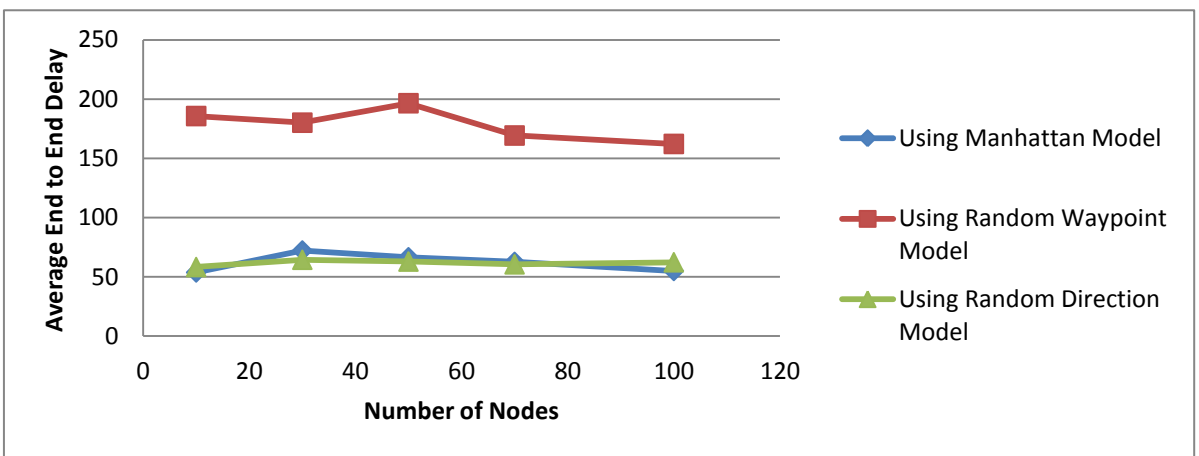


Figure. 6.2. Average End to End Delay of DSDV using FTP Traffic

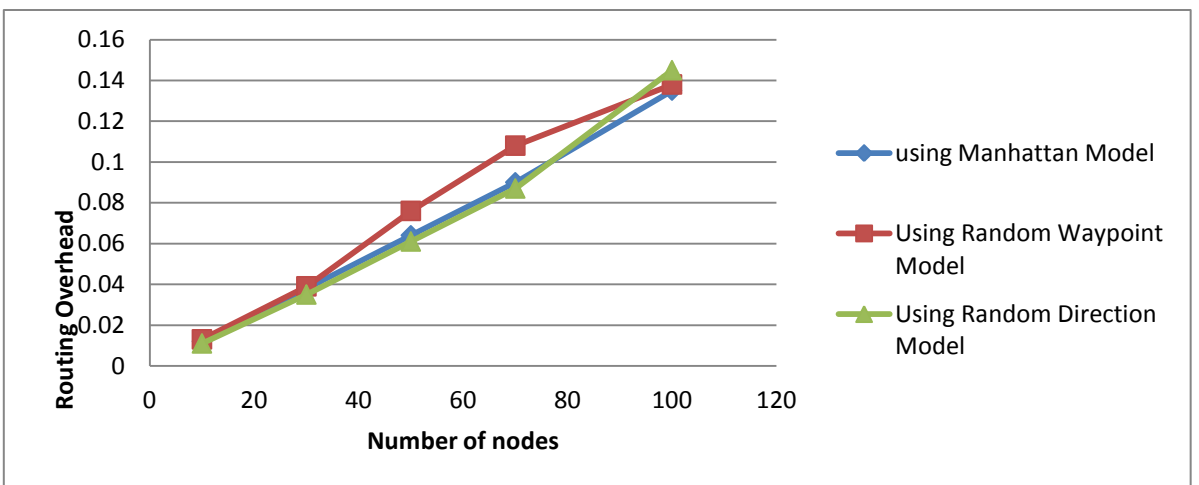


Figure 6.3. Routing Overhead of DSDV using FTP Traffic

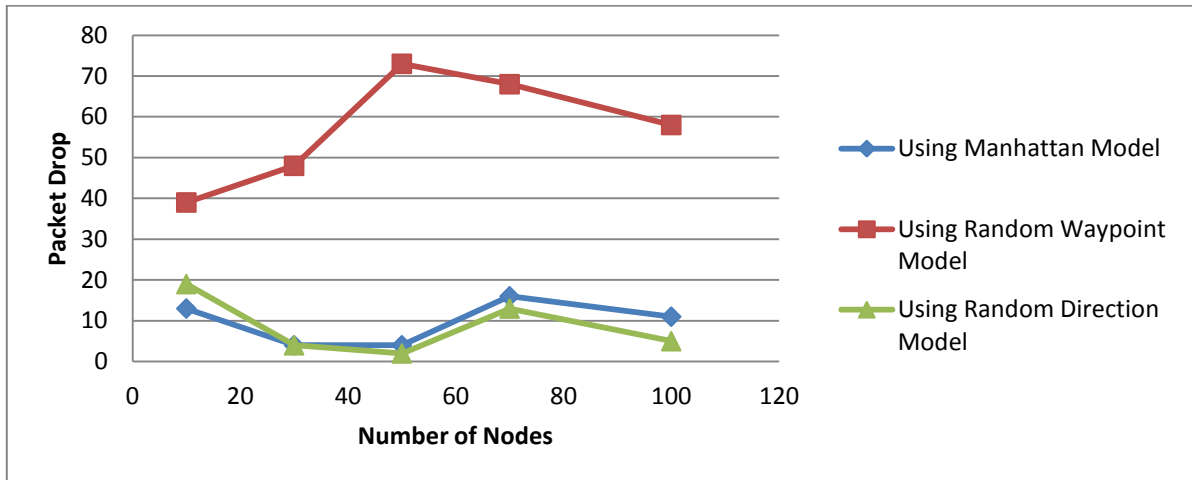


Figure 6.4. Packet Drop of DSDV using FTP Traffic

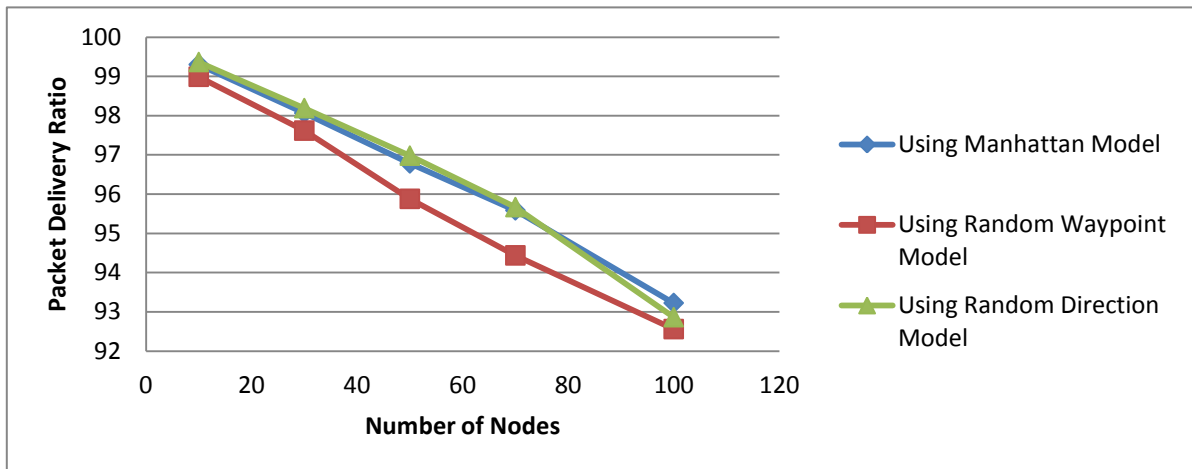


Figure. 6.5. Packet Delivery Ratio of DSDV using FTP Traffic

Analysis:

- Figure 6.1 shows that the throughput tends to decrease with the increase of network size. Manhattan model offered higher throughput than the other two models. Random waypoint performed worst in terms of throughput.
- Average end to end delay was more in case of random waypoint model as shown in Figure 6.2. Manhattan model and random direction model performed well in terms of average end to end delay as these models exhibited less delay.
- Routing overhead was more in case of random waypoint model. Here again, Random Direction and Manhattan Grid models performed well (Figure 6.3).
- More packet drop was exhibited in Random waypoint model than other models (Figure 6.4). Packet Delivery ratio was more in case of Manhattan Grid and Random

Direction models and Random waypoint, as expected, gave lesser delivery ratio (Figure 6.5).

6.1.2. Performance Evaluation of TORA Protocol Under FTP Traffic and different mobility models

Experiment: Comparison of average end to end delay and packet delivery ratio of TORA protocol under different mobility models and FTP Traffic.

Objective: To analyze which model gives the best performance under FTP Traffic.

Results: The results have been shown from Figure 6.6 to Figure 6.8.

