



ADDIS ABABA UNIVERSITY  
SCHOOL OF GRADUATE STUDIES  
FACULTY OF TECHNOLOGY  
ELECTRICAL AND COMPUTER ENGINEERING  
DEPARTMENT

**Performance Evaluation and Nodes' Mobility Effect Analysis on AODV based  
Internet Gateway Discovery Algorithms in Social Networks**

By  
Wondwossen Kassahun

A thesis submitted to the school of Graduate studies of Addis Ababa University in  
partial fulfillment of the requirements for the degree of

Masters of Science in Electrical and Computer Engineering  
(Computer Engineering)

March, 2010  
Addis Ababa, Ethiopia

Performance Evaluation and Nodes' Mobility Effect Analysis on AODV based Internet Gateway  
Discovery Algorithms in Social Networks

ADDIS ABABA UNIVERSITY  
SCHOOL OF GRADUATE STUDIES  
FACULTY OF TECHNOLOGY  
DEPARTMENT OF ELECTRICAL AND COMPUTER  
ENGINEERING

Performance Evaluation and Nodes' Mobility Effect Analysis on AODV  
based Internet Gateway Discovery Algorithms in Social Networks

By

Wondwossen Kassahun

Advisor

Dr. Kumuda Raimond

ADDIS ABABA UNIVERSITY  
SCHOOL OF GRADUATE STUDIES

ADDIS ABABA UNIVERSITY  
SCHOOL OF GRADUATE STUDIES  
FACULTY OF TECHNOLOGY

Performance Evaluation and Nodes' Mobility Effect Analysis on AODV  
based Internet Gateway Discovery Algorithms in Social Networks

By

Wondwossen Kassahun

APPROVAL BY BOARD OF EXAMINERS

Dr. Mengesha Mamo

Chairman Dept. of Graduate  
Committee

\_\_\_\_\_  
Signature

Dr. Kumudha Raimond

Advisor

\_\_\_\_\_  
Signature

Dr. Ing. Hailu Ayele

Internal Examiner

\_\_\_\_\_  
Signature

Dr. Eng. Tamrat Bayle

External Examiner

\_\_\_\_\_  
Signature

## **Table of Contents**

## **page**

|                                                                     |     |
|---------------------------------------------------------------------|-----|
| Acknowledgements .....                                              | VI  |
| List of Figures.....                                                | VII |
| List of Tables.....                                                 | IX  |
| List of Acronyms.....                                               | X   |
| Abstract .....                                                      | XII |
| Chapter one: Introduction .....                                     | 1   |
| 1.1 Back Ground.....                                                | 1   |
| 1.2 Statement of the Problem .....                                  | 4   |
| 1.3 Objective of the Thesis.....                                    | 5   |
| 1.3.1 General Objective.....                                        | 5   |
| 1.3.2 Specific Objective.....                                       | 5   |
| 1.4 Thesis Scope.....                                               | 6   |
| 1.5 Contribution.....                                               | 6   |
| 1.6 Methodologies.....                                              | 7   |
| 1.6.1 Literature Survey in the Area.....                            | 7   |
| 1.6.2 Modeling and Simulation.....                                  | 7   |
| 1.6.3 Mechanism of Driving Conclusion.....                          | 8   |
| 1.7 Thesis organization.....                                        | 8   |
| Chapter Two: Literature Survey.....                                 | 9   |
| 2.1 Related Works.....                                              | 9   |
| Chapter Three: Mobile Ad Hoc Networking.....                        | 15  |
| 3.1 Reactive, Proactive and Hybrid Routing Protocols .....          | 15  |
| 3.2 AD HOC on demand distance vector (AODV) routing.....            | 16  |
| 3.2.1 Route Discovery.....                                          | 17  |
| 3.2.2 Route Maintenance.....                                        | 20  |
| 3.2.3 Advantages of AODV Routing protocol.....                      | 20  |
| 3.3 The Protocol Stack.....                                         | 21  |
| Chapter Four: Internet Connectivity for Mobile Ad Hoc Networks..... | 23  |
| 4.1 Connectivity to the Internet.....                               | 23  |
| 4.2 Inter-working.....                                              | 23  |
| 4.3 Enhanced AODV Protocol.....                                     | 24  |
| 4.3.1 The Enhanced Route Request.....                               | 24  |
| 4.3.2 The Enhanced Route Reply.....                                 | 25  |
| 4.3.3 The Gateway advertisement (GWADV).....                        | 25  |
| 4.3.4 The Default Route.....                                        | 26  |
| 4.3.5 The Expanding Radial Ring Search.....                         | 26  |
| 4.3.6 Internet Gateway Operation When Receiving Requests.....       | 27  |
| 4.3.7 The Routing Table.....                                        | 27  |
| 4.3.8 Operation of Intermediate Node on Receiving Requests.....     | 29  |
| 4.3.9 Unreachable Gateway.....                                      | 29  |
| 4.4 Gateway Discovery .....                                         | 30  |

Performance Evaluation and Nodes' Mobility Effect Analysis on AODV based Internet Gateway  
Discovery Algorithms in Social Networks

|                                                                |     |
|----------------------------------------------------------------|-----|
| 4.4.1 Proactive Gateway Discovery.....                         | 30  |
| 4.4.2 Reactive Gateway Discovery.....                          | 31  |
| 4.4.3 Hybrid Gateway Discovery.....                            | 31  |
| Chapter Five: Impact of Node Mobility and Mobility Models..... | 33  |
| 5.1 Mobility Influence on Gateway Discovery Algorithms.....    | 33  |
| 5.2 Mobility Models.....                                       | 33  |
| 5.2.1 Entity Models.....                                       | 34  |
| 5.2.1.1 Random Walk Mobility Model.....                        | 34  |
| 5.2.1.2 Random Way Point Mobility Model.....                   | 35  |
| 5.2.2 Group Mobility Models.....                               | 36  |
| 5.2.2.1 Reference Point Group Mobility Model.....              | 36  |
| 5.2.3 New Mobility Models.....                                 | 36  |
| 5.2.3.1 Obstacle Mobility Model.....                           | 37  |
| 5.2.3.2 Community Based Mobility Model.....                    | 38  |
| Chapter Six: Simulation System.....                            | 41  |
| 6.1 Network Simulator (NS2) Overview.....                      | 41  |
| 6.2 Simulation Model.....                                      | 42  |
| 6.2.1 Simulation Environment.....                              | 42  |
| 6.2.2 Simulation Setup.....                                    | 42  |
| 6.2.3 Performance Evaluation Metrics.....                      | 43  |
| 6.2.4 Mobility Model and Traffic Type.....                     | 47  |
| 6.2.5 Communication Model.....                                 | 47  |
| 6.3 Simulation Parameters.....                                 | 47  |
| 6.4 Scenarios.....                                             | 49  |
| Chapter Seven: Simulation Results and Discussions.....         | 52  |
| 7.1 Results Obtained Using Scenario One.....                   | 52  |
| 7.2 Results Obtained Using Scenario Two.....                   | 59  |
| 7.3 Results Obtained Using Scenario Three.....                 | 65  |
| 7.4 Results Obtained Using Scenario Four.....                  | 71  |
| 7.5 Summery of the Results.....                                | 77  |
| Chapter Eight: The Proposed Solution.....                      | 77  |
| 8.1 Design of the Solution.....                                | 78  |
| 8.1.1 Predictive GWDA.....                                     | 78  |
| 8.2 Implementation of the Solution.....                        | 84  |
| 8.3 Result and Discussion.....                                 | 86  |
| 8.4 Summery of the Results.....                                | 91  |
| Chapter Nine: Conclusion and Recommendation.....               | 99  |
| 9.1 Conclusion.....                                            | 92  |
| 9.2 Future Works.....                                          | 94  |
| References.....                                                | 95  |
| Appendix A.....                                                | 99  |
| Appendix B.....                                                | 103 |

### **Acknowledgement**

My deepest gratitude goes to My Advisor, Dr. Kumudah Raimond, for her dedication, continuous follow-up and support throughout the thesis work.

I would like to thank Dr. Alihamedian from LUND University, for his useful support while implementing the gateway discovery algorithms. I wish to extend my acknowledgement to my families and friends for their invaluable encouragement and supports.

Finally, I would like to thank almighty God, as nothing would have been possible without his help.

## List of figures

|                                                                                           |    |
|-------------------------------------------------------------------------------------------|----|
| Figure 1.5: Thesis Scope.....                                                             | 6  |
| Figure 3.1: Route Request and Reverse path Setup in AODV protocol.....                    | 18 |
| Figure 3.2: Forward Path Setup in AODV protocol.....                                      | 18 |
| Figure 3.3: Packet Processing in AODV.....                                                | 19 |
| Figure 3.4: The OSI Model, TCP/IP Suite and MANTE Protocol Stack.....                     | 22 |
| Figure 4.1: The Protocol Stacks used by Mobile Nodes, Gateways and Internet<br>Nodes..... | 24 |
| Figure 4.2: Packet format of the RREQ_I message.....                                      | 25 |
| Figure 4.3: Packet format of the RREP_I message.....                                      | 25 |
| Figure 5.1: Example of Initial Configuration (Community based mobility).....              | 39 |
| Figure 6.1: Simplified Users' View of NS2.....                                            | 41 |
| Figure 6.2: Screen Shot of the Simulation Environment on NS2.....                         | 43 |
| Figures for Scenario One Results                                                          |    |
| Figure 7.1: Percentage of handover time vs. traveling nodes' speed.....                   | 52 |
| Figure 7.2: Average handover time vs. traveling nodes' speed.....                         | 53 |
| Figure 7.3: Discovery time in percentage vs. traveling nodes' speed.....                  | 53 |
| Figure 7.4: Average Discovery time vs. traveling nodes' speed.....                        | 53 |
| Figure 7.5: Percentage of disconnection time vs. traveling nodes speed.....               | 54 |
| Figure 7.6: Expanding radial-ring search time vs. traveling nodes' speed.....             | 54 |
| Figure 7.7: Packet delivery fraction vs. traveling nodes' speed.....                      | 54 |
| Figure 7.8: Average end-to-end delay vs. traveling nodes' speed.....                      | 55 |
| Figure 7.9: Routing overhead vs. traveling nodes' speed.....                              | 55 |
| Figure 7.10: Normalized routing load vs. traveling nodes' speed.....                      | 55 |
| Figures for Scenario Two Results                                                          |    |
| Figure 7.11: Percentage of handover time vs. traveling nodes' speed.....                  | 59 |
| Figure 7.12: Average handover time vs. traveling nodes' speed.....                        | 60 |
| Figure 7.13: Discovery time in percentage vs. traveling nodes' speed.....                 | 60 |
| Figure 7.14: Average Discovery time vs. traveling nodes' speed.....                       | 60 |
| Figure 7.15: Percentage of disconnection time vs. traveling nodes speed.....              | 61 |

|                                                                                |    |
|--------------------------------------------------------------------------------|----|
| Figure 7.16: Expanding radial-ring search time vs. traveling nodes' speed..... | 61 |
| Figure 7.17: Packet delivery fraction vs. traveling nodes' speed.....          | 61 |
| Figure 7.18: Average end-to-end delay vs. traveling nodes' speed.....          | 62 |
| Figure 7.19: Routing overhead vs. traveling nodes' speed.....                  | 62 |
| Figure 7.20: Normalized routing load vs. traveling nodes' speed.....           | 62 |