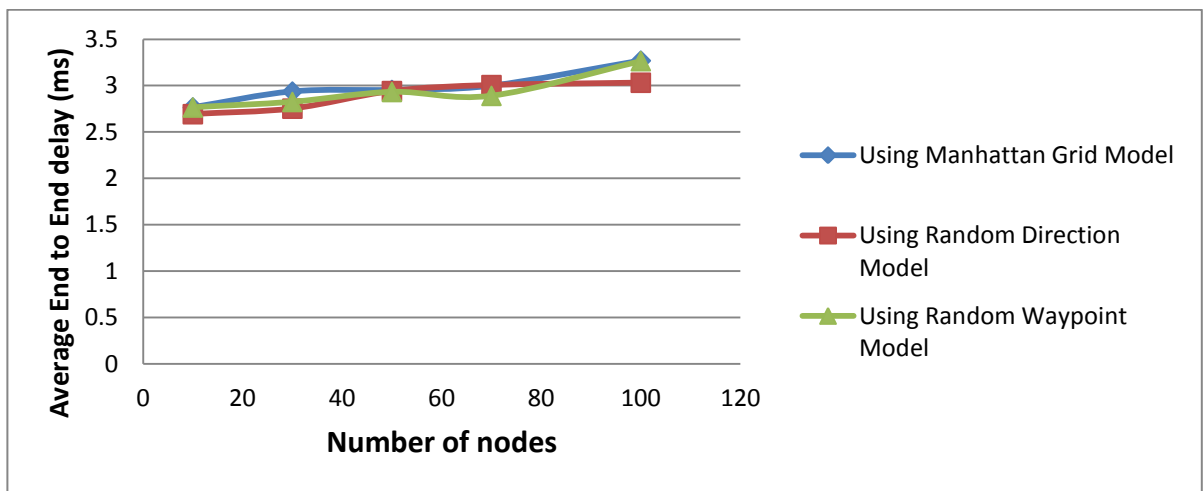


Figure 6.6. Average End to End Delay of TORA using FTP Traffic

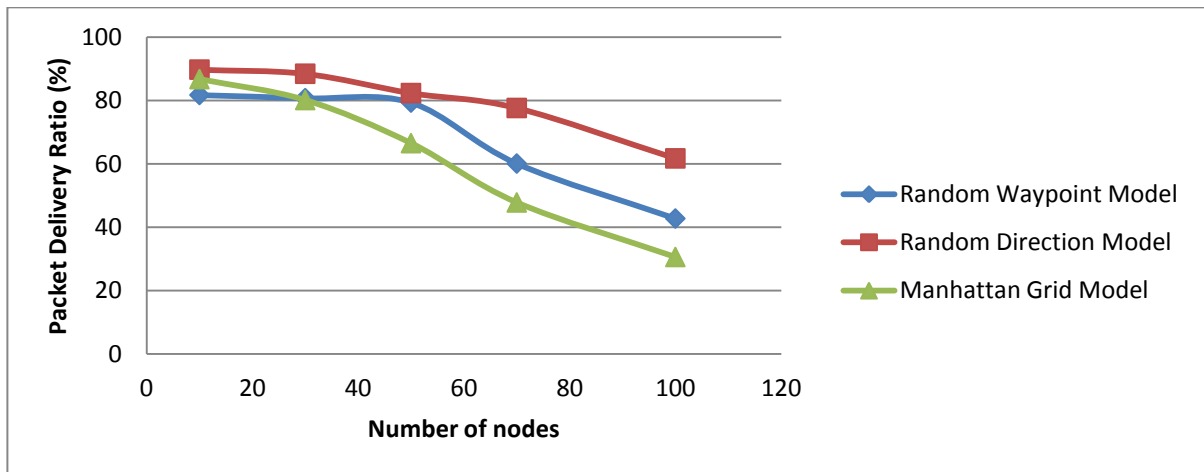


Figure 6.7. Packet Delivery Ratio of TORA using FTP Traffic

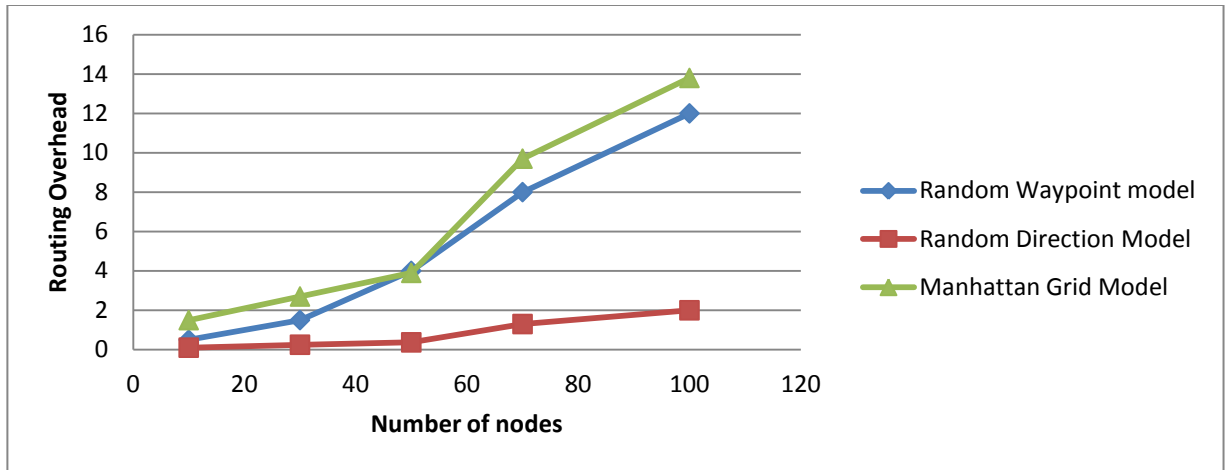


Figure 6.8. Routing Overhead of TORA using FTP Traffic

Analysis

- The average end to end delay was less under Random Direction model (Figure 6.6).
- In Figure 6.7, it can be observed that the packet Delivery ratio was better in case of Random Direction model.
- Random Direction model offered lesser Routing overhead (Figure 6.8).

6.2. Overall Results and Discussions for Evaluation of TORA and DSDV under Different Mobility Models and Traffic Patterns

The results discussed in section 6.1 and Appendix E indicate that Manhattan Grid model gave consistent performance for DSDV protocol and Random Direction model have best overall performance for TORA protocol. The best performing traffic pattern in both the cases was observed to be FTP.

The reason behind the better performance of DSDV under Manhattan Grid model is its proactive nature because of which it is best suited for predictable mobility along a city section grid rather than in random directions.

The reason behind the better performance of TORA under Random Direction mobility model is the fact that it is best suited for highly dynamic networks in which nodes move in random directions unpredictably. The results have been highlighted in Table 6.1.

Table 6.1. Best performing mobility models under different traffic patterns for DSDV and TORA

Protocol	Best Mobility Model for FTP Traffic	Best Mobility Model for TELNET Traffic	Best Mobility Model for HTTP Traffic
DSDV	Manhattan Grid Model	Manhattan Grid Model	All models gave similar performance
TORA	Random Direction Model	Random Direction Model	Random Direction Model

6.3. Investigation into Poor Performance of Original TORA with rise in Number of Connections using Random Direction Model and FTP Traffic

The performance of TORA protocol was good with single traffic connection but as the number of connections increase, its performance degrades. This problem has been detailed in section 5.1. This behavior can be observed through following results. The best performing combination of Random Direction model and FTP traffic as obtained in section 6.1.2. has been used for this analysis.

Experiment: Performance comparison of original TORA using single traffic connection and 10 traffic connections.

Objective: To identify the problem behind poor performance of TORA.

Results: The results are shown in Figure 6.9. to Figure 6.11.

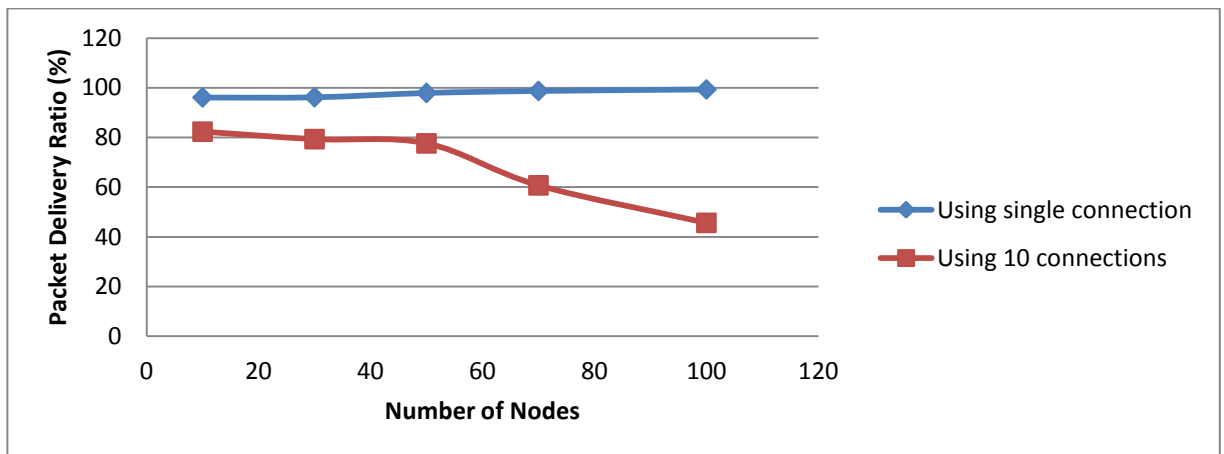


Figure 6.9. Packet Delivery Ratio of Original TORA protocol

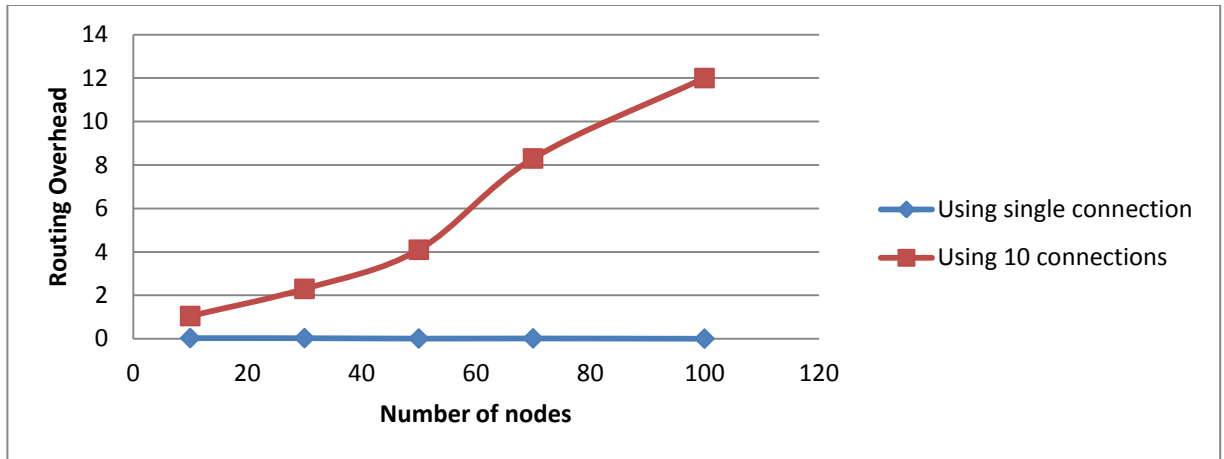


Figure 6.10. Routing Overhead of Original TORA protocol

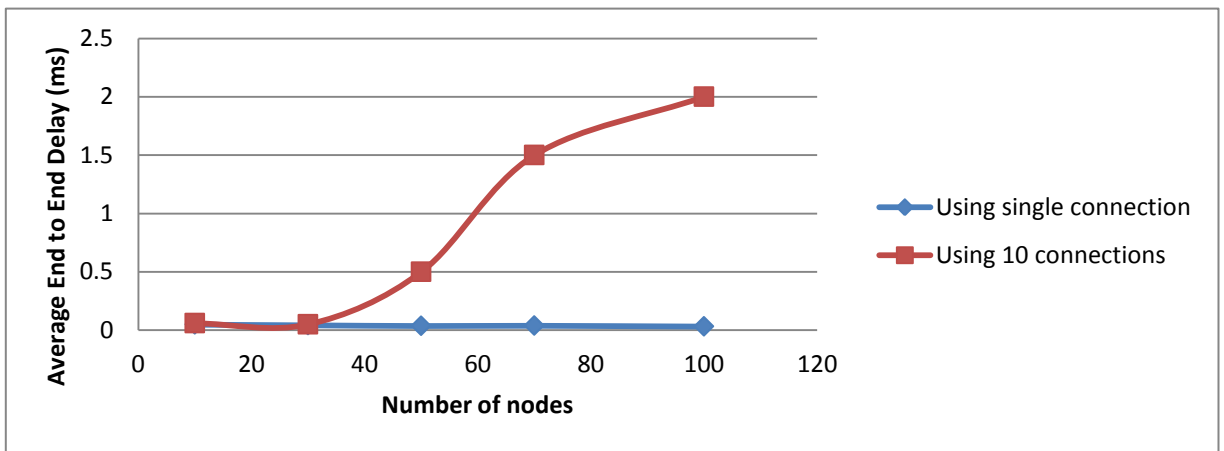


Figure 6.11. Average End to End Delay of Original TORA protocol

Analysis: TORA performs well in terms of Packet delivery ratio, routing overhead and average end to end delay using single connection. With rise in number of connections to 10, the performance of TORA starts degrading. This problem has been credited to the link status sensing mechanism of IMEP on which TORA is based for its link detection. This problem has been rectified by implementing the modification discussed in section 5.2, leading to a new version of original TORA named as Enhanced-TORA.

6.4. Performance Evaluation of Enhanced TORA under different Mobility models and FTP Traffic

After making the changes as discussed in section 5.2, Enhanced TORA was further compared under different mobility models and FTP traffic as discussed ahead.

Experiment: Calculation of Packet Delivery Ratio, Average End to End Delay and Routing Overhead of Enhanced-TORA under three chosen mobility models and FTP traffic.

Objective: To find out which mobility model gives best performance for Enhanced-TORA under 10 connections.

Results: The results have been shown from Figure 6.12 to Figure 6.14.

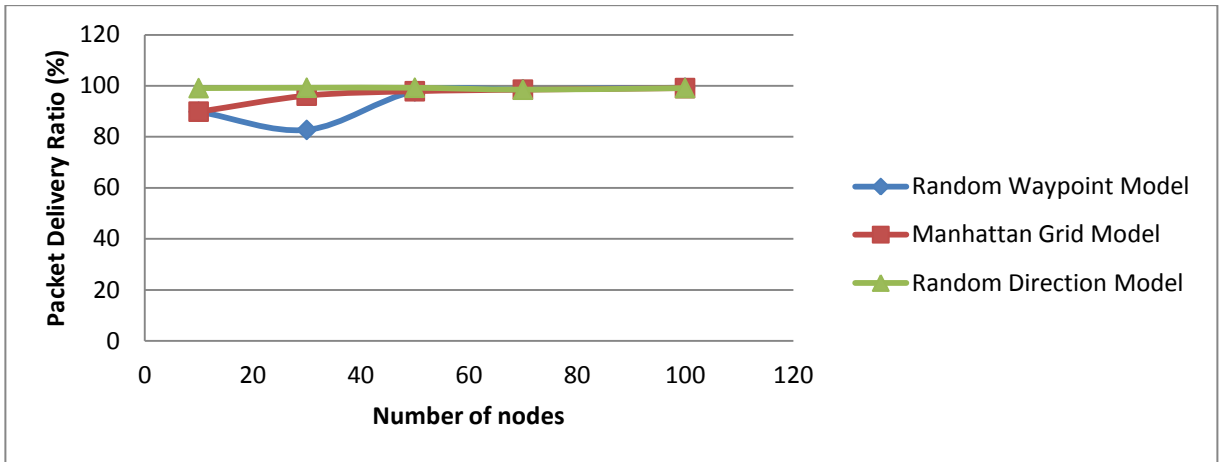


Figure 6.12. Packet Delivery Ratio of Enhanced TORA under different mobility models

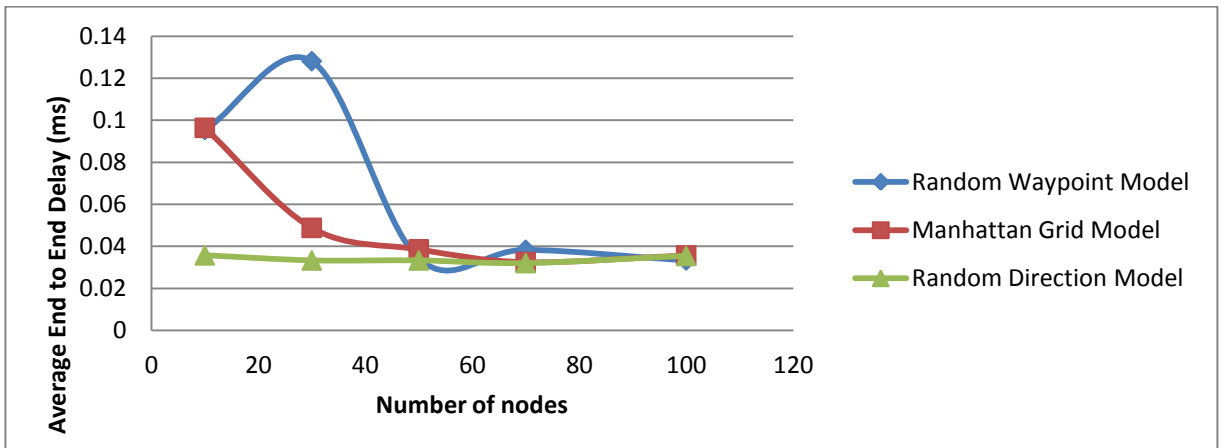


Figure 6.13. Average End to End Delay of Enhanced TORA under different mobility models

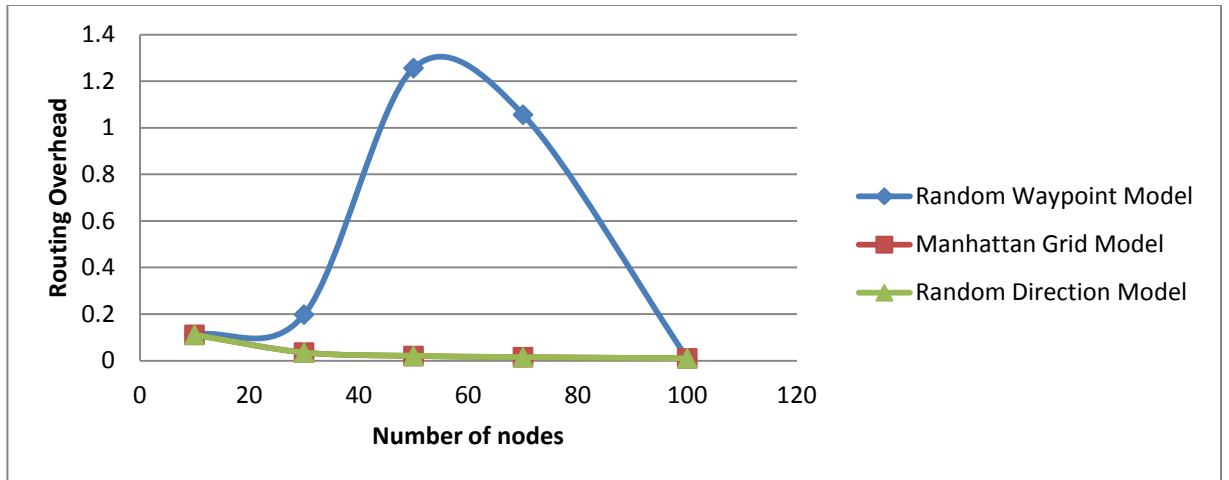


Figure 6.14. Routing Overhead of Enhanced TORA under different mobility models

Analysis: It was observed that Random Direction model gave consistent performance and was better than the other two models for Enhanced-TORA under assumed simulation environment. This highlights the use of original as well as Enhanced-TORA in highly dynamic networks where node movements are unpredictable.

6.5. Performance Comparison of Enhanced-TORA with Original TORA using 10 Number of Connections

The performance of Enhanced-TORA was compared with Original TORA with 10 number of connections so as to find out whether our enhanced protocol is better than the original one. The best performing Random Direction mobility model and FTP traffic as detailed in sections 6.1.2 and 6.4 has been used in this analysis.

Experiment: Performance comparison of Enhanced TORA with original TORA using 10 connections under Random Direction model and FTP traffic.

Objective: To rectify the problem of poor performance of TORA with rising number of connections.

Results: The results are shown in Figure 6.15. to Figure 6.17.