#### Figures for Scenario Three Results

|                                                                                |    |
|--------------------------------------------------------------------------------|----|
| Figure 7.21: Percentage of handover time vs. traveling nodes' speed.....       | 65 |
| Figure 7.22: Average handover time vs. traveling nodes' speed.....             | 65 |
| Figure 7.23: Discovery time in percentage vs. traveling nodes' speed.....      | 66 |
| Figure 7.24: Average Discovery time vs. traveling nodes' speed.....            | 66 |
| Figure 7.25: Percentage of disconnection time vs. traveling nodes speed.....   | 66 |
| Figure 7.26: Expanding radial-ring search time vs. traveling nodes' speed..... | 67 |
| Figure 7.27: Packet delivery fraction vs. traveling nodes' speed.....          | 67 |
| Figure 7.28: Average end-to-end delay vs. traveling nodes' speed.....          | 67 |
| Figure 7.29: Routing overhead vs. traveling nodes' speed.....                  | 68 |
| Figure 7.30: Normalized routing load vs. traveling nodes' speed.....           | 68 |

#### Figures for Scenario Four Results

|                                                                                |    |
|--------------------------------------------------------------------------------|----|
| Figure 7.31: Percentage of handover time vs. traveling nodes' speed.....       | 71 |
| Figure 7.32: Average handover time vs. traveling nodes' speed.....             | 72 |
| Figure 7.33: Discovery time in percentage vs. traveling nodes' speed.....      | 72 |
| Figure 7.34: Average Discovery time vs. traveling nodes' speed.....            | 72 |
| Figure 7.35: Percentage of disconnection time vs. traveling nodes speed.....   | 73 |
| Figure 7.36: Expanding radial-ring search time vs. traveling nodes' speed..... | 73 |
| Figure 7.37: Packet delivery fraction vs. traveling nodes' speed.....          | 73 |
| Figure 7.38: Average end-to-end delay vs. traveling nodes' speed.....          | 74 |
| Figure 7.39: Routing overhead vs. traveling nodes' speed.....                  | 74 |
| Figure 7.40: Normalized routing load vs. traveling nodes' speed.....           | 74 |
| Figure 8.1: Prediction algorithm operation at intermediate node.....           | 82 |
| Figure 8.2: Prediction algorithm operation at the source node.....             | 83 |

Figures for Scenario three Results of the Solution

|                                                                      |    |
|----------------------------------------------------------------------|----|
| Figure 8.3: Packet delivery fraction vs. traveling nodes' speed..... | 86 |
| Figure 8.4: Average end-to-end delay vs. traveling nodes' speed..... | 86 |
| Figure 8.5: Routing overhead vs. traveling nodes' speed.....         | 87 |
| Figure 8.6: Normalized routing load vs. traveling nodes' speed.....  | 87 |

Figures for Scenario Four Results of the Solution

|                                                                      |    |
|----------------------------------------------------------------------|----|
| Figure 8.7: Packet delivery fraction vs. traveling nodes' speed..... | 88 |
| Figure 8.8: Average end-to-end delay vs. traveling nodes' speed..... | 89 |
| Figure 8.9: Routing overhead vs. traveling nodes' speed.....         | 89 |
| Figure 8.10: Normalized routing load vs. traveling nodes' speed..... | 89 |

**List of tables**

|                                                               |    |
|---------------------------------------------------------------|----|
| Table 4.1 Routing table of a mobile node.....                 | 28 |
| Table 7.1.Simulation parameter settings for scenario one..... | 50 |
| Table 7.2.Simulation parameter settings for scenario two..... | 50 |

## **List of Acronyms**

|            |                                                   |
|------------|---------------------------------------------------|
| AEED       | Average End to End Delay                          |
| AODV       | Ad hoc On-demand Distance Vector                  |
| CDMA       | Code Division Multiple Access                     |
| CBR        | Constant Bit Rate                                 |
| DAG        | Direct Acyclic Graph                              |
| DSR        | Dynamic Source Routing                            |
| FN         | Fixed Node                                        |
| GW         | Gateway                                           |
| GWDA       | Gateway Discovery Algorithm                       |
| GWADV      | Gateway advertisement                             |
| GPRS       | General Packet Radio Service                      |
| GSM        | Global System for Mobile Communication            |
| HIPERLAN 2 | High-Performance Local Area Network type 2        |
| ICMP       | Internet Control Message Protocol                 |
| IEEE       | Institute of Electrical and Electronics Engineers |
| IETF       | Internet Engineering Task Force                   |
| IP         | Internet Protocol                                 |
| IPv6       | Internet Protocol version 6                       |
| LAN        | Local Area Network                                |
| LLC        | Logical Link Control                              |
| MAC        | Media Access Control                              |
| MANET      | Mobile Ad hoc Network                             |
| MN         | Mobile Node                                       |
| NAM        | Network Animator                                  |
| Ns2.29     | Network simulator version 2.29                    |
| OSI        | Open Systems Interconnection                      |
| OTcl       | Object Tool command language                      |
| OLSR       | Optimized Link State Protocol                     |

Performance Evaluation and Nodes' Mobility Effect Analysis on AODV based Internet Gateway  
Discovery Algorithms in Social Networks

|        |                                      |
|--------|--------------------------------------|
| PDA    | Personal Digital Assistant           |
| RREQ_I | Enhanced Route Request               |
| RREP_I | Enhanced Route Reply                 |
| RREP   | Route REPLY                          |
| RREQ   | Route REQuest                        |
| RPGM   | Reference Point Group Mobility Model |
| TCI    | Tool command script language         |
| TCP    | Transmission Control Protocol        |
| TTL    | Time To Live                         |
| UDP    | User Datagram Protocol               |
| ZRP    | Zone Routing Protocol                |

## **Abstract**

Mobile Ad hoc networks (MANETs), with gateway (GW) discovery protocols for internet connectivity, allows portable devices to establish communication with hosts in the fixed network (or to the internet).

The Ad hoc on Demand Distance Vector (AODV) MANET routing protocol, enhanced to support proactive, reactive and hybrid Gateway Discovery Algorithms (GWDAs), has been proposed and applied in various literature works. For MNs to get connection to the internet, the GW node has to be discovered by using a GWDA and configured. The frequent topology change due to nodes' movement in MANETs is a challenge in affecting performance of the GWDAs causing frequent link failures. This results in frequent GW discoveries and hence increased total discovery, handover between GW nodes and disconnection time. The link failure also causes packet drops that are being exchanged to the internet. In this thesis work, the impact of nodes' mobility on the performance of reactive, proactive and hybrid GWDAs is studied thoroughly based on the density of nodes, traveler's speed and strength of social interaction between nodes. As (MNs) are nothing but portable devices that are carried by humans, a more realistic mobility model (community based mobility model), that can show human socialization and cooperation behavior is used for the simulation. The results of this work showed that, the increase in travelers' speed up to 10m/s shows improvement in the performances of the algorithms as it helps the distribution of MNs in the simulation area, but after that, performance of all the three algorithms degrades. The high social interaction helps to attain the best performance at lower speed than the low social interaction mobility condition but as travelers' speed increases it aggravates the performance degradation of the algorithms. The less dense environment is observed to aggravate the effect of mobility on all algorithms. Comparatively the proactive GWDA, with greater routing overhead, out performs the other two. An algorithm which improves the reactive GWDA by performing discovery before a GW route fails, called predictive

GWDA is proposed in this thesis work. It is observed to perform much better than the reactive GWDA and has nearly equivalent performance with the proactive GWDA, but with a 5% to 7% lesser overhead.

**Key words:** MANET, AODV, GW Discovery, GWDA, Proactive, Reactive, Hybrid, Community based mobility model.

## **Chapter one: Introduction**

### **1.1 Background**

Wireless networks, in recent years, have become increasingly popular in the computing industry. This is particularly true within the past decade, which has seen wireless networks being adapted to enable mobility.

There are currently two variations of mobile wireless networks. The first is known as the infrastructured network which is a network with fixed and wired gateways. The bridges for these networks are known as base stations [1]. A mobile unit within these networks connects to, and communicates with, the nearest base station, that is within its communication radius. As the mobile travels out of range of one base station and into the range of another, a “handoff” or in GSM (Global System for Mobile Communication) “handover” occurs from the old base station to the new, and the mobile is able to continue communication seamlessly throughout the network. Typical applications of this type of network include office wireless local area networks (WLANs), (GSM) [1] [2].

Internet connection in infrastructure networks is performed through base stations. The mobile unit however, should be within a cell (area in the communication range of a base station antenna). The base station and the mobile unit should be in LOS. General Packet Radio Service (GPRS), wide band code division multiple access (WCDMA), and code division multiple access (CDMA2000) are different technologies that are implemented for voice and data communication in the world today. As multi-hop communication is not possible in such networks, a number of base stations need to be installed to have a full coverage, which increases the cost of deployment and also affects the beauty of the environment where a number of base stations are installed.

The second type of mobile wireless network is the infrastructureless mobile network, commonly known as an ad hoc network. Infrastructureless networks have no fixed routers; all nodes are capable of movement and can be connected

dynamically in an arbitrary manner. The power capacity of individual nodes and wireless bandwidth are limited. Additionally, the channel condition can vary greatly. Routes in these networks may constantly change as a result of the mobility of the nodes. The ease of deployment and lack of the need for any infrastructure make MANET an attractive choice for a variety of applications. The term “ad hoc” means “happening only for a particular purpose or need”, so it implies that the network is meant for special purpose. It should be able to set up with minimum manual effort and short time.

MANETs have primarily been used for tactical network related applications to improve battlefield communications. The dynamic nature of military operations means that the military cannot rely on access to a fixed pre-placed communications infrastructure in battlefield. The existing wireless communication also has limitations in that radio signals are subject to interference and for radio frequency higher than 100MHz rarely propagate beyond line of sight (LOS) [1]. MANETs creates a suitable framework to address these issues by providing a multi-hop wireless network without pre-placed infrastructure and connectivity beyond LOS.

Internet connection in MANETs in relation to their applications is one of the areas of focus of researchers these days. The applications of MANET is currently extended to commercial purposes, because they help to realize network services for mobile users in areas with no communication infrastructure, or when the use of infrastructure requires wireless extensions (E.g. Internet)[1][2]. These advantages make ad hoc networking an attractive option for future wireless networks. They are expected to become important part of the fourth generations (4G) architecture [2], [20] where a major goal is provision of pervasive computing environments that can seamlessly and ubiquitously support users in accomplishing their tasks, in accessing information or communicating with users at anytime, anywhere and from any device [1], [2]. While accomplishing such a service, the demand for connectivity to the internet is inevitable.

To provide the connectivity to the internet, nodes of the ad hoc network can connect to the fixed network through gateways. Gateways can be used by mobile terminals to seamlessly communicate with other nodes in the fixed network. Therefore, a user inside an ad hoc network must be able to discover the GW according to its application requirements. The discovered GW then has to be configured to have a globally valid address to the Mobile Nodes (MNs) for incoming and outgoing traffic. [6]

Several approaches to enhance ad-hoc routing protocols to support MNs accessing the Internet were developed. First, there is the so called proactive approach, where the Internet GW periodically broadcasts advertisements into the MANET to indicate its presence. Secondly, in the reactive approach, a MN of the MANET asks reactively for GW services by broadcasting solicitations [3]. A hybrid discovery algorithm is a combination of the proactive and the reactive algorithm. In this case, the GW sends advertisements, which are only forwarded for a limited number of hops. A MN which does not receive an advertisement for specified time period will additionally search for a GW with the aid of solicitation messages [3], [6], [10], [12].