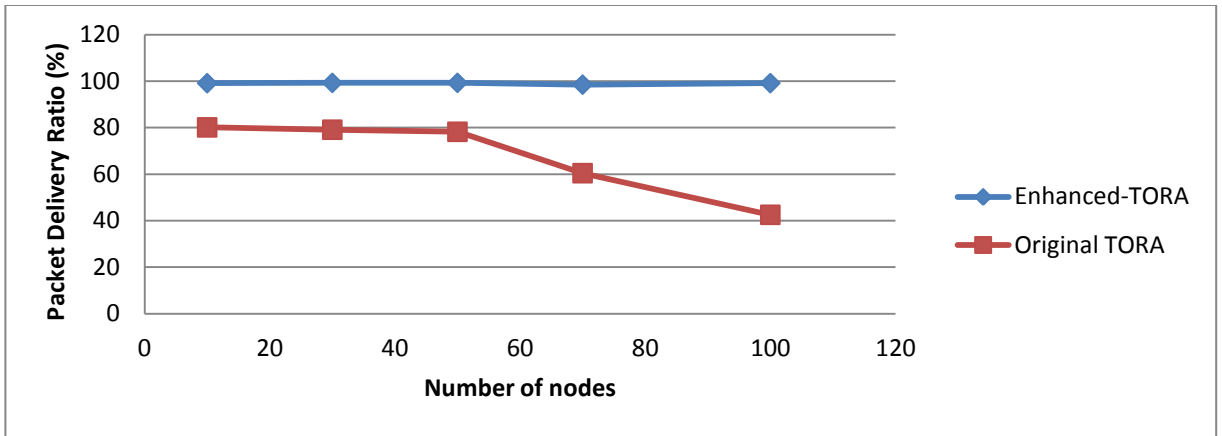


Figure 6.15. Packet Delivery Ratio Comparison of Original and Enhanced-TORA

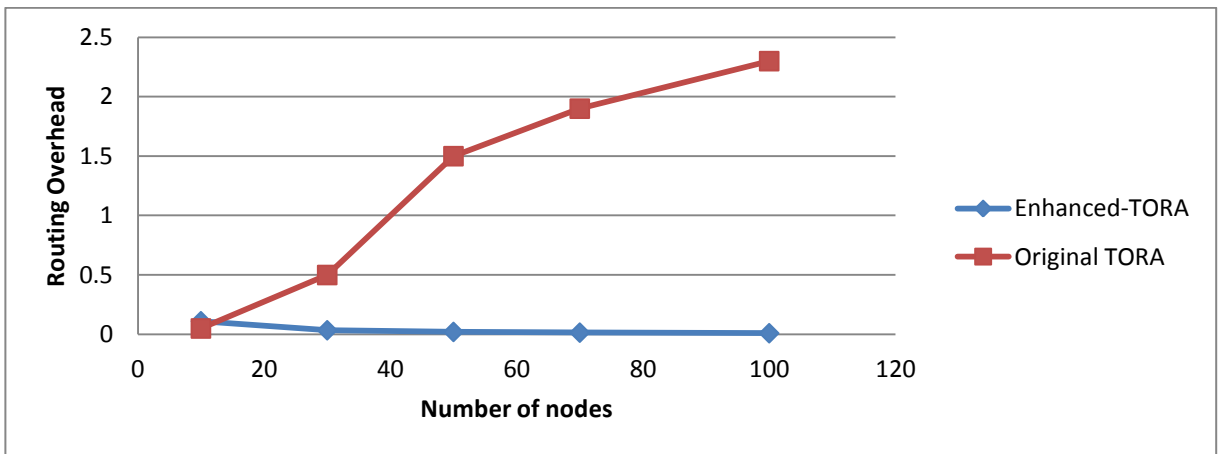


Figure 6.16. Routing Overhead Comparison of Original and Enhanced-TORA

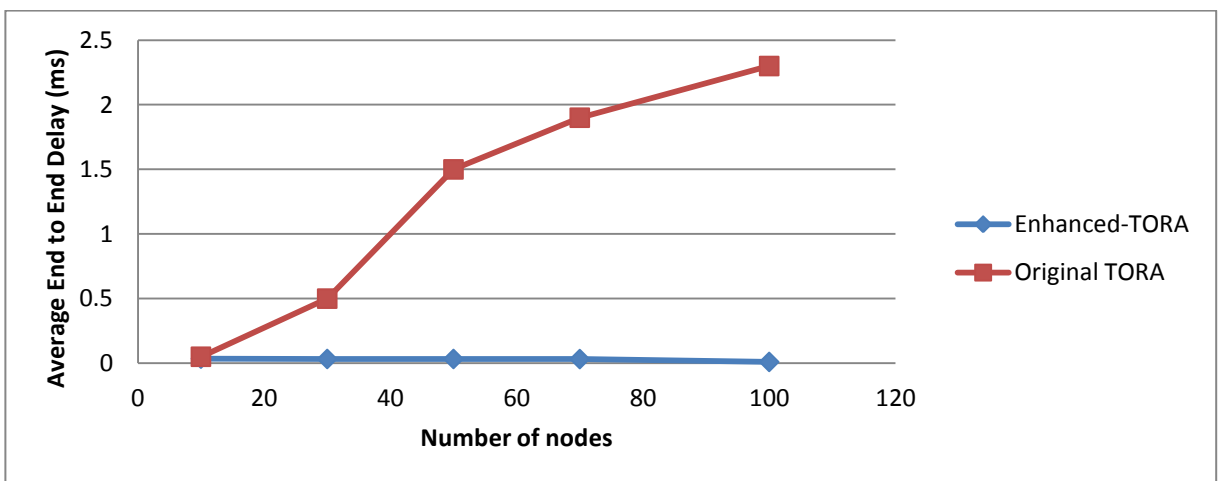


Figure 6.17. Average End to End Delay Comparison of Original and Enhanced-TORA

Analysis: The performance of Enhanced-TORA was way better than original TORA in terms of packet delivery ratio, routing overhead and average end to end delay after performing the modification under the assumed simulation environment. So it can be observed that the suggested modification results in overall improvement of TORA with rising number of connections.

6.6. Chapter Summary

In this chapter, the performance of DSDV and TORA MANET routing protocols is evaluated under different mobility models and traffic patterns. The graphical results have been analyzed and then investigations have been made into the behavior of TORA routing protocol with increase in number of connections. Then Enhanced TORA is also evaluated under different mobility models. The results of Enhanced TORA are compared with those of original TORA under best performing Random Direction model and FTP traffic combination.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

The research area of routing in MANETs is of important concern because we are living in the age of “data” which should be routed efficiently across the network. The unpredictable topology and dynamic mobile nature of MANETs make them applicable in areas like disaster management, military conflicts, medical emergencies, education etc. Therefore the mobility of nodes play a vital role and mobility models used to implement various movement scenarios of nodes in MANETs should be analyzed for various routing protocols. As discussed in (Bekmezci, I. et al, 2013), a new area of Flying Ad Hoc Networks (FANETs) have emerged out of the concepts of MANETs forming the basis of Unmanned Aerial Vehicles (UAVs) which are being widely employed in military and civilian applications, search and destroy operations, border surveillance, managing wildfire, wind estimation, disaster monitoring, remote sensing, traffic monitoring etc

The performance analysis of the MANET routing protocols DSDV and TORA was carried out in a comprehensive manner under different mobility models (Random Waypoint, Random Direction and Manhattan Grid) and traffic patterns (FTP, HTTP, TELNET) to analyze which mobility model and traffic pattern combination is best for these protocols. It is concluded that Manhattan Grid mobility model and FTP traffic gave the best performance for DSDV. Simulation of TORA was initially unsuccessful because of its error-prone implementation in NS-2. These errors were rectified and performance of TORA was evaluated under chosen simulation environment. Based on the results obtained, it is concluded that Random Direction model gave best performance for TORA under all the three traffic patterns (FTP however offers better performance than the other patterns).

The performance of TORA protocol was further investigated and degradation in performance of TORA was observed with rise in number of connections. This problem was eradicated by increasing the maximum number of OBM retransmissions in implicit link sensing mechanism of IMEP as suggested by K. H. Lim and A. Datta. Enhanced-TORA was then compared under the three chosen mobility models and it

was observed that Random Direction model gave best performance for Enhanced-TORA protocol also thereby highlighting the use of this protocol in highly dynamic networks where nodes make unpredictable movements in random directions. The performance of Enhanced-TORA was then compared with original TORA under the best performing combination of Random Direction model and FTP traffic. Results indicate that Enhanced-TORA outperformed original TORA in the chosen simulation environment.

In future, more protocols like AODV, DSR, SEAD etc can be analyzed under different mobility models and benchmarked against the protocols used in this work. With regard to TORA, the link sensing mechanism as well as other service mechanisms of IMEP can be further investigated to improve the performance of TORA.

REFERENCES

- Ahmed, G. F. Barskarb, R. and Barskarc, N. (2012). An Improved DSDV Routing Protocol for Wireless Ad Hoc Networks. Elsevier International Conference on Communication, Computing & Security. pp. 822-831. National Institute of Technology, Rourkela, India.
- Al-Maashri, A. and Ould-Khaoua, M. (2006). Performance analysis of MANET routing protocols in the presence of self-similar traffic. Proceedings of 31st IEEE Conference on Local Computer Networks. pp. 801-807. Tampa, Florida, USA.
- Antenna Basics. Banner Engineering: More Sensors More Solutions Notes < <http://info.bannerengineering.com/cs/groups/public/documents/literature/132113.pdf> >. Accessed 2013 Aug, 21.
- Aschenbruck, N. Ernst, R. Gerhards-Padilla, E. and Schwamborn, M. (2010, March). BonnMotion: A Mobility Scenario Generation and Analysis tool. Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques. pp. 51. Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering. Belgium.
- Bai, F. and Helmy, A. (2004). A survey of mobility models. *Chapter Notes in Wireless Adhoc Networks*. pp. 1-30. University of Southern California, USA.
- Balanis C. A., *Antenna Theory: Analysis and Design*, 2nd ed., John Wiley & Sons, 1997.
- Bang, A. O. and Ramteke, P. L. (2013, Sept.). MANET: History, Challenges And Applications. *International Journal of Application or Innovation in Engineering & Management*. **2(9)**: 249-251.
- Bekmezci, I., Sahingoz, O. K., & Temel, Ş. (2013). Flying ad-hoc networks (FANETs): a survey. *Ad Hoc Networks*, 11(3), 1254-1270.
- Bettstetter, C. Hartenstein, H. and Pérez-Costa, X. (2004). Stochastic properties of the random waypoint mobility model. *Wireless Networks*. **10(5)**: 555-567.

- BonnMotion A Mobility Scenario Generation and Analysis Tool. (2013, July). <sys.cs.uos.de/bonnmotion/doc/BonnMotion_Docu.pdf>. Accessed 2013 Sept, 3. University of Osnabruck, Germany.
- Camp, T. Boleng, J. and Davies, V. (2002). A survey of mobility models for ad hoc network research. *Wireless communications and mobile computing* **2(5)**: 483-502.
- Chlamtac, I., Conti, M., & Liu, J. J. N. (2003). Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks*, 1(1), 13-64.
- Cordeiro C. and Agarwal P., "Mobile Ad Hoc Networking". OBR Research Centre for Distributed and Mobile Computing, University of Cincinnati, USA.
- Corsen M. and Park V. (1997, Nov 26), An Internet MANET Encapsulation Protocol (IMEP) specification draft-ietf-manet-imep-spec-00.txt. <tools.ietf.org/html/draft-ietf-manet-imep-spec-00> Accessed 2013 Nov, 1.
- Corsen, M. S. Papademetriou, S. Papadopoulos, P. Park, V. and Qayyum, A. (1999, Aug). An Internet MANET Encapsulation Protocol (IMEP) Specification draft-ietf-manet-imep-spec-01.txt, <tools.ietf.org/html/draft-ietf-manet-imep-spec-01>. Accessed 2013 Nov, 5.
- Dhenakaran, S. and Parvathavarthini, A. (2013). An Overview of Routing Protocols in Mobile Ad-Hoc Network. *International Journal of Advanced Research in Computer Science and Software Engineering* **3(2)**: 14-30.
- Dubey, A. (2013, March). HOW TO INSTALL NS2.35 IN UBUNTU 12.04. <<http://princeabhinav.blogspot.in/>>. Accessed 2013 June, 3.
- El-Nabi, T. (2005, April). Modeling and Simulation of a Routing Protocol for Ad Hoc Networks Combining Queuing Network Analysis and Ant Colony Algorithms. PhD Thesis. University of Duisburg-Essen. Germany.
- Erlend. (2010). *Running TORA in ns-2.34 on Ubuntu 10.04*. <<http://erl1.wordpress.com/2010/08/10/running-tora-in-ns-2-34-on-ubuntu-10-04/>> Accessed 2013 July, 15.