In MANET, connection to the GW node is established by using multi-hop communication. If any two of the mobile nodes on the route to the internet are found to be out of the communication range of each other, the link between them will fail. This in turn causes the failure of the route to the internet. As a consequence of the route failure, packets to be exchanged with the internet through the GW node will be dropped. Moreover, in order to continue the communication with the internet, a new discovery has to be performed, which means, additional time for discovery. The time spent to find a new route also increases the end to end delay of packet transmission. This in turn affects the performance of the GWDAs and hence the network.

## **1.2 Statement of the Problem**

The two major factors that lead to link failure are node mobility and collision due to interference [26]. Interference occurs when nodes come closer to each other, so it is also directly related to the movement of nodes. But the effect of interference relative to the effect of mobility can be minimized, if the nodes are moving in a less dense environment and in a wider area. The main focus of this thesis works is studying the effect of mobility on the performance of AODV based internet GWDAs.

The link failure due to mobility, in MANETs occurs because of the movement of MNs outside the radio range of each other, following a specified movement pattern. That means, the movement pattern followed in simulation studies, is a very important factor when evaluating performance of routing protocols and algorithms. The mobility model used determines the movement patterns of MNs. But, MNs are nothing but different communication devices that are carried by humans (E.g. Mobile phones, PDAs, Laptops etc). And hence a more realistic mobility model that can better represent the human socialization and cooperation behaviors should be applied for the simulation study.

In the related works (Section 2.1), evaluations of the three GWDAs are performed by varying the advertisement interval of the GW nodes. In these works, though the focus is not studying the effect of mobility, a randomly generated movement patterns by using random-way point mobility model are used. However, results obtained by simulation studies depends highly on how realistic the mobility model applied in the simulation is[13], hence using unrealistic mobility model, affects the soundness of the results obtained and may lead to wrong conclusions. Generally, the effect of mobility on performance of AODV based internet GWDAs with realistic movement patterns of MNs has not been given sufficient emphasis and studied.

Community based mobility model is a more realistic mobility model [13]. As clearly stated in Section (5.2.3), this model takes into account the human socialization behaviors. And movement of nodes is governed by the social attraction between individual nodes and also among groups. It can better represent the real time scenario, and is suitable when studying the effects of mobility on performance of protocols and algorithms in MANETs.

Therefore, the questions, how does the social mobility condition affects the performance of each GWDAs?, and which of the three GWDAs is more resistive to the effects of social mobility on its performance?, are interesting to answer.

## **1.3 Objective of the Thesis**

### **1.3.1 General Objective**

The main objectives are to analyze the impact of social mobility on the performance of AODV based internet GWDAs and to perform a comparative evaluation of the algorithms by using a more realistic mobility model.

Additionally, propose and test predictive GWDA.

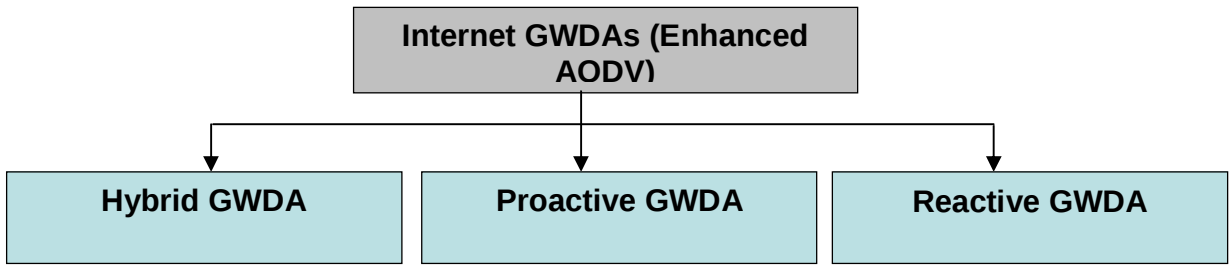
### **1.3.2 Specific Objective**

The specific Objectives of the thesis are:

- o Studying AODV based internet GWDAs.
- o Studying community based mobility model and its implementation on NS2 simulator.
- o Integrate the algorithms on NS2 simulator and determine the simulation environment, parameters, scenarios to be applied and metrics to be used.
- o Perform simulation.
- o Conduct analyses to see, how mobility affects the performance of each GWDAs.
- o Carry out comparative evaluation among the three GWDAs.
- o Additionally, propose the predictive GWDA and perform simulation study.

## 1.4 Thesis Scope

GWDAs are basically proactive, reactive or hybrid depending on whether the discovery is initiated by the gateway, a MN or both based on the advertisement zone respectively. The implementation of the algorithms can use Mobile IP together with MANET routing protocols [22], or it can be implemented by using the extension of MANET routing protocols [6], [12], [21], [24]. This thesis work focuses on studying the effect of mobility on the performance of the three GW discovery techniques. AODV routing protocol is the best protocol in terms of resisting the effects of mobility [8], [4]. As a result, internet GWDAs which are based on AODV routing protocol are applied for the purpose of this work.



*Figure 1.5: Thesis Scope*

Applying realistic mobility (which can represent human social interaction), simulation setups that can maximize the effect of mobility and all relevant metrics, analysis and comparative evaluation is performed on the three internet GWDAs' performance. Beyond achieving the objective of this thesis work, a predictive GWDA has been proposed and tested.

## 1.5 Contribution

This thesis work presents a thorough analysis, showing how the AODV based internet GWDAs' performance is affected with social mobility conditions. For this purpose, selections of appropriate scenarios and relevant metrics have been performed. In the related works the focus is not studying the effect of mobility and in most cases random movement patterns are used, which affects the soundness of the results obtained and may lead to wrong conclusions [13]. They

also applied Packet Delivery Fraction(PDF), Average End-to-End Delay and Routing Overhead metrics, but in this work, 10 metrics has been used to get information from different point of views about the impact of mobility on GWDAs' performance.

As the average value fails to show effects related to the number of occurrences, average and percentage values are used. Additionally, a new approach for computation of metrics, like GW discovery time, disconnection time etc directly from the trace file has been introduced. Based on the analysis, the selection of the best GWDA, in terms of resisting the effects of social mobility is made. This facilitates for further works, like enhancing it, so that; it will have even a better performance, resisting the effects of mobility.

After attaining the objective of this thesis work, a predictive GWDA, which adds additional packets on the reactive GWDA to predict GW route failure and initiate discovery in advance is proposed and tested.

## **1.6 Methodologies**

### **1.6.1 Literature survey in the area**

There has been enough literature review on the previous works in the area. Information regarding different GWDAs' way of operation, comparison between them with increment in the advertisement interval time, and absence of study on the effect of social mobility on performance of the algorithms has been examined. Additionally, the information concerning the different mobility models is obtained from literature review.

### **1.6.2 Modelling and Simulation**

In this thesis work, the mobility of the nodes connecting to the internet are represented in such a way that, they are connected using the other nodes as multi-hop link through the GW node to hosts on the fixed network. Two GW nodes, two hosts, two fixed routers and 30 MNs are used for the simulation. Community based mobility model is used to study the effect of mobility. Variable nodes' speed, simulation area and social interaction are applied to create the

scenarios and about 10 metrics (chapter-6) are used for a thorough analysis on the impact of mobility on the performance of reactive, proactive and hybrid GWDAs. Ns 2.29 allinone on Linux operating system is used for performing the simulations.

### **1.6.3 Mechanism of Driving Conclusion**

The effect of mobility on performance of AODV based internet GWDAs have been analyzed in this thesis work. All relevant metrics and more realistic mobility model (i.e. community based mobility model) have been used for the analysis and comparison on NS-2 simulator.

The simulation results and the analysis made considering the relationship between the metrics will be used to drive conclusion for the objectives of the work.

### **1.7 Thesis organization**

The rest of this paper is organized as follows: chapter two deals with literature review of related works. The general basic concepts about mobile ad hoc networking, Internet connectivity and working concepts of three GWDAs are discussed in chapter three and four respectively. In chapter five and six, the effect of node mobility on GW discovery process and the simulation system used in the thesis work are discussed; in chapter Seven, Results of the simulation and the corresponding discussion is provided. In Chapter eight, the proposed solution design, implementation and evaluation is presented. Finally, the conclusion and recommendations together with the possible outlooks for future work are provided in chapter nine.

## **Chapter Two: Literature Survey**

### **2.1 Related Works**

In this section, previous research works, which are directly or indirectly related to this thesis work, are examined. The area of focus and the limitations of these works are also discussed.

The research paper by M.Rosenschon et al. [3] describes an approach for GW discovery based on HELLO packets of the AODV protocol. The performance of the algorithm in terms of the discovery time and the handover delay is measured for AODV based reactive, proactive and hybrid GWDAs. Link layer detection scheme is a modern approach rather than using HELLO messages, which adds extra overhead in the network. In this work, the focus is not studying the effect of mobility on the performance; rather it performed the evaluation based on the advertisement intervals of the GW nodes. The mobility model applied in the simulation is unrealistic (i.e. random way point mobility model). Additionally, a single mobile test node has been applied to check performance of the algorithms while the rest of the nodes are made stationery, which is again an unrealistic assumption.

A second paper by the same researchers [10] considered a fixed speed range and a fixed pause time with random way point mobility model for evaluating the algorithms. In this case also, the focus is not on studying the effect of mobility on performance. The mobility model is still unrealistic [13]. In both the papers, the GW discovery time and the handoff time metrics are computed by using formulas. The formulas were derived based on simulation area, GW advertisement interval, number of nodes and radio range rather than using directly the trace file measurements. Node mobility characteristics like speed and direction of movement are not considered.

The ad hoc routing protocol AODV is one of the promising routing protocols investigated by the MANET working group. It can be used in a mobile ad hoc network to route packets between MNs. However, it cannot provide Internet access to the MNs because it does not support routing between a fixed network like the Internet and a mobile ad hoc network. In the Internet draft “Global Connectivity for IPv6 Mobile Ad Hoc Networks” a solution is presented where the AODV protocol can be modified in such a way that it can route packets not only within a mobile ad hoc network, but also to a fixed, wired network. The research work in [6] aims to implement this solution in NS2 simulation environment and compare the different approaches for GW discovery. However, the evaluation is performed by varying the internet gateways nodes' advertisement interval, for a fixed speed range using random way point mobility model. Additionally, the dimension used in the simulation doesn't allow nodes to escape outside the radio-range of each other and hence the effect of interference could be greater than mobility. In this paper, the metrics like GW discovery time, handoff time and expanding-radial-ring-search time are not considered.

In [12], a proactive GWDA, that takes into account the size of interface queue for GW selection in addition to minimum hop metric, is implemented. This work is based on the research paper mentioned in [6], which concludes proactive approach as the best one among the three techniques based on the evaluation made by varying the advertisement interval of the GW nodes. In this paper, the evaluation of the new approach is shown to perform better as the number of packets exchanged is increased from 5packets/Sec to 30Packets/Sec, for fixed packet size of 512 bytes. The focus is not the effect of mobility but rather the effect of traffic load. Random way point mobility model for fixed speed range is applied for the simulation, which is unrealistic [13]. The 500m dimension taken for the width of the simulation area limits the mobility of the nodes not to escape outside the radio range of each other. The paper didn't show the GW discovery time and handoff time metrics for the evaluation.

The three GWDAs based on the extension of AODV MANET routing protocol has been evaluated in [21]. The evaluation is performed by varying the maximum speed of the random way point mobility model. Three different numbers of nodes and the corresponding simulation areas have been considered, keeping the density of the nodes nearly constant. The limitations of this thesis work are: firstly, the mobility model applied is not capable of showing mobility condition related to human socialization behavior [13]. And as a result it is not suitable to study the effect of mobility on performance. Secondly, the effect of mobility on performance is dominant than other factors like interference, in case of wider simulation area and lesser density of nodes. But in this work, a nearly constant density of nodes is considered. Thirdly, the speed range considered is not sufficient enough to represent mobility conditions in cities which can reach beyond 20m/s. The fourth limitation is, the research paper didn't consider the GW discovery time and handover time metrics.

Jahanzeb Farooq [22], in his paper discusses two issues; the mobility and connectivity of individual nodes within the ad hoc network, and, the mobility and connectivity of the ad hoc network with the Internet. Integration of Mobile IP with ad hoc routing protocols is provided as a solution for internet connectivity. The paper presents the theoretical explanations on how to provide the solution but didn't conduct any simulation.

In [13], it is explained that, movement of MNs is strongly affected by humans to socialize or to cooperate; this is known to associate in particular ways that can be mathematically modeled and also have been studied in social science for years. Accordingly, a new mobility model founded on social network theory has been proposed. The model allows collections of hosts to be grouped together in a way that is based on social relationships among the individuals. This grouping is then mapped to a topographical space, with movements influenced by the strength of social ties that may also change in time. The researchers have

validated their model with real traces by showing that their community based mobility model is a very good approximation of human movement patterns.

Analytical model to evaluate Proactive, Reactive and Hybrid GWDAs, which are based on extension of AODV Ad hoc routing protocol is presented in [4]. The nodes are assumed to be located on a square area at equal distance from each other; additionally, adaptive GWDA is suggested based on the dynamic adjustment of the scope of the GW advertisement packets. The simulation uses unrealistic mobility model to evaluate the performance of the algorithm (random way point mobility model) by varying advertisement interval rather than the speed of nodes.

In [23], Internet access of mobile devices in wireless ad hoc network via internet GW nodes is considered. Description of problems and alternative approaches as solutions for access router discovery, addressing, and routing has been proposed. For the path selection problem, prefix cache is suggested in each node to allow the node to optimize the routing path to ad hoc nodes in adjacent ad hoc network. Simulation study is not shown.

Comparison of different techniques to interconnect Ad Hoc networks with the internet (i.e. techniques proposed by researchers on IETF draft papers: Wakikawa [16], Jelger [17], Singh [18] and Ruiz [19]) are evaluated by using three mobility models, namely: Random-way Point and Gauss Markov mobility model [34] and Manhattan Grid mobility models in [15]. The models applied are old common models and Speed of 5m/s and 15m/s are considered, which doesn't represent the practical mobility speed ranges. The metrics considered doesn't include GW discovery time and handoff times.

There are essentially two possible types of mobility patterns that can be used to evaluate mobile network protocols and algorithms by means of simulations according to survey performed in [5]. These are traces and synthetic models. It also states that, traces are obtained by means of measurements of deployed systems and usually consist of logs of connectivity or location information,

whereas synthetic models are mathematical models, such as sets of equations, which try to capture the movement of the devices. The survey emphasizes the effect of choosing appropriate mobility model to find a sound result in simulation studies and also points out some more areas as future research work.

The integration of mobile ad hoc networks into IP based access networks is implemented using the extension of AODV MANET routing protocol in [6]. And the advertisement interval of the GW node is made to vary dynamically by using a control system which has been designed using analytical model. The research paper states that, this gives a better and adaptive GWDA. But the evaluation has been performed using random way point mobility model with maximum speed range varied from 2m/s up to 5m/s. On top of using unrealistic mobility model, the speed range applied cannot represent practical speed ranges. Additionally, the simulation doesn't consider the phenomena before 50<sup>th</sup> second of the simulation time.

The following papers are related to the predictive GWDA proposed in this thesis work. All preemptive solutions are related to MANET routing protocols not directly with GWDAs. But it is worth mentioning them.

In [25], the authors propose a preemptive route maintenance extension to on demand routing protocols. The received transmission power is used to estimate when a link is expected to break. A link is considered to break when the power of the signal received is below some threshold. Route repair is the responsibility of the source node after receiving a warning about the possibility of link break on an active route for destination. (Dynamic source routing)DSR and AODV are used in the work.

Prediction based DSR is presented in [26]. The authors used the prediction technique to enhance the PDF of this MANET routing protocol. The work showed that proactive route maintenance can be used to enhance the performance of the protocol as it can be affected by the route break due to node mobility.

In [27], authors presented the mobility prediction based algorithm for route maintenance in MANET. They incorporate the prediction for DSR. Their simulation result showed that their algorithm to offer significant benefits to DSR.

Another prediction based preemptive AODV is presented in [28]. The authors use the interpolation technique: Lagarange to predict the power after some time  $t$ , which is the function of times of three received power signal used by the algorithm and the average route discovery time. The prediction algorithm starts when the received packet power level is bellow some threshold value.

From the above literature survey performed in this thesis work, a research paper which has targeted to show the effect of mobility on the performance of internet GWDAs using realistic mobility model has not been observed. But, a lot of evaluation works on the three GWDAs among themselves and also with proposed solutions by varying the internet gateways advertisement interval has been performed with fixed speed range. And in one of the papers [21], evaluation has been performed by varying maximum speed of nodes up to 20m/s using random mobility model. In all these cases, the mobility models used failed to reflect humans' socialization and cooperation behaviors (i.e. very critical point in new realistic mobility models). This indicates the importance of testing the algorithms for the impact of mobility on their performance by using a realistic mobility model. The speed ranges applied, the simulation setup used, number of sources applied etc has also been observed as limitations of the research works to reflect the real time scenario. Besides, consideration of all relevant metrics for evaluation is observed as an additional short come. Therefore, in this thesis work, performance evaluation with respect to social mobility and selection of the best algorithm is made.

The prediction protocols discussed in the literature are pure MANET routing protocols. They are not used for internet GWDAs.

The next chapter discusses some basic concepts related to mobile Ad Hoc networking.

## **Chapter Three: Mobile Ad Hoc Networking**

This chapter gives an overview of Mobile Ad Hoc Networking. In section 3.1, reactive, proactive and Hybrid routing protocols are discussed. In section 3.2, AODV routing protocol, which is one of the reactive routing protocols is introduced. In section 3.3, the protocol stacks used in the Internet and MANET and their comparison with the Open Systems Interconnection (OSI) model is discussed.

### **3.1 Reactive, Proactive and Hybrid Routing Protocols**

Traditional distance-vector and link-state routing protocols [3] [6] are proactive, in that they maintain routes to all nodes, including nodes to which no packets are sent. For that reason, they require periodic control messages, which lead to scarce resources such as power and link bandwidth being used more frequently for control traffic as mobility increases. One example of proactive routing protocol is Optimized Link State Routing Protocol (OLSR) [1], [2]. Reactive routing protocols, on the other hand, operate only when there is a need for communication between two nodes. This approach allows the nodes to focus either on routes that are being used or on routes that are in process of being set up. Examples of reactive routing protocols are AODV [1], [9], and Dynamic Source Routing (DSR) [11], [2].

Both proactive and reactive routing have specific advantages and disadvantages that make them suitable for certain types of scenarios. Proactive routing protocols have their routing tables updated at all times, thus the delay before sending a packet is minimal. However, routing tables that are always updated require periodic control messages that are flooded in the whole network - an operation that consumes a lot of time, bandwidth and energy. On the other hand, reactive routing protocols determine routes between nodes only when they are explicitly needed to route packets. However, whenever there is a need for sending

a packet, the MN must first find the route if the route is not already known. This route discovery process may result in considerable delay.

Combining the proactive and reactive approaches result in a hybrid routing protocol. A hybrid approach shares the disadvantages and advantages of the two approaches combined. The Zone Routing Protocol (ZRP) [11] is an example for hybrid reactive/proactive routing protocol. Each MN proactively maintains routes within a local region (referred to as the routing zone). MNs residing outside the zone can be reached with reactive routing. The above mentioned three approaches applied in MANET routing protocols to discovery routes can also be applied in discovering a GW node to get connection to the internet [14], [6], [12], [21], [24]. Enhancements have been made to AODV MANET routing protocol to support the three types of internet GWDAs and those algorithms are discussed in the next chapter. AODV MANET routing protocols are discussed in the following section.

### **3.2 AD HOC on demand distance vector (AODV) routing**

Ad hoc On-Demand Distance Vector, AODV, is a distance vector routing protocol that is reactive [1], [2], [8]. The reactive property of the routing protocol implies that MNs requests a route only when they need to communicate with another MN and does not require maintaining routes to destinations to which it is not communicating. AODV guarantees loop-free routes by using sequence numbers that indicate how new, or fresh, a route is. AODV requires each node to maintain a routing table containing one route entry for each destination that the node is communicating with. The discovery process is composed of Route Discovery process and Route Maintenance Process. Packets used in this routing protocol are RREQ (Route Request), RREP (Route Reply) and RERR (Route Error).

### **3.2.1 Route Discovery**

Whenever a source node desires a route to a destination node to which it does not already have a route, it broadcasts a RREQ message to all its neighbors. The neighbors update their information for the source and create reverse route entries for the source node in their routing tables. A neighbor receiving a RREQ may send a RREP, if it is either the destination or if it has an unexpired route to the destination. If any of these two cases is satisfied, the neighbor unicasts a RREP back to the source. Along the path back to the source, intermediate nodes that receive the RREP create forward route entries for the destination node in their routing tables. If none of the two cases mentioned are satisfied, the neighbor rebroadcasts (forwards) the RREQ.

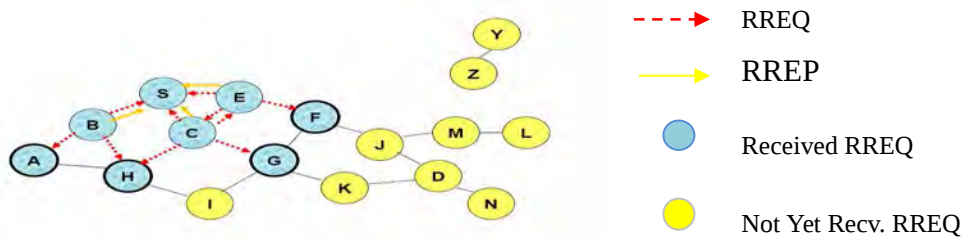
Each MN keeps a cache where it stores the source IP address and ID of the received RREQs during the last path discovery time. If a MN receives another RREQ with the same source IP address and RREQ ID during this period, it is discarded. Hence, duplicated RREQs are prevented and not forwarded.

When searching for a route to the destination node, the source node uses the expanding ring search technique to prevent unnecessary network-wide dissemination of RREQs. This is done by controlling the value of the time to live (TTL) field in the IP header. The first RREQ message sent by the source has  $TTL = TTL\_START$ . The value of TTL defines the maximal number of hops a RREQ can move through the MANET.