- Foh, C. H. and Zukerman, M. (2002, Feb). Performance analysis of the IEEE 802.11 MAC protocol. Proceedings of European Wireless Conference, pp. 184-190.
- Gafni, E. M. and Bertsekas, D. P. (1981). Distributed Algorithms for generating loop-free routes in networks with frequently changing topology. *IEEE transactions on communications*. **29**:11-18.
- Gauthier, V. (2009). Mobility model in ad hoc network. <<http://www-public.it-sudparis.eu/~gauthier/MobilityModel/mobilitymodel.html>> Accessed 2013 May, 6.
- Gupta, A. K. Sadawarti, H. and Verma, A. K. (2010). Performance Analysis of AODV, DSR & TORA Routing Protocols. *International Journal of Engineering and Technology*. **2(2)**: 226-231.
- Gupta, M. and Kaushik, S. (2012). Performance Comparison of AODV, OLSR and TORA Routing Protocols for MANETS. *International Journal of Computational Engineering Research*. **2(3)**: 704-711.
- He, G. (2002). Destination-sequenced distance vector (DSDV) protocol. *Networking Laboratory Notes*, Helsinki University of Technology. Finland.
- Henderson, T. (2011) Network Components in a Mobile Node, <<http://www.isi.edu/nsnam/ns/doc/node173.html>>. Accessed 2013 Oct, 15.
- Henderson, T. (2011). Radio Propagation Models. <<http://www.isi.edu/nsnam/ns/doc/node216.html>>. Accessed 2013 Dec, 9.
- Hoebeke, J. Moerman, I. Dhoedt, B. and Demeester, P. An Overview of Mobile Ad Hoc Networks: Applications and Challenges. Ghent University, Belgium.
- Hoffman, A. and Romanoff, E. (2012). Ad-Hoc Routing Protocols. <www.cs.rit.edu/~ear7631/distributed/presentation2.pdf>. Accessed 2013 March, 1.
- Hyytiä, E. (2005). Random Waypoint Model. <<http://www.netlab.tkk.fi/~esa/java/rwp/rwp-model.html>>. Accessed 2013 April, 5.
- Ilyas M. (2003). *The Handbook of Ad Hoc Wireless Networks*, pp. 30-32. CRC Press, USA,

- Islam, K. B. (2011). *Simulating TORA routing protocol in NS-2.34 on Ubuntu 10.04*. <<http://khaledben.wordpress.com/2011/02/27/simulating-tora-routing-protocol-in-ns-2-34-on-ubuntu-10-04/>>. Accessed 2013 Nov, 9.
- Issariyakul, T. and Hossain, E. (2012). *An introduction to network simulator NS2*. Springer.
- Javadi, M.M. (2007). *Mobility Simulator Manhattan Grid Mobility Model*. <<http://www.masoudmoshref.com/old/myworks/documentpages/mobisim/manhattan.html>>. Accessed 2013 Nov, 20.
- Khiavi, M. V. Jamali, S. and Gudakahriz, S. J. (2012). Performance Comparison of AODV, DSDV, DSR and TORA Routing Protocols in MANETs. *International Research Journal of Applied and Basic Sciences* **3(7)**: 1429-1436.
- Kulenica, E. and Pham, C. (2002). *802.11 MAC Protocol*. <ipsit.bu.edu/sc546/sc546Fall2002/ieee80211/Presentation.ppt>. Accessed 2013 Aug, 23.
- Kumar, M. Sharma, S.C. and Chandras. (2012). Effective Analysis of Different Parameters in Ad hoc Network for Different Protocols. *International Journal of Smart Sensors and Ad Hoc Networks*. **1(3)**: 248-273.
- Kumar, R. and Singh, P. (2013, Aug). Performance analysis of AODV, TORA, OLSR and DSDV Routing Protocols using NS2 Simulation. *International Journal of Innovative Research in Science, Engineering and Technology*, **2(8)**: 231-253.
- Kumari, S. Maakar, S. Kumar, S. and Rathy, R. K. (2011). Traffic Pattern Based Performance Comparison of AODV, DSDV & OLSR MANET Routing Protocols Using Freeway Mobility Model. *International Journal of Computer Science and Information Technologies*. **2(4)**: 1606-1611.
- Lesiuk, B. C. (1998). Routing in Ad Hoc Networks of Mobile Hosts. Mech590 Report, University of Victoria, British Columbia.
- Lim, K. H. and Datta, A. (2012, April). An In-depth Analysis of the Effects of IMEP on TORA Protocol. Proceedings of IEEE Wireless Communications and Networking Conference. pp. 3051-3056. Shanghai, China.

- Loo, J., Mauri, J. L., & Ortiz, J. H. (Eds.). (2012). *Mobile Ad Hoc Networks: Current Status and Future Trends*. CRC Press. USA.
- Manoj, K. Parmanand, S. C. and Singh, S. P. (2009). Performance of QoS Parameter in Wireless Ad hoc Network (IEEE 802.11 b). Proceedings of the World Congress on Engineering and Computer Science. **1**: 20-22. San Francisco, USA.
- Meenaghan, P. and Delaney, D. (2004). An introduction to NS, Nam and OTcl scripting. *National University of Ireland*.
- Mishra, N. Maurya, N. and Gaur, N. (2012, Oct). Performance Evaluation of Mobile Ad Hoc Network Extension to Routing Protocols Simulation Based. *International Journal of Computer Science & Information Technology*. **2(10)**: 2231-2471.
- Mishra, P. (1999), *Routing Protocols for Ad Hoc Mobile Wireless Networks*. <http://www.cse.wustl.edu/~jain/cis788-99/ftp/adhoc_routing/>. Accessed 2014 Jan, 13.
- Narra, H. Cheng, Y. Cetinkaya, E. K. Rohrer, J. P. and Sterbenz, J. P. (2011, March). Destination-sequenced distance vector (DSDV) routing protocol implementation in ns-3. Proceedings of the 4th International Conference on Simulation Tools and Techniques. pp. 439-446. Barcelona, Spain.
- Pal, A. Singh, J. P. Dutta, P. Basu, P. and Basu, D. (2011, July). A study on the effect of traffic patterns on Routing protocols in Ad-hoc network following RPGM Mobility model. Proceedings of IEEE International Conference on Signal Processing, Communication, Computing and Networking Technologies. pp. 233-237. Noorul Islam Centre for Higher Education Thuckalay, Tamil Nadu, India.
- Panda, S. and Mohapatra, R. (2009). Implementation and Comparison of Mobility Models In Ns-2. B.Tech Thesis. National Institute of Technology, Rourkela, India.
- Park, V. D. and Corsen, M. S. (1997) A highly adaptive distributed routing algorithm for mobile wireless networks. Proceedings of 16th IEEE Infocom Conference Computer Communications. Vol. 3. pp. 1405-1413. Kobe, Japan.

- Patil, A. P. Sambaturu, N. and Chunhaviriyakul, K. (2009). Convergence Time Evaluation Of Algorithms in MANETs. *International Journal of Computer Science and Information Security*, **5(1)**: 144-149.
- Patil, V.P. (2012). Effect of Traffic Pattern on Packet Delivery Ratio in Reactive Routing Protocol of Manet. *International Organization of Scientific Research- Journal of Electronics and Communication Engineering (IOSRJECE)*, **2(2)**: 30-35
- Perkins, C. E. (2000) *Ad Hoc Networking*. Addison Wesley. NJ.
- Perkins, C. E. and Bhagwat, P. (1994, Sept.). Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers. Proceedings of ACM SIGCOMM Conference on Communications Architectures, Protocols and Applications. pp. 234-244. London.
- Pucha, H. Das, S. M. and Hu, Y. C. (2007). The performance impact of traffic patterns on routing protocols in mobile ad hoc networks. *Computer Networks*. **51(12)**: 355-361.
- Radwan, A. A. Mahmoud, T. M. and Houssein, E. H. (2011). Performance Measurement of Some Mobile Ad Hoc Network Routing Protocols. *International Journal of Computer Science Issues*. **8(1)**: 107-112.
- Rajeswari, S. and Venkataramani, Y. (2011). Traffic Performance Analysis of MANET Routing Protocol. *International Journal of Distributed and Parallel Systems*, **2(3)**: 64-74.
- Roy, R. R. (2011). *Handbook of mobile ad hoc networks for mobility models*. Springer.
- Royer, E. M. Lee, S. J. and Perkins, C. E. (2000). The effects of MAC protocols on ad hoc network communication. In *Wireless Communications and Networking Conference*, vol. 2, pp. 543-548. Chicago, USA.
- Royer, E. M., & Toh, C. K. (1999). A review of current routing protocols for ad hoc mobile wireless networks. *Personal Communications, IEEE*, **6(2)**, 46-55.

- Sahraei, S. (2006, March). *TORA modifications in ns2.29*.
<<http://mailman.isi.edu/pipermail/ns-developers/2006-March/002250.html>>
Accessed 2013 Dec, 21.
- Sarkar, S. K. Basavaraju, T. and Puttamadappa, C. (2007). *Ad hoc mobile wireless networks: principles, protocols and applications*: CRC Press. New York, USA.
- Shah S. (2001). CAD-HOC: A CAD LIKE TOOL For generating mobility benchmarks in ad-hoc networks. M.S Thesis. University of Florida, USA.
- Singh, D. Goel, R. and Niwas, R. (2013, Oct). Impact of CBR and FTP Traffic Patterns on the Performance of AODV, DSR and WRP Routing Protocols in MANET. *International Journal of Innovations in Engineering and Technology*. **3(1)**: 306-314.
- Soujanya, B. Sitamahalakshmi, T. and Divakar, C. (2011). Study of Routing Protocols in Mobile ad-hoc networks. *International Journal of Engineering Science and Technology*. **3(4)**: 2622-2631.
- Thompson, K., Miller, G. J., & Wilder, R. (1997). Wide-area Internet traffic patterns and characteristics. *IEEE Network*. **11(6)**: 10-23.
- Tyagi, S. S. and Chauhan, R. K. (2010). Performance Analysis of Proactive and Reactive Routing Protocols for Ad Hoc Networks. *International Journal of Computer Applications*. **1(14)**: 27-30.
- Xu, S. and Saadawi, T. (2001, June). Does the IEE 802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Networks?. *IEEE Communications Magazine-Challenges in Mobile Ad Hoc Networking*. **39(6)**: 130-137. New York, USA.
- Zubairi J. A. (2009). Applications of Modern High Performance Networks, pp. 23. Bentham Science Publishers. USA.

APPENDIX A

Simulation of DSDV, TORA and Enhanced-TORA

A.1. DSDV Performance Evaluation: There are three categories for simulation of DSDV routing protocol. The protocol was evaluated under three mobility models and three traffic types by using various performance metrics by varying the network size. A total of 45 simulations were performed.

Table A.1. Simulation scenario for DSDV protocol

Mobility Model	Traffic Type	Number of Nodes	Performance Metrics
Random Waypoint Model	FTP	10, 30, 50, 70, 100	Throughput, Average End to End Delay, Routing overhead, Packet Loss and Packet Delivery Ratio
Random Direction Model	TELNET		
Manhattan Grid Model	HTTP		

A.2. Performance Evaluation of TORA: The code of TORA was error-prone. After solving the errors as discussed in section 5.1, the simulation of TORA also involved 45 simulations.

Table A.2. Simulation scenario for TORA protocol

Mobility Model	Traffic Type	Number of Nodes	Performance Metrics
Random Waypoint Model	FTP	10, 30, 50, 70, 100	Average End to End Delay, Routing Overhead and Packet Delivery Ratio
Random Direction Model	TELNET		
Manhattan Grid Model	HTTP		

A.3. Performance Evaluation of Enhanced-TORA: The Enhanced-TORA protocol was simulated according to Table A.3. involving 15 simulations.