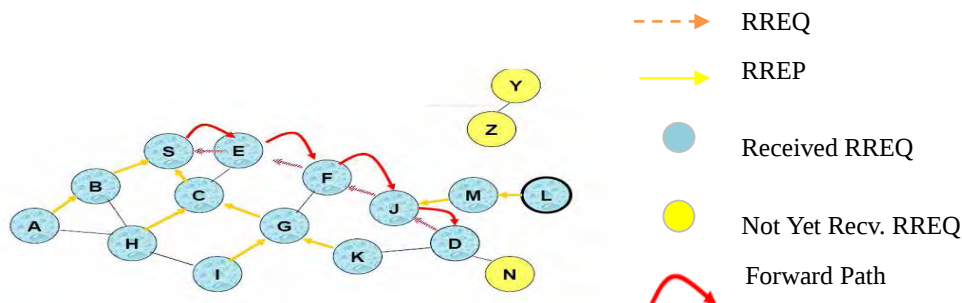
Apart from setting the TTL, the timeout for receiving a RREP is also set. If the RREQ times out without reception of a corresponding RREP, the source broadcasts the RREQ again. This time TTL is incremented by  $TTL\_INCREMENT$ , i.e. the TTL of the second RREQ message is  $TTL\_START + TTL\_INCREMENT$ . This continues until a RREP is received or until TTL reaches  $TTL\_THRESHOLD$ . If TTL reaches  $TTL\_THRESHOLD$ , a RREQ is sent with  $TTL = NET\_DIAMETER$ , which disseminates the RREQ widely, throughout the MANET. Broadcasting a RREQ

with  $TTL=NET\_DIAMETER$  is referred to as a network-wide search. If a source node does a network-wide search and still does not receive a RREP, it may try again to find a route to the destination node, up to a maximum of  $RREQ\_RETRIES$  times.

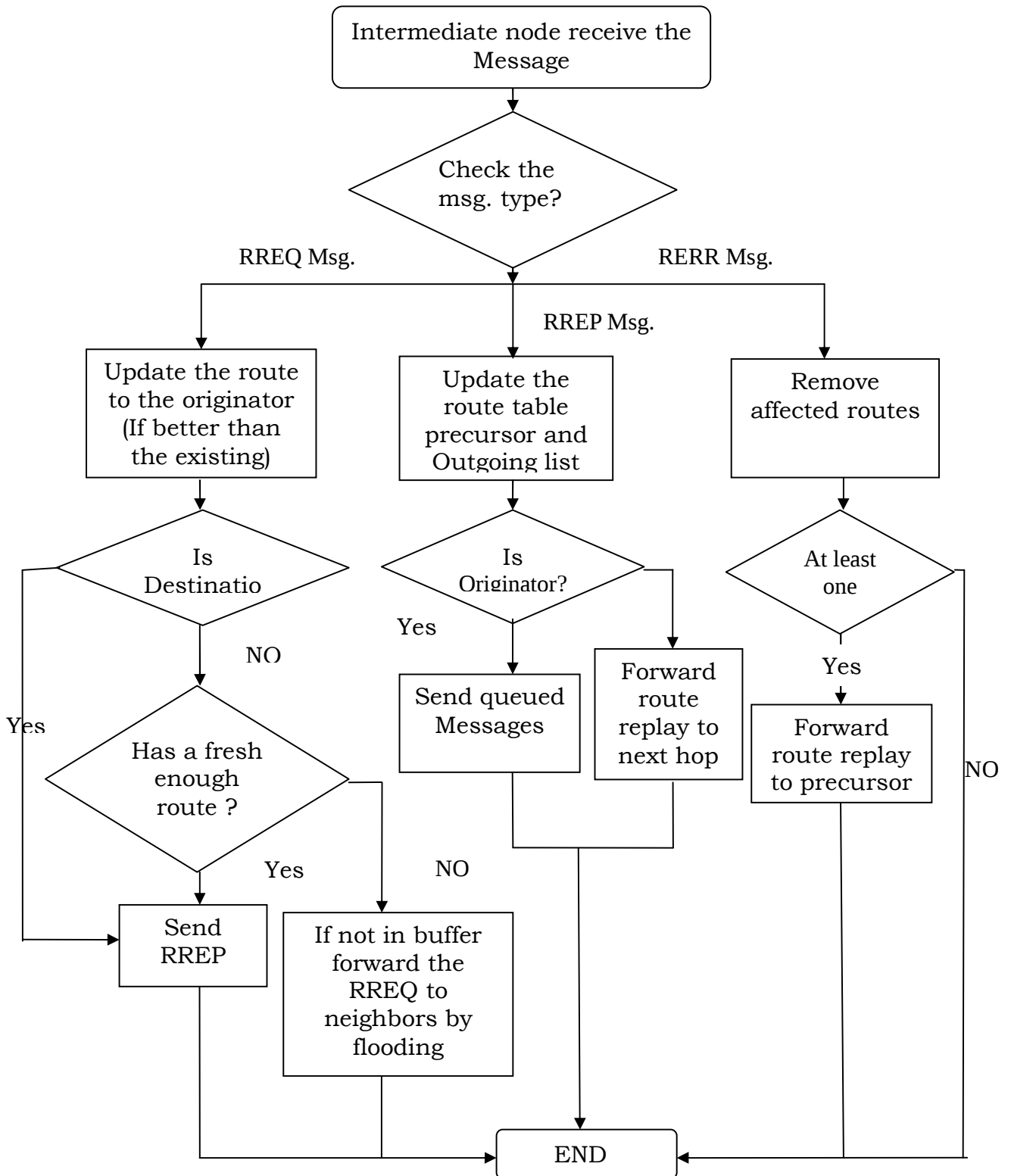
In figure 3.1, it is shows that, node "S" broadcasts a RREQ (the broken and red colored line) message to its neighboring nodes "C", "B", "E" and the neighboring nodes further broadcast the RREQ message in the Ad hoc network. The solid (yellow) lines indicate the link in reverse path (RREP). In this figure, it is shown that, MNs A, B, C, H, E, G, F and S have received the RREQ message. Figure 3.2 shows, the forward path setup towards the destination when the RREP message is unicasted to the source MN. The forward route setup is indicated by the downward curved solid arrows from "S" to "D". Figure 3.3 is a flow chart showing the packets (RREQ, RREP and RERR) processing in AODV.



*Fig 3.1: Route request and reverse path set up in AODV*



*Fig 3.2: forward path set up in AODV for destination "D"*



*Fig 3.3: Packet processing in AODV*

### **3.2.2 Route Maintenance**

When a link in a route breaks, the node upstream of the broken link invalidates all its routes that use the broken link. Then, the node broadcasts a *route error* (RERR) message to its neighbors (TTL is set to one). The RERR message contains the IP address of each destination which has become unreachable due to the link break. Upon reception of a RERR message, a node searches its routing table to see if it has any route(s) to the unreachable destination(s) (listed in the RERR message), which use the originator of the RERR as the next hop. If such routes exist, they are invalidated and the node broadcasts a new RERR message to its neighbors. This process continues until the source receives a RERR message. The source invalidates the listed routes as previously described and reinitiates the route discovery process if needed.

### **3.2.3 Advantages of AODV Routing Protocol**

Research works had shown that, using AODV in MANET routing has a lot of advantages over the other routing protocols [4], [8]. In this thesis work, the focus is on studying the effect of mobility on performance of GWDAs. So the concern is choosing the most mobility resistive MANET protocol, both for routing in the ad hoc network and its extension for GW discovery purposes. The following are advantages of AODV protocol.

- o **Minimal space complexity:** The algorithm makes sure that the nodes that are not in the active path do not maintain route information.
- o **Maximum utilization of the bandwidth:** As the protocol does not require periodic global advertisements, the demand on the available bandwidth is less.
- o **Simple:** It is simple with each node behaving as a router, maintaining a simple routing table.
- o **Coping up with dynamic topology and broken links:** Because of its reactive nature, AODV can handle highly dynamic behavior of MANETs. Additionally, when the nodes in the network move from their places and

the topology is changed or the links in the active path are broken, the intermediate node that discovers this link breakage propagates an RERR packet. And the source node re-initializes the path discovery if it still desires the route. This ensures quick response to broken links. This quality of AODV is worth considering when one has to study mobility impacts.

### **3.3 The Protocol Stack**

In this section, the protocol stack for MANETs is described. This gives a comprehensive picture of, and helps to better understand, MANETs. Figure 3.4 shows the protocol stack of MANET which consists of five layers: physical layer, data link layer, network layer, transport layer and application layer. It has similarities to the TCP/IP protocol suite. As can be seen, the OSI model's session, presentation and application layers are merged into one section, the application layer in MANET and TCP/IP suite.

OSI is a layered framework for the design of network systems that allows for communication across all types of computer systems. Because TCP/IP was designed before the OSI model, its layers do not correspond exactly to the OSI layers. The lower four layers are the same in both models but the fifth layer in the TCP/IP suite (the application layer) is equivalent to the combined session, presentation and application layers of the OSI model.

The main difference between MANET and TCP/IP suite protocol stacks lies in the network layer. MNs (which are both hosts and routers) use an ad hoc routing protocol to route packets. In the physical and data link layer, MNs run protocols that have been designed for wireless channels. Some options are the IEEE standard for wireless LANs, IEEE 802.11, the European ETSI standard for a high-speed wireless LAN, HIPERLAN 2, and finally an industry approach towards wireless personal area networks, i.e. wireless LANs at an even smaller range, Bluetooth [1]. In the simulation tool used in this thesis work, the standard IEEE 802.11 is used.

| OSI MODEL    | TCP/IP SUITE | MANET<br>PROTOCOL STACK |                   |
|--------------|--------------|-------------------------|-------------------|
| APPLICATION  | APPLICATION  | APPLICATION             |                   |
| PRESENTATION |              |                         |                   |
| SESSION      |              |                         |                   |
| TRANSPORT    | TRANSPORT    | TRANSPORT               |                   |
| NETWORK      | NETWORK      | NETWORK                 | AD HOC<br>ROUTING |
| DATA LINK    | DATA LINK    | DATA LINK               |                   |
| PHYSICAL     | PHYSICAL     | PHYSICAL                |                   |

*Figure 3.4: The OSI model, TCP/IP suite and MANET protocol stack.*

When the connection of MANET to the internet using extension of AODV MANET routing protocol is considered, the network layer is divided into two parts: The fixed network and Ad Hoc Routing in the MANET. The protocol used in the fixed network part is Internet Protocol (IP) and the protocol used in the ad hoc routing part is AODV.

In the transport layer, the User Datagram Protocol (UDP) is used in this work. The Transmission Control Protocol (TCP) is not used because different research works revealed that, TCP does not perform well in MANETs. This is because of the fact that, in wired networks, lost packets are almost always due to congestion but in MANETs, lost packets are more often caused by other reasons like link breakage due to mobility or interference [1].

The next chapter discusses in detail about the connectivity of MANETs to internet using the enhanced AODV routing protocol. The three types of GWDAs are also described.

## **Chapter Four: Internet Connectivity for Mobile Ad Hoc Networks**

This chapter investigates inter-working between MANETs and the Internet. The need and the means of Internet connectivity for MANETs are highlighted in section 4.1. In Sections 4.2 inter-working is discussed. In section 4.3 the enhanced AODV for gateway discovery and internet connectivity is described. Section 4.4 expresses the three types of gateway discovery algorithms, which are realized based on the enhanced MANET AODV routing protocol.

### **4.1 Connectivity to the Internet**

Although, autonomous, stand-alone MANET is useful in many cases, a MANET connected to the Internet is much more desirable. MNs desiring communication with the fixed internet requires global connectivity. However, all routing protocols for MANETs typically maintain routes locally within the MANET only.

In this thesis work, the access to the Internet from a multi-hop wireless network by using the extension of AODV MANET routing protocol is applied. The Internet draft “Global Connectivity for IPv6 Mobile Ad Hoc Networks” [14] describes how to provide Internet connectivity to MANETs. In particular, it explains how a MN and a GW should operate. Further, it proposes and illustrates how to apply a method for discovering gateways by extending the existing AODV routing protocol. Based on this, a lot of research works have been performed as stated in chapter two.

### **4.2 Inter-working**

Whenever a MN is to send packets to a fixed network, it must transmit the packets to a GW [3]. The protocol stacks involved during communication between a MANET and the fixed Internet node is shown in Figure 4.1. A GW acts as a bridge (not the network device) between a MANET and the Internet. Therefore, it has to implement both the MANET protocol stack and the TCP/IP suite, as shown in the figure 4.1.

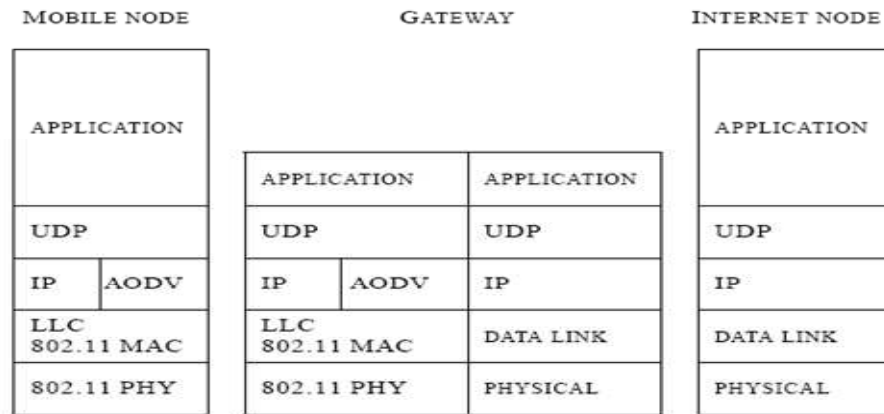


Figure 4.1: The protocol stacks used by MNs, GWs and Internet nodes.

The protocol stack used by the MN is the MANET protocol stack discussed previously on (Section 3.3). The fixed Internet node uses the TCP/IP suite. A GW node must be able to translate between these two “languages”, and understand both the architectures.

### 4.3 Enhanced AODV Protocol [14],[6]

The enhanced AODV MANET routing protocol to support the three types of GWDAs (Section 4.4) and hence internet connectivity is presented in the following sub sections.

#### 4.3.1 The Enhanced Route Request

The enhanced RREQ message contains exactly the same fields with the same functions as the ordinary RREQ message, except for a flag as shown in fig 4.2. This flag is called ‘Internet-Global Address Resolution Flag’ and is referred to as the I-flag.

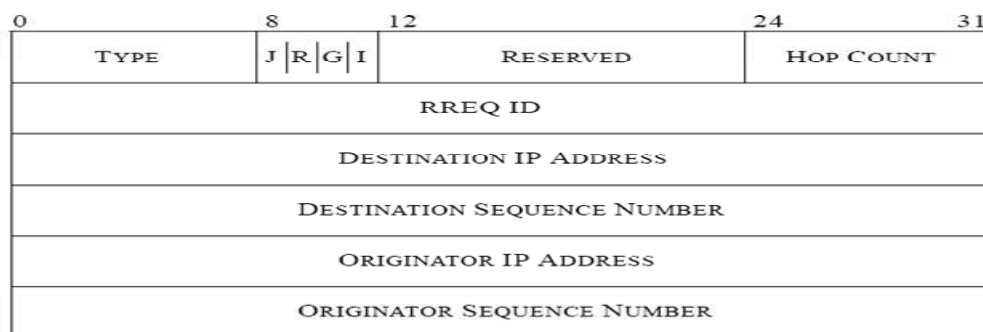


Figure 4.2: Packet Format of the RREQ\_I message.

The, I-flag is used for global address resolution and it indicates that the source

node requests global connectivity. The RREQ\_I message plays the same role as the router solicitation message of Internet Control Message Protocol (ICMP). The RREQ\_I message is used to reactively discover a gateway.

#### **4.3.2 The Enhanced Route Reply**

The enhanced RREP message contains exactly the same fields with the same functions as the ordinary RREP message, except for a flag. This flag is the same flag that has extended the RREQ message to the RREQ\_I message, namely the Internet-Global Address Resolution Flag (or the I-flag). Hence, the RREP message extended with the I-flag is referred to as the RREP\_I message. Figure 4.3 shows the format of the RREP\_I message. The, I-flag is used for global address resolution and, if set, it indicates that this RREP contains information about a gateway. The RREP\_I message plays the same role as the router advertisement message of ICMP.

|                             |           |          |           |           |    |
|-----------------------------|-----------|----------|-----------|-----------|----|
| 0                           | 8         | 11       | 19        | 24        | 31 |
| TYPE                        | R   A   I | RESERVED | PREFIX SZ | HOP COUNT |    |
| DESTINATION SEQUENCE NUMBER |           |          |           |           |    |
| DESTINATION IP ADDRESS      |           |          |           |           |    |
| ORIGINATOR IP ADDRESS       |           |          |           |           |    |
| LIFETIME                    |           |          |           |           |    |

*Figure 4.3: The format of a Route Reply message extended with the I-flag.*

#### **4.3.3 The Gateway Advertisement (GWADV)**

GWADV is approximately a RREP\_I message but it is extended to have a GWADV\_ID, just like the RREQ ID of the RREQ packet in AODV MANET routing protocol. The GWADV\_ID helps to avoid duplicated advertisement messages. When a MN receives a GWADV, it first checks to determine whether a GWADV with the same originator IP address and GWADV\_ID already have been received during the last broadcast ID save seconds (i.e. 6 Sec). If such a GWADV message has not been received, the message is rebroadcasted. Otherwise, if such a GWADV message has been received, the newly received GWADV is discarded.

Hence, duplicated GWADVs are not forwarded and the advertisement is flooded throughout the network or for limited number of hops or else not flooded at all depending on the type of GWDA implemented.

#### **4.3.4 The Default Route (Route to the Gateway)**

A MN needs to learn the location and address of a GW to be able to have access to the Internet. In other words, the MN needs a route to a gateway, which it uses as its default route to send packets to the Internet. This GW information can be obtained in three different ways. One option is to rely on periodically advertised messages from the gateway (GWADVs), or by sending a RREQ\_I to the ALL\_MANET\_GW\_MULTICAST address (i.e. by sending to the GW nodes' group address). There is also a third option, for the sake of updating the default route entry, the GW nodes are made to reply RREQ messages with RREP\_I messages, as a result, a MN can get default route by sending RREQ message to the gateway. However, this happens only when a MN is performing radial ring search before it gets the information, whether the destination node is within the ad hoc network or in the fixed network. The first two different approaches specify the different methods for GW discovery as described in the next Chapter.

#### **4.3.5 The Expanding Radial-Ring Search**

Assume that a MN 'S' wants to communicate with another node 'D' and that 'S' does not have any route to 'D' in its routing table. Hence, 'S' does not know whether 'D' is a MN (located within the MANET) or a fixed node (FN) (located on the Internet).

Using AODV as the ad hoc routing protocol, 'S' broadcasts a RREQ, requesting for a route to 'D'. If 'D' is a MN, the node itself or another MN with a fresh route to it will unicast back a RREP to MN 'S'. However, if 'D' is a FN, no MN will send a reply to 'S'. The expanding radial-ring search is performed before assuming the location of 'D' as FN in the internet. To be absolutely sure that 'D' is not a MN located within the MANET, 'S' must do, at least, one network-wide search. If no RREP message is returned, 'S' assumes that 'D' is a FN. Hence, the packets are

sent to the Internet by using the default route.

#### **4.3.6 Internet-Gateway Operation when Receiving Requests**

When a GW receives a RREQ, it looks in its routing table searching for the destination IP address specified in the RREQ message. If the address is not found in the routing table, the GW sends a RREP\_I back to the originator of the RREQ to create or update the default route information at the requesting node. On the other hand, if the GW finds the host route in its routing table, it does not unicast back a RREP\_I to the originator of the RREQ. This is because; the requesting node will assume that the destination node is inside the MANET (which is wrong). In this way, a MN may obtain or update its default route. If the MN is to communicate with the Internet later, this default route can be used and hence, the MN does not have to send another request message in order to find a route to a gateway.

If a GW receives several RREQs for the same destination, a GW should send a RREP\_I as long as it does not find the destination address in its routing table. But since expanding ring search is used, a GW may receive several RREQs for the same destination address. The question is, should the GW reply every RREQ with a RREP\_I or only some of them? The chief advantage of sending a RREP\_I for every received RREQ is that the route to the GW and the default route gets updated. The chief disadvantage is that network resources are used. However, since the RREP\_Is are unicast and not broadcasted to the requesting node, there will not be that much traffic generated.

#### **4.3.7 The Routing Table**

Another issue that is worth discussing is how the routing table should change after a network-wide search without receiving any corresponding RREP. Assume that a source MN has done a network-wide search, without receiving any corresponding RREP. Hence, the source node assumes that the destination node is a FN located on the Internet.

After this assumption is made, the source node sends its data packets using the

default route. What the source node actually has to do is, create a route entry for the destination node in its routing table as shown in table 4.1. If the route entry for the fixed destination node is not created in its routing table, the source node cannot find the address to the FN when the next data packet is generated and hence, the source needs to do another time consuming network-wide search.

Although it is necessary for the source node to create a route entry for the FN in its routing table, there is a disadvantage. The disadvantage is that, a MN will have to create a route entry for every FN that it wants to communicate with, in its routing table. To further elaborate this point, let us consider the case when a MN desires to communicate with many FNs. No doubt that it's routing table will grow rapidly. However, this is not as alarming as it sounds. In AODV MANET routing protocol, the routes that are not used will expire and eventually be deleted after a certain time, preventing the routing table growth indefinitely.

| <b>Destination</b> | <b>Next Hop</b> |
|--------------------|-----------------|
| FN                 | Default         |
| Default            | Gateway         |
| Gateway            | MN              |

*Table 4.1. The routing table of a MN.*

Table 4.1 shows how the routing table of a MN "S" should look like after creation of a route entry for a FN. If "S" wants to communicate with the FN,"S" sends its data packets to its neighboring MN\_A. When MN\_A receives the data packet it searches its routing table to see if it has a valid route to FN. If a valid route to FN is found, the data packets are sent to the next hop specified by the route entry. On the other hand, if MN\_A does not find a valid route to FN and if the destination is a FN located on the Internet, MN\_A creates (or updates) a route entry for FN in its routing table. Next, it forwards the data packets to a GW which forwards them toward their destination.

#### **4.3.8 Operation of Intermediate Node on Receiving RREQs**

When an intermediate MN receives a RREQ message, it searches its routing table for a route to the destination. If the destination is a FN, the intermediate node must not send a RREP back to the originator of the request message even if the route is found. Because if the intermediate node sends a RREP back to the originator of the RREQ message, the originator considers that the destination is a MN that can be reached via the intermediate node. It is important for the originator of the RREQ to know that the destination is a FN and not a MN, as they are processed differently.

#### **4.3.9 Unreachable Gateway**

If a MN cannot reach any gateway, although the destination is a FN, it multicasts a RREQ\_I message to the IP address for the group of all gateways in the MANET. However, since GW is unreachable for the MN, the RREQ\_I message is not received by a GW (any gateway). MN uses the expanding ring search technique when it broadcasts RREQ\_I messages, but not even a single RREQ\_I message with the TTL value set to NET\_DIAMETER is received by any gateway, assuming MN cannot reach any intermediate MN that can forward the RREQ\_I message on its behalf. After doing a network-wide search without receiving any corresponding RREP\_I message from any gateway, MN pauses for a while. When the pause is finished, MN does another network-wide search and pauses again if no RREP\_I is received. This procedure continues until MN moves close to a GW or an intermediate MN, so it can receive a RREP\_I from a gateway. When a GW is found, MN sends its data packets to the FN through the found gateway.