Table A.3. Simulation scenario for Enhanced TORA protocol

Mobility Model	Traffic Type	Number of Nodes	Performance Metrics
Manhattan Grid Model Random Waypoint Model Random Direction Model	FTP	10, 30, 50, 70, 100	Average End to End Delay, Routing Overhead and Packet Delivery Ratio

APPENDIX B

Traffic Sources (FTP, TELNET, HTTP)

Following codes have been incorporated in the Tcl scripts of both DSDV and TORA protocols for different scenarios as mentioned in Appendix A.

B.1. Code incorporated in Tcl Script for FTP Traffic:

```
# Set a TCP connection between node_(0) and node_(1)
set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]
$ns attach-agent $node_(0) $tcp
$ns attach-agent $node_(1) $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns at 10.0 "$ftp start"
```

B.2. Code incorporated in Tcl Script for TELNET Traffic:

```
# Set a TCP connection between node_(0) and node_(1)
set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]
$ns attach-agent $node_(0) $tcp
$ns attach-agent $node_(1) $sink
$ns connect $tcp $sink
set telnet [new Application/telnet]
$telnet attach-agent $tcp
$ns at 10.0 "$telnet start"
```

B.3. Code incorporated in Tcl Script for HTTP Traffic:

```
set pgp [new PagePool/Math]
set server [new Http/Server $ns $node_(0)]
$server set-page-generator $pgp
set client [new Http/Client $ns $node_(1)]
$client set-page-generator $pgp
set startTime 1.0
$ns at $startTime "start-connection"
$client connect $server
```

APPENDIX C

Scenarios for Different Mobility Models

The mobility models- Random Waypoint, Random Direction and Manhattan Grid have been used to generate movements for various nodes. All the nodes were kept mobile. Following codes were incorporated in the Tcl Scripts for movement generation by using BonnMotion 2.1a tool.

C.1. Movement Generation for a node using Manhattan Grid Model

```
$node_(0) set X_ 170.0
$node_(0) set Y_ 67.6032544230527
$ns_ at 0.0 "$node_(0) setdest 170.0 50.0 0.8112658814001998"
$ns_ at 21.698502089931935 "$node_(0) setdest 190.0 50.0 0.8112658814001998"
$ns_ at 46.35133226378457 "$node_(0) setdest 190.0 10.0 0.8112658814001998"
$ns_ at 95.65699261148984 "$node_(0) setdest 199.50715585730904 10.0
0.811265881400204"
$ns_ at 107.37590755081919 "$node_(0) setdest 209.50715585730904 10.0
0.7412048313403017"
$ns_ at 120.86745479548199 "$node_(0) setdest 210.0 10.0 1.024448793063696"
$ns_ at 121.3485370575022 "$node_(0) setdest 210.0 29.507155857309044
1.0244487930637616"
$ns_ at 140.3901484971234 "$node_(0) setdest 210.0 30.0 1.1013012920292913"
$ns_ at 140.8376592248942 "$node_(0) setdest 205.49284414269096 30.0
1.1013012920296217"
$ns_ at 144.93023215650555 "$node_(0) setdest 200.49284414269096 30.0
1.0389897452617165"
$ns_ at 149.74259919169754 "$node_(0) setdest 200.267242390331 30.0
0.8764609320684494"
```

C.2. Movement generation for a node using Random Waypoint Model:

\$node_(0) set X_ 102.61322687349079

\$node_(0) set Y_ 91.89862982365727

\$ns_ at 0.0 "\$node_(0) setdest 131.83413572889532 32.05962239006789
1.415951124770891"

\$ns_ at 47.03026355298198 "\$node_(0) setdest 131.83413572889532
32.05962239006789 0.0"

\$ns_ at 104.05929680855297 "\$node_(0) setdest 147.4427107580999
54.87022371221987 0.6016378564186469"

C.3. Movement generation for a node using Random Direction Model

\$node_(0) set X_ 41.42070418057756

\$node_(0) set Y_ 10.0

\$ns_ at 0.0 "\$node_(0) setdest 41.42070418057756 10.0 0.0"

\$ns_ at 25.200466374290954 "\$node_(0) setdest 90.34431286816832
56.041948827793036 0.5383168990620121"

APPENDIX D

AWK Scripts

The trace files generated after running the simulation are analysed with help of AWK scripts. Various metrics were calculated using the following AWK scripts.

D.1. Average End to End Delay and Packet Delivery Ratio AWK script

```
BEGIN {
seqno = -1; droppedPackets = 0; receivedPackets = 0; count = 0; }
#packet delivery ratio
if($4 == "AGT" && $1 == "s" && seqno < $6) { seqno = $6;}
else if(($4 == "AGT") && ($1 == "r")) { receivedPackets++;}
else if ($1 == "D" && $7 == "tcp" && $8 > 512){ droppedPackets++;}
#end-to-end delay
if($4 == "AGT" && $1 == "s") { start_time[$6] = $2; } else if(($7 == "tcp") && ($1 ==
"r")) end_time[$6] = $2; }
else if($1 == "D" && $7 == "cbr") { end_time[$6] = -1; } } END {
for(i=0; i<=seqno; i++) { if(end_time[i] > 0) { delay[i] = end_time[i] - start_time[i];
count++;}
else { delay[i] = -1;} } for(i=0; i<count; i++) {
if(delay[i] > 0) { n_to_n_delay = n_to_n_delay + delay[i]; }}
n_to_n_delay = n_to_n_delay/count; print "\n"; print "Packets Generated = " seqno+1;
print "Packets Received = " receivedPackets;
print "Packet Delivery Ratio = " receivedPackets/(seqno+1)*100 "%";
print "Packets Dropped = " droppedPackets;
print "Average End-to-End Delay = " n_to_n_delay * 1000 " ms"; print "\n"; }
```

D.2. RoutingOverhead.awk

```
BEGIN { recvd = 0; # to calculate total number of data packets received
rt_pkts = 0; # to calculate total number of routing packets received } {
# Check if it is a data packet
```

```

if (( $1 == "r" ) && ( $7 == "cbr" || $7 == "tcp" ) && ( $4 == "AGT" )) recvd++;
# Check if it is a routing packet
if (($1 == "s" || $1 == "f") && $4 == "RTR" && ($7 == "AODV" || $7 == "message" || $7
=="DSR" || $7 == "OLSR")) rt_pkts++;}END {
printf("\n"); printf("Normalized Routing Load = %.3f\n", rt_pkts/recvd); printf("\n");}

```

D.3. Throughput.awk

```

BEGIN { recvdSize = 0; startTime = 150; stopTime = 0 } { event = $1; time = $2;
node_id = $3; pkt_size = $8; level = $4
# Store start time
if (level == "AGT" && event == "s" && pkt_size >= 512) { if (time < startTime) {
startTime = time } }
# Update total received packets' size and store packets arrival time
if (level == "AGT" && event == "r" && pkt_size >= 512) {if (time > stopTime) {
stopTime = time}
# Rip off the header
hdr_size = pkt_size % 512; pkt_size -= hdr_size
# Store received packet's size
recvdSize += pkt_size } } END {
printf("Average          Throughput[kbps]          =          %.2f\t\t
StartTime=%.2f\tStopTime=%.2f\n", (recvdSize/(stopTime-
startTime))*(8/1000), startTime, stopTime) }

```

APPENDIX E

Graphical Results for DSDV and TORA under TELNET and HTTP Traffic

E.1. Performance Evaluation of DSDV Protocol Using TELNET Traffic

Experiment: Comparison of Throughput, Average End to End Delay, Packet Delivery Ratio and Routing Overhead of DSDV protocol under different mobility models and TELNET Traffic.

Objective: To analyze which model gives the best performance under TELNET Traffic.

Results: The results have been shown in Figure E.1. to Figure E.4.