### **4.4 Gateway Discovery**

There are three types of GWDAs depending on the GW configuration phase initiation and also on the method of route update. If the configuration phase is initiated by the gateway, proactive method is used. But, if the initiation is made by a MN, reactive method is used. The combination of these two methods is

called hybrid proactive/reactive method. The basic difference between the algorithms is highlighted below. All the enhancements discussed in Section (4.3) that are made on the AODV routing are mandatory for the implementation of the algorithms. The three GWDAs are discussed in the following sections.

#### **4.4.1 Proactive Gateway Discovery**

The proactive GW discovery is initiated by the GW itself. The GW periodically broadcasts a GWADV message which is transmitted after expiration of the gateway's advertisement interval timer that is the time between two consecutive advertisements must be chosen with care so that the network is not flooded unnecessarily. All MNs residing in the gateway's transmission range receive the advertisement.

Upon receipt of the advertisement, the MNs that do not have a route to the GW create a route entry for it in their routing tables. MNs that already have a route to the GW update their route entry for the gateway. Next, the advertisement is forwarded by the MNs to other MNs residing in their transmission range. To assure that all MNs within the MANET receive the GW advertisement. The number of retransmissions is determined by network diameter. However, this will lead to enormously many unnecessary duplicated advertisements. This is disadvantage, which is general for all proactive approaches.

Additionally, the fact that the message is flooded through the whole MANET periodically is a very costly operation. Limited resources in a MANET, such as power and bandwidth, will be excessively used.

#### **4.4.2 Reactive Gateway Discovery**

The reactive GW discovery is initiated by a MN that is to initialize or update information about the gateway. The MN broadcasts a RREQ\_I to IP address for the group of all gateways in a MANET. Thus, only gateways are addressed by this message and only they process it. Intermediate MNs that receive the message just forward it by broadcasting it again. Since the message format is RREQ, which has a unique request id field duplicated RREQ\_Is are discarded. Upon receipt of a

RREQ\_I, a GW unicasts back a RREP\_I which, among other things, contains the IP address of the gateway.

The advantage of this approach is that RREQ\_Is are sent only when a MN needs the information about reachable gateways. Hence, periodic flooding of the complete MANET, which has obvious disadvantage, is prevented.

#### **4.4.3 Hybrid Gateway Discovery**

A third approach is called hybrid, (i.e. combination of proactive and reactive) GW discovery method. For MNs in a certain range around a gateway, proactive GW discovery is used. MNs residing outside this range use reactive GW discovery to obtain information about the gateway.

The GW periodically broadcasts a GWADV message, which is transmitted after expiration of the gateway's advertisement timer. All MNs residing in the gateway's transmission range receive the GWADV. Upon receipt of the message, the MNs that do not have a route to the GW create a route entry for it in their routing tables. MNs that already have a route to the GW update their route entry for the gateway. Next, the GWADV is forwarded by the MNs to other MNs residing in their transmission range. The maximal number of hops a GWADV can move through the MANET is defined by the zone radius of Advertisement Zone. This defines the range within which proactive GW discovery is used.

When a MN residing outside this range needs GW information, it broadcasts a RREQ\_I in a similar fashion as it was discussed for the reactive GWDA. MNs receiving the RREQ\_I just rebroadcast it. Upon receipt of this RREQ\_I, the GW unicasts back a RREP\_I.

This chapter presented the different enhancements made on AODV routing protocol and techniques applied for getting connectivity to the internet. It has also discussed the different algorithms used to discover the internet-gateway. The next chapter will discuss the effect of node mobility on performance of routing protocols and GWDAs and the different types of mobility models.

## **Chapter Five: Impact of Node Mobility and Mobility Models**

### **5.1 Mobility Influence on the Gateway Discovery Algorithms**

In MANET, with the dynamic topology network, the end to end connection is established without the use of base station (static intermediate entity). If a mobile node and a gateway node are found to be out of the communication range of each other, the route to internet between them will use intermediate nodes. Since in MANET, nodes are free to move, the established route to the GW through intermediate nodes is affected by the movement of the nodes on the path. When a node which is part of a route to the internet moves, it may go out of the radio range of communication coverage of the immediate neighbor node, which is part of the same route. If the two nodes are out of the communication range of each other, the link between them will fail. This in turn causes the failure of the route to the internet. As a consequence of the route failure, packets to be exchanged with the internet through the GW node will be dropped. Besides, in order to continue the communication with the internet, a new path to the GW needs to be discovered using one of the three GWDAs, which means additional time for discovery. The time spent to find a new route also increases the end to end delay of packet transmission.

The effect of nodes' mobility depends on the relative speed between nodes. The relative speed in turn depends on the speed of each node and its direction of movement which is governed by the mobility pattern model used. This shows the network connectivity depends on the type of nodes' movement pattern followed.

### **5.2 Mobility Models**

Mobility models are used to describe the movements of nodes. Each model gives an algorithm that is used to randomize the movement of nodes. Mobility pattern is the actual set of movements that result from applying the mobility model to

one or more nodes. There are two types of mobility models used in the simulation of networks: traces [36] and synthetic models. Traces are those mobility patterns that are observed in real life systems. Traces provide accurate information, especially when they involve a large number of participants and an appropriately long observation period. However, new network environments (e.g. ad hoc networks) are not easily modeled if traces have not yet been created. In this type of situation, it is necessary to use synthetic models. Synthetic models attempt to realistically represent the behaviors of MNs without the use of traces.

The synthetic models are divided into two categories, the old mobility models and the new realistic mobility models. The old mobility models are in turn divided into two: the entity mobility models and the group mobility models. Entity mobility models give a statistical movement description of a single node. This model can be used to generate the mobility pattern several times. Group mobility model randomizes the movement of nodes such that the movement of nodes in the same group is correlated. One typical way of accomplishing this is to have a group center for each group, which moves and then allows the node positions to vary around that point. The following three sections describe the commonly used existing mobility models in Ad hoc.

### **5.2.1 Entity Models**

Entity models give movement patterns to single nodes, which are independent of each other. Out of the available entity mobility models, the Random Walk Mobility Model and the Random Waypoint Mobility Models are the two most common mobility models used by researchers.

#### **5.2.1.1 Random Walk Mobility Model [29]**

The Random Walk Mobility Model was developed to impersonate the erratic movement of entities in nature. A MN moves from its current location to a new location by randomly choosing a direction and speed in which to travel. The new speed and direction are both chosen from pre-defined ranges, [speedmin; speedmax] and  $[0; 2\pi]$  respectively. Each movement in the Random Walk Mobility

Model occurs either after a constant time interval  $t$  or after travelling a constant distance  $d$ . and at the end of which a new direction and speed are calculated. If a MN which moves according to this model reaches a simulation boundary, it bounces off the simulation border with an angle determined by the incoming direction. The MN then continues along this new path. A random walk, on a one or two-dimensional surface, returns to the origin with complete certainty, i.e., a probability of 1.0. This characteristic ensures that the random walk represents a mobility model that tests the movement of entities around their starting points, without worrying about the entities wandering away and never to return. The Random Walk Mobility Model is a widely used mobility model, and is sometimes referred to as Brownian motion.

The Random Walk Mobility Model is a memory less mobility pattern because it retains no knowledge concerning its past location and speed values. The current speed and direction of a MN is independent of its past speed and direction. This characteristic can generate unrealistic movements such as sudden stops and sharp turns. If the specified time (or specified distance) a MN moves using the Random Walk Mobility Model is short, then the movement pattern is random roaming pattern restricted to a small portion of the simulation area.

### **5.2.1.2 Random Waypoint Mobility Model [31]**

The Random Waypoint Mobility Model includes pause time between changes in the direction and/or speed. A MN begins by staying in one location for a certain period of time (i.e., a pause time). Once this time expires, the MN chooses a random destination in the simulation area and speed that is uniformly distributed between  $[speedmin, speedmax]$ . The MN then travels towards the newly chosen destination at the selected speed. Upon arrival, the MN pauses for a specified time period before starting the process again. The movement pattern of a MN using the Random waypoint Mobility Model is similar to the Random Walk Mobility Model if pause time is zero and the model have equal minimum speeds as well as equal maximum speeds for all MNs. The Random Waypoint Mobility Model is also a widely used mobility model.

### **5.2.2 Group Mobility Models**

The entity mobility models represent multiple nodes whose actions are completely independent of each other. However, there are many situations where it is necessary to model the behavior of MNs as they move together. In order to model such situations, a group mobility model is needed to simulate this cooperative characteristic. The commonly used model in this group is the reference point Group Mobility Model (RPGM).

#### **5.2.2.1 Reference point Group Mobility Model (RPGM)[30]**

The RPGM model represents the random motion of a group of MNs as well as the random motion of each individual MN within the group. Group movements are based upon the path traveled by a logical center for the group. The motion of the group center completely characterizes the movement of its corresponding group of MNs, including their direction and speed.

Individual MNs randomly move about their own pre-defined reference points, whose movements depend on the group movement.

Mathematically, the relation between the individual node  $I$  and the group movement is known as follows:

$$\mathbf{V}_i^t = \mathbf{V}_{\text{group}}^t + \mathbf{RM}_i^t \quad (5.1)$$

Where  $\mathbf{V}_i^t$  is the velocity of a group member  $I$ ,  $\mathbf{V}_{\text{group}}^t$  is the velocity of the group, and  $\mathbf{RM}_i^t$  is a random vector deviated by group member  $I$  from its reference point.

### **5.2.3 New Mobility Models**

The majorities of the commonly used mobility models are based on the random walk model and cannot reflect the reality [33]. With increasing awareness of the limitations of those models [34] and the importance of a realistic mobility model in the evolution of MANET protocols [35], there are some research works conducted on realistic mobility models [33], [13], [36]. In the following section, two available models are presented.

### **5.2.3.1 Obstacle Mobility Model [4]**

It is a geographically constrained mobility model. Rather than moving in pen space, the MN may face an obstacle. In real life movement, to avoid obstacles on the way, the MN is required to change its trajectory. Therefore, obstacles do affect the movement behavior of MNs. Moreover, the obstacles also impact the way radio propagates. For example, for the indoor environment, typically, the radio system could not propagate the signal through obstacles without, severe attenuation. For the outdoor environment, the radio is also subject to the radio shadowing effect. When integrating obstacles in the mobility model, both its effect on node mobility and on the radio propagation should be considered.

The obstacle mobility model [33] considers the effect of obstacles on the nodes mobility. The model works by having obstacles in the form of rectangular boxes that are randomly placed on the simulation field. People in real life may follow predefined pathways between building, the location of those building or obstacles, a Voronoi graph [32] is computed to construct the pathways that are equidistant from the nearby buildings. This observation is consistent with the common sense that the pathways tend to lay halfway in-between the adjacent buildings. Moreover, in this model, the nodes (e.g., students on campus) are allowed to enter and exit buildings.

Once the pathway graph is defined, the movements of MNs are restricted on the pathways. Thus, the MNs are likely to travel in a semi-definitive (i.e. pseudo random) way. After the MN randomly chooses a new destination on the pathway graph, it moves towards it by following the shortest path through the predefined pathway graph. This shortest path is calculated by the Dijakstra's algorithm in the Voronoi Diagram [32].

In this model, a MN is required to choose a proper movement trajectory to avoid running into such obstacles. Moreover, when the radio propagates through an obstacle, the signal is assumed to be fully absorbed by the obstacle. More specifically, if an obstacle is in-between two nodes, the link between these nodes is considered broken until one moves out of the shadowed area of the obstacle.