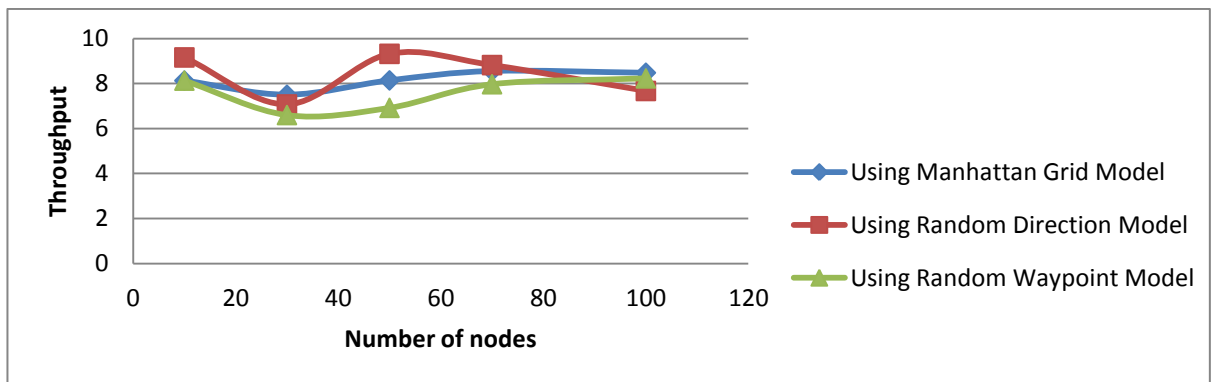


Figure. E.1. Throughput of DSDV using TELNET Traffic

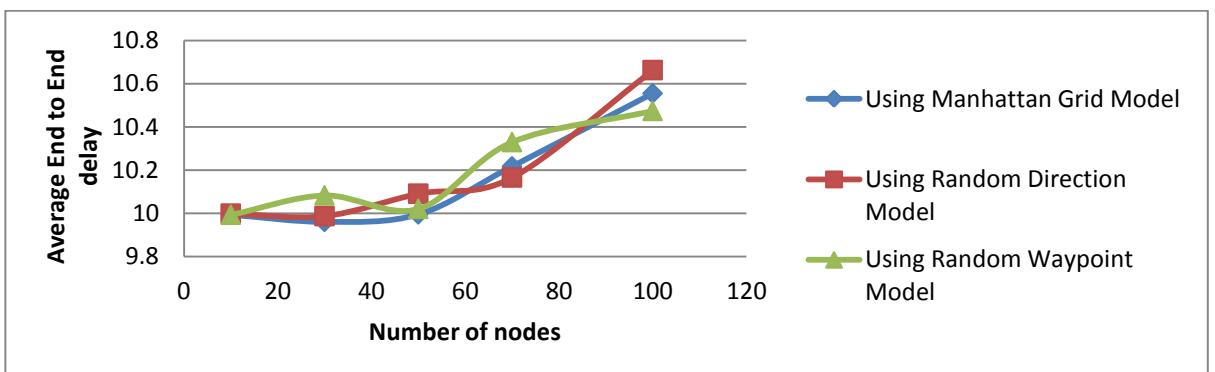


Figure. E.2. Average End to End Delay of DSDV using TELNET Traffic

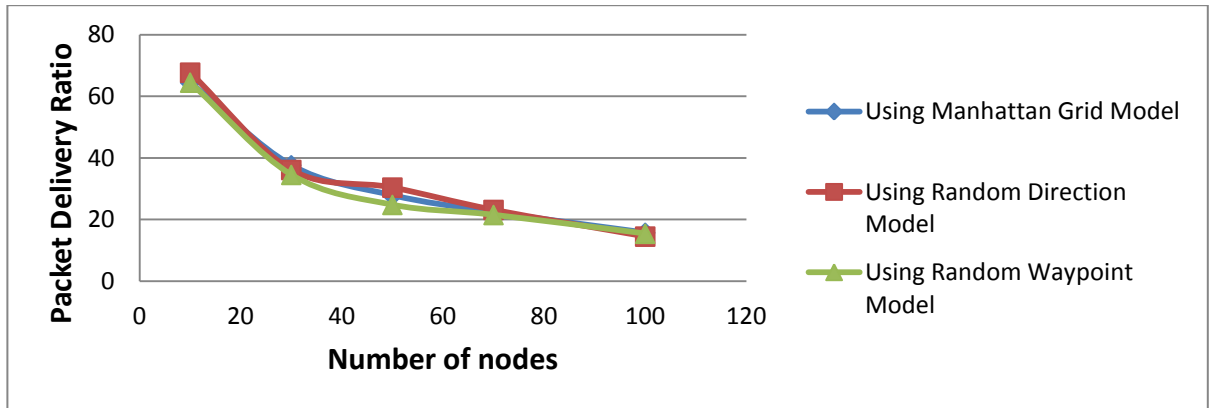


Figure E.3. Packet Delivery Ratio of DSDV using TELNET Traffic

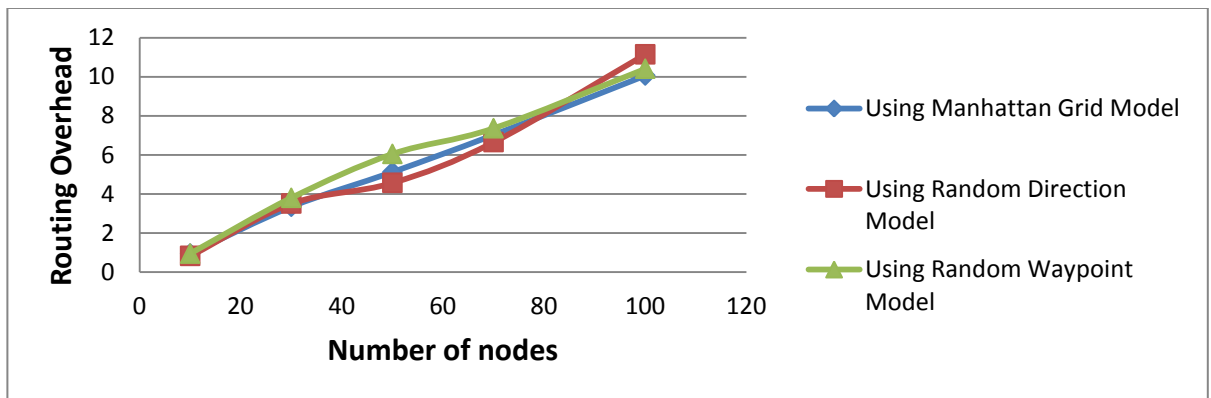


Figure E.4. Routing Overhead of DSDV using TELNET Traffic

Analysis

- As shown in Figure E.1., initially the throughput offered by Random Direction model is more but as the network size grows throughput starts declining. Whereas in Manhattan Grid and Random waypoint models offer consistent throughput.
- From Figure E.2., it is analyzed that the average end to end delay rises with increase of network size. However, Manhattan model offers better performance than the other two models.
- Packet Delivery ratio decreases with increase of network size (Figure E.3.).
- From Figure E.4. it can be observed that routing overhead also rises with increase of network size. Manhattan model offers lesser overhead as compared with other models.

- It has been observed that for TELNET traffic, all the three models gave consistent performance.

E.2. Performance Evaluation of DSDV Protocol Using HTTP Traffic

Experiment: Comparison of average end to end delay, routing overhead and packet delivery ratio of DSDV protocol under different mobility models and HTTP Traffic.

Objective: To analyze which model gives the best performance under HTTP Traffic.

Results: The results have been shown in Figure E.5. to Figure E.7.

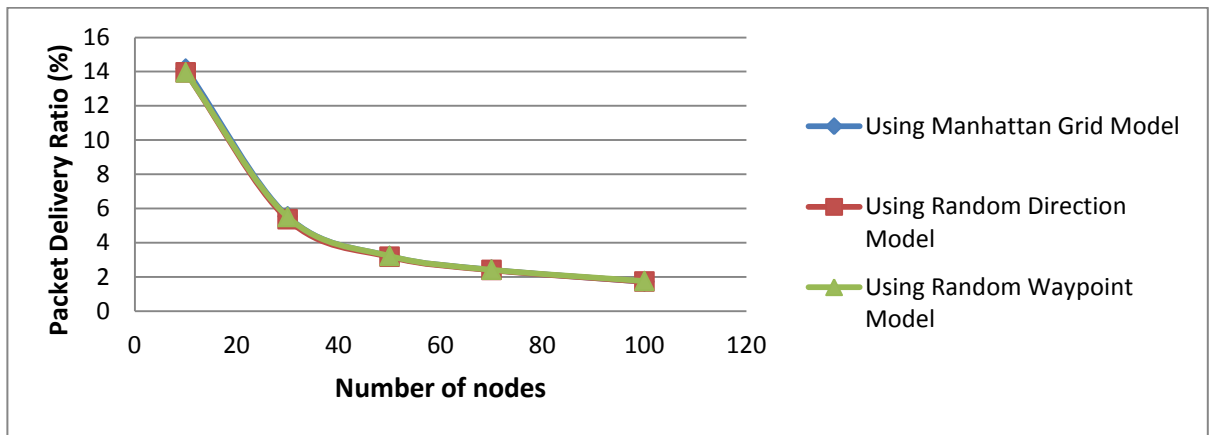


Figure E.5. Packet Delivery Ratio of DSDV using HTTP Traffic

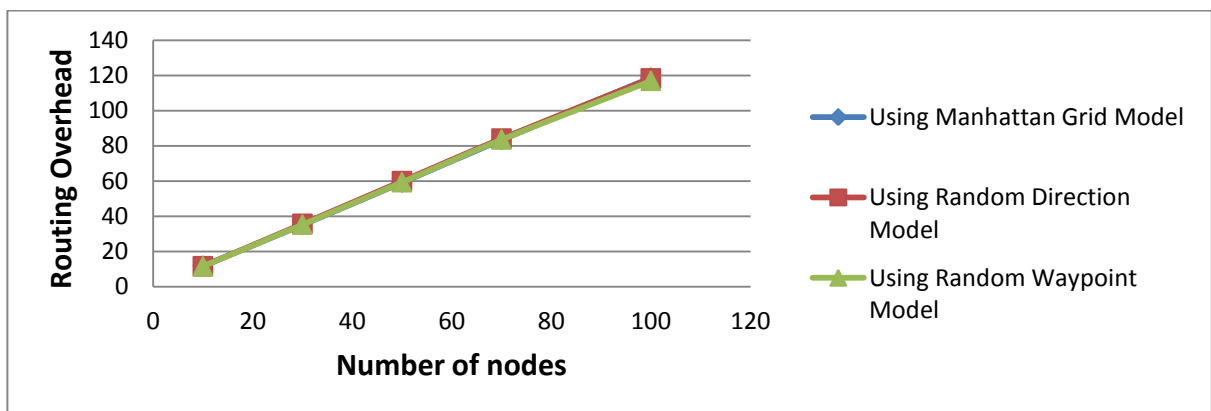


Figure E.6. Routing Overhead of DSDV using HTTP Traffic

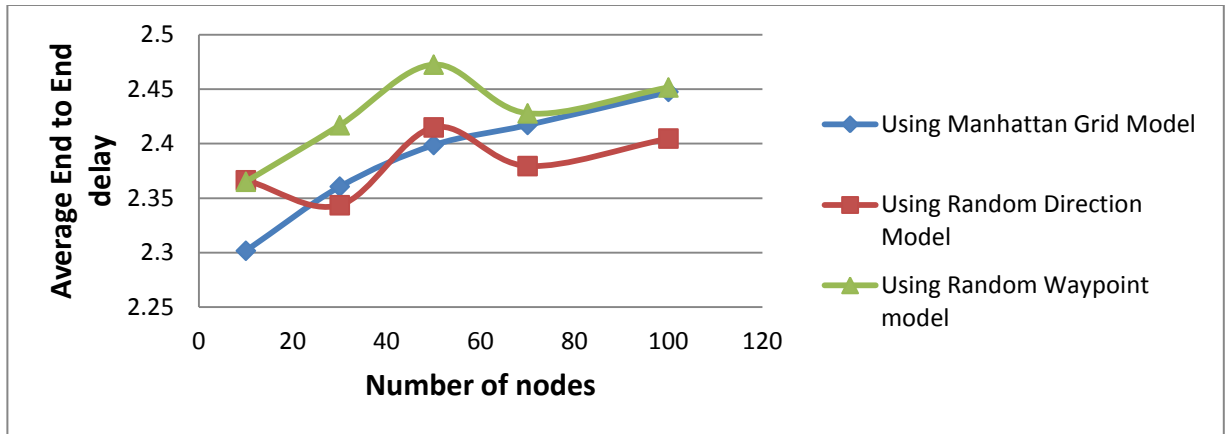


Figure E.7. Average End to End Delay of DSDV using HTTP Traffic

Analysis

- As depicted in Figure E.5., the packet delivery ratio decreases with increase of network size. And it was seen that for HTTP traffic, the three models performed similarly.
- Routing overhead increases with growth of network size. And the three models performed in similar manner (Figure E.6.).
- Random waypoint model performed the worst in this case. Manhattan model gave the best performance and random direction model showed fluctuating behavior (Figure E.7.).
- It has been observed that Manhattan Grid model offered better performance although the performance offered by all three models was quite similar for HTTP traffic.

E.3. Performance Evaluation of TORA Routing Protocol Using TELNET Traffic

Experiment: Comparison of average end to end delay, packet delivery ratio and routing overhead of TORA protocol under different mobility models and TELNET Traffic.

Objective: To analyze which model gives the best performance under TELNET Traffic.

Results: The results have been shown in Figure E.7 to Figure E.9.

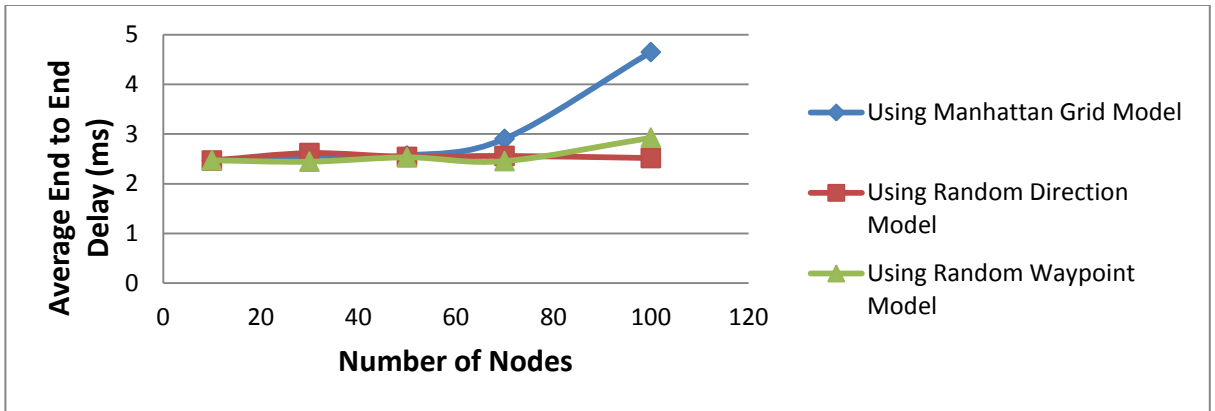


Figure E.7. Average End to End Delay of TORA using TELNET Traffic

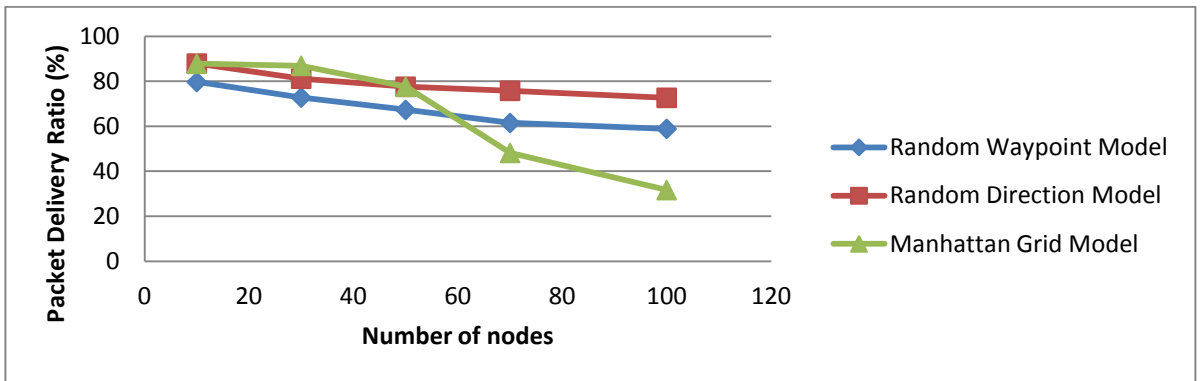


Figure E.8. Packet Delivery Ratio of TORA using TELNET Traffic

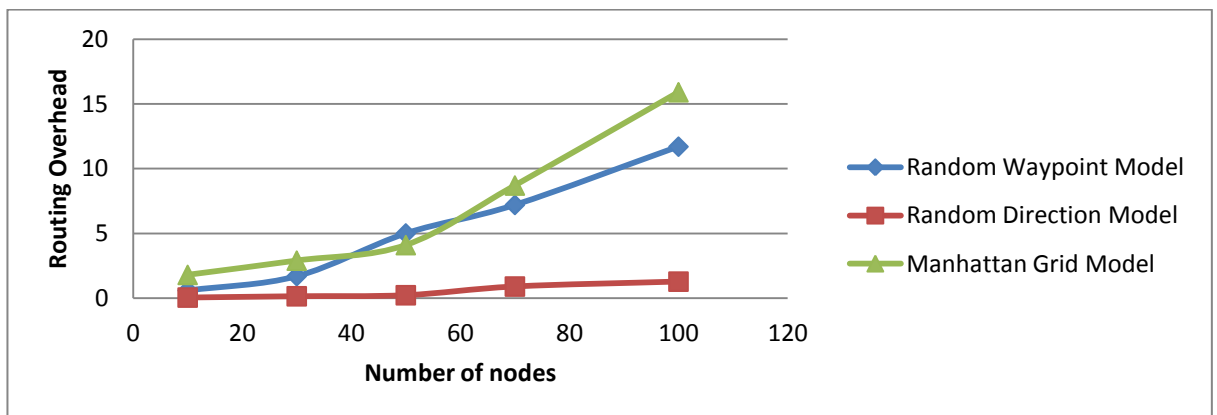


Figure E.9. Routing Overhead of TORA using TELNET Traffic

Analysis:

- Figure E.7. shows that the average end to end delay was lesser in random direction model. Manhattan Grid model performed worst in this case.

- Random Direction model gives the best Packet delivery ratio for TORA as shown in Figure E.8. Manhattan grid model performs well initially, but as the number of nodes rise, the packet delivery ratio degrades.
- Routing overhead is lesser in case of Random Direction model (Figure E.9).
- It is hence observed that Random Direction Model gives the best overall performance for TORA under TELNET traffic.

E.4. Performance Evaluation of TORA Routing Protocol Using HTTP Traffic

Experiment: Evaluation of average end to end delay and packet delivery ratio of TORA protocol under different mobility models and TELNET Traffic.

Objective: To analyze which model gives the best performance under TELNET Traffic.

Results: The results have been shown in Figure E.10. to Figure E.11.

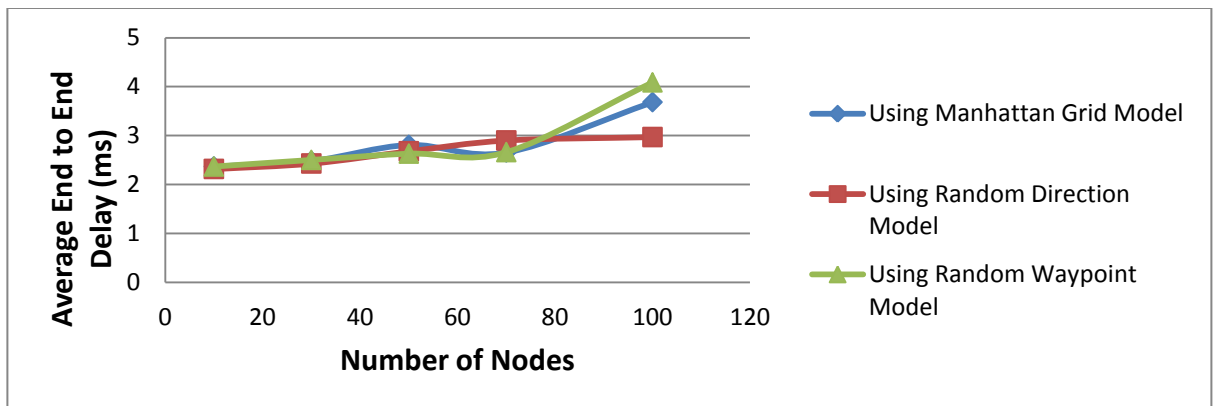


Figure E.10. Average End to End Delay of TORA using HTTP Traffic

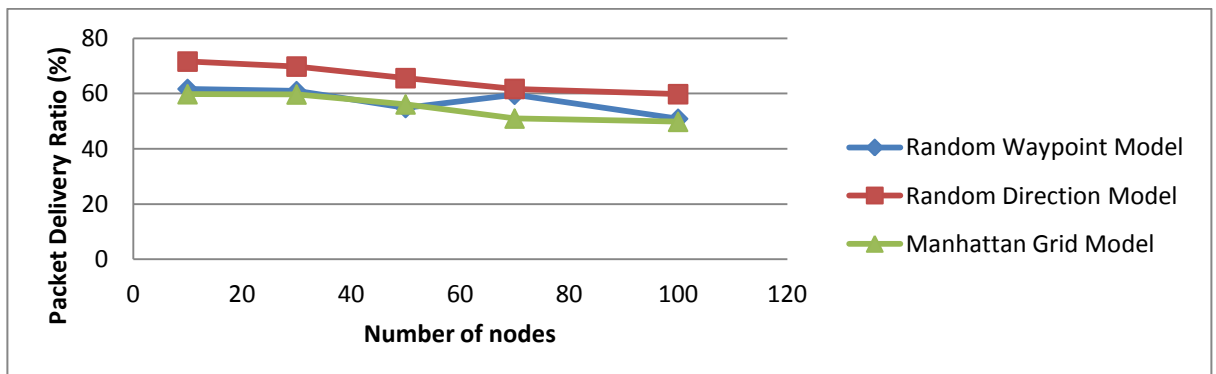


Figure E.11. Packet Delivery Ratio of TORA using HTTP Traffic

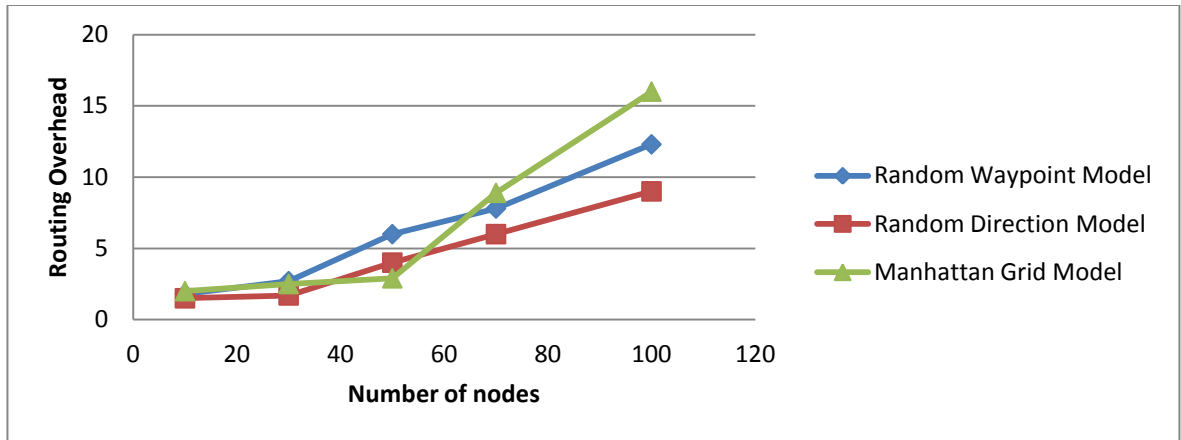


Figure E.12. Routing Overhead of TORA using HTTP Traffic

Analysis

- Random Direction model exhibited lesser delay than the other two models (Figure E.10.).
- As shown in Figure E.11, the Packet Delivery Ratio was good in case of Random Direction model and the other two models gave almost same performance.
- Performance of TORA in terms of Routing overhead is good under Random Direction model (Figure. E.12).
- Hence it can be observed that Random Direction model is the best performing model in case of HTTP traffic for TORA.