### **5.2.3.2 Community Based Mobility Model [13]**

The concept behind the development of Community based mobility model is that, in MANETs, mobile devices are usually carried by humans, so the movement of such devices is necessarily based on human decisions and socialization behavior. For instance, it is important to model the behavior of individuals moving in groups and between groups, as clustering is likely in the typical and ad hoc networking deployment scenarios like disaster relief teams, platoons of soldiers, group of vehicles, workers in work places etc.

The model is designed to capture the above mentioned human behavior. It is based on group mobility that is heavily dependent on the structure of the relationships among the people carrying the devices. Existing group mobility models fail to capture this social dimension [34].

The model uses the concept of mathematical network called social networks [13], where the interaction level between the nodes can be described by weighted graphs. The model uses an interaction matrix, called  $m$ , to store the interaction graph weights; with an entry  $m_{ij}$  represents the interaction level between node  $i$  and  $j$ . The interaction level ranges from 0 to 1, where 0 stands for no interaction and 1 for maximum. If the interaction is greater than or equal to 0.25, a social network connection between the nodes will be formed. A connectivity matrix is used to represent the connection information between the nodes; 0 for the absence of connection and 1 for the presence of the connection.

The simulation area is divided into grids. At the beginning of the MNs movement, a goal is assigned to the host. More formally, a host is associated to a certain square 'S' if its goal is inside 'S'. The goal is simply a point on the grid which acts as final destination of movement like in the Random Way-Point model, with the exception that the selection of the goal is not random.

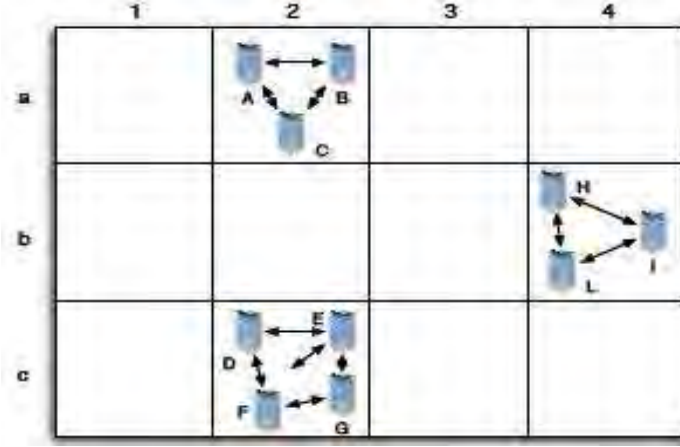


Figure 5.1: Example of initial configuration

The network is initialized using the principle of Caveman model [37]. At the start of simulation, there are different isolated communities (i.e. K differently fully connected graphs) as shown in the figure 5.1. Then after, the edges between different caves (communities) are re-wired with certain probability called rewiring probability ( $w$ ) with the maximum values of 0.2[13].

Once the node reached its goal point (square), a new goal will be established based on a value called social attraction calculated as shown in equation 5.2, which is, attraction to the node from other squares.

The social attraction of a square is a measure of its importance in terms of the social relationship for the host taken into consideration. The social importance is calculated by evaluating the strength of the relationships with the hosts that are moving towards that particular square (i.e. with the hosts having a current goal inside that particular square).

$$SA_{p,qi} = \frac{\sum_{j=1, j \in C_{s,p,q}}^n m_{i,j}}{w}$$

(5.2)

Where  $W$  is number of hosts in the square,  $m_{i,j}$  is an interaction value for  $i$  and  $j$  hosts moving towards the square, and  $C_{s,p,q}$  is the set of hosts associated with the square  $SA_{p,qi}$  ( $p, q$  are the positions of the square).

In real life, humans will be part of different groups, based on their relationship with the groups, in a single day. For example, in day time parents and children of a family may be part of their working and school groups respectively. But, at the end of the day, all will leave the groups where they have been and will form a new group called “family”. To model this phenomenon, the model lets different social groups to be configured in a single simulation time. With the reconfiguration interval( $r$ ), the model controls how long a node can be a part of its present group. In this chapter, the impact of node mobility and some of the available mobility models which are being used by the researchers are discussed.

In this thesis, the community based mobility model is used for analyzing the effect of mobility on the performance of GWDAs. This model is mandatory for this thesis work due to the fact that it is a realistic model that well represents peoples' socialization behavior. The next chapter presents the simulation work related information.

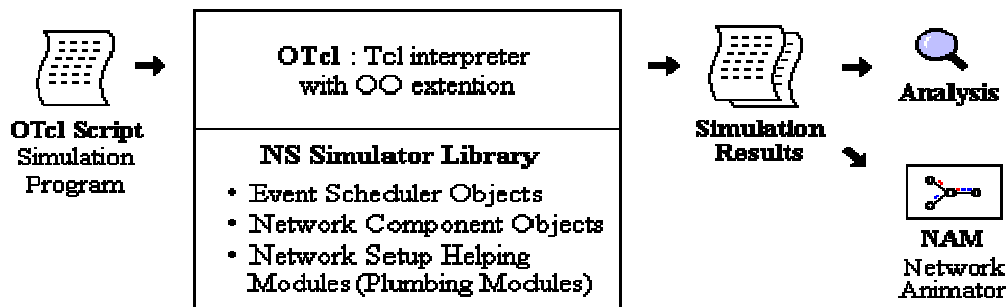
## Chapter Six: Simulation System

In this chapter, a brief overview of NS2 simulator, the simulation model, simulation environment, performance metrics, mobility models and traffic types, simulation parameters and finally the scenario designs used for meeting the purpose of this thesis work are presented.

### 6.1 Network Simulator (NS2) Overview

Network Simulator 2(NS2), is a discrete event NS. It has been developed by the University of California at Berkeley and the VINT project [7]. It is popular in academia for its extensibility (due to its open source model) and plentiful online documentation. NS2 is popularly used in the simulation of routing and multicast protocols, among others, and is heavily used in researches based on ad hoc networks. NS supports an array of popular network protocols, offering simulation results for wired and wireless networks. NS2 is licensed for use under version 2 of the GNU General Public License. [7]

NS2 supports two languages, system programming language C++ for detail implementation and scripting language TCL for configuring and experimenting with different parameters quickly. NS-2 has all the essential features like abstraction, visualization, emulation, and traffic & scenario generation.



*Fig 6.1: Simplified Users' View of NS2*

## **6.2 Simulation Model**

The simulation model applied in this thesis work takes the traffic connection to hosts in the fixed network through the GW nodes and the community based mobility model of the mobile ad hoc nodes as an input to the simulation environment (i.e. tcl script, three types of GWDAs, the network simulator and animator etc) to generate trace files. The trace file has been used to measure each of the metrics. And, analysis has been performed to find out the effect of mobility on the performance of GWDAs and also to find out relatively, the less affected GWDA.

### **6.2.1 Simulation Environment**

The simulations have been conducted on an Intel Pentium IV; 2GHz core 2 Duo Processor, 2GB of RAM running Suse Linux 10.3. NS2 version 2.29 has been used for implementing the GWDAs and for analyzing their performance with respect to mobility. NS2 is chosen basically, because of the fact that, it is the most popular simulator in the area with rich functionalities and additionally, most of the related thesis works have been performed by using this tool.

### **6.2.2 Simulation Setup**

The Simulation environment is setup, by placing two GW nodes, which are fixed and are connected to two routers on the fixed network. Each router is connected to a host in the fixed network. The routers are also connected to each other to facilitate routing from any GW to any host in the fixed network. The GW nodes are located at (200,200) and (200,800). This is common for all the three algorithms. The 600m distance between the gateways facilitates MNs to get connection to the internet in shorter handoff time. The gateways are placed at the bottom of the simulation area to get connection to the internet through a number of hops, so that the effect of mobility of intermediate nodes can also be observed. 30 MNs have been considered. Out of which, five of them are communicating with the fixed nodes through the gateways (Figure 6.2,).

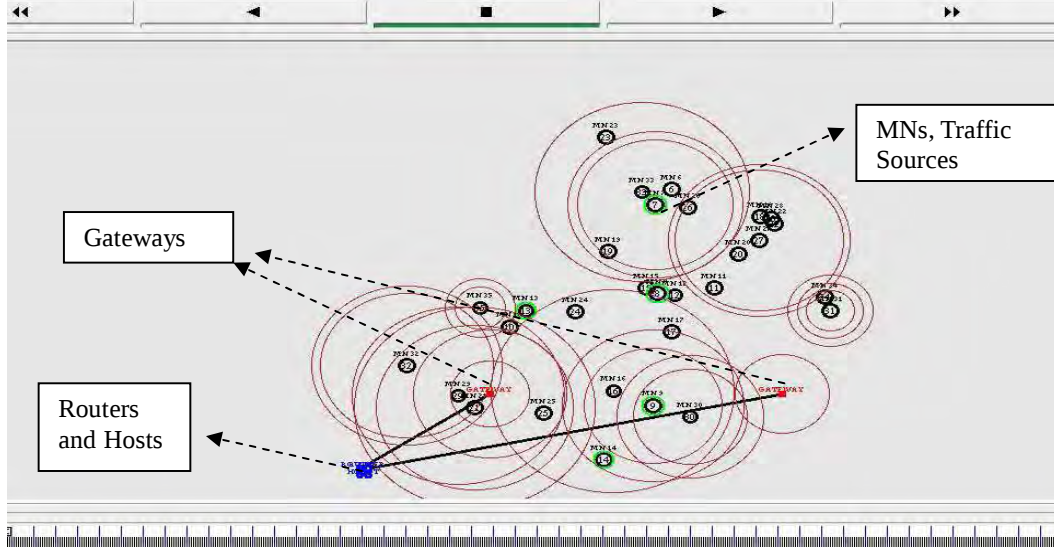


Figure 6.2: Screen shot of the simulation Environment on NS2

### 6.2.3 Performance Evaluation Metrics

The effect of mobility of MNs on the performance of AODV based internet GWDAs is studied in this thesis work. For this purpose all relevant metrics for the sake of a thorough analysis has been implemented. The Metrics description and the approach followed in this work to calculate the values directly from trace files are given below.

1. *Packet Delivery Ratio*: Shows the fraction of user data packets that were successfully delivered. In this case, it is calculated as the ratio of packets received by the gateways ( $\#p_{rcv\_GW1}$  and  $\#P_{rcv\_GW2}$ ) to the number of CBR data generated by the source nodes ( $\#P_{sent}$ ).

$$PDF = \frac{\#p_{rcv\_GW1} + \#P_{rcv\_GW2}}{\#P_{sent}} \quad (6.1)$$

2. *Average End-to-End Delay*: Shows the average time data packets spent to reach to the destinations (i.e. Node processing + queuing delay + Transmission and propagation delay). The summation of difference between

packet sending ( $P_{start\_time}[i]$ ) and packet receiving ( $P_{end\_time}[i]$ ) time for each data packets is calculated and is divided by the total number of packets generated (N).

$$Avg.e - eDelay = \frac{\sum_{i=1}^N (P_{end\_time}[i] - P_{start\_time}[i])}{N} \quad (6.2)$$

3. *GW Discovery Time*: Time spent to discover a valid route to a GW node. There can be a lot of discoveries for a single source node within one simulation time due to frequent disconnections. For calculating Discovery Time three cases have been considered

3.1 *Expanding Radial Ring Search Time*: Search performed with in the ad hoc network before using the default route to the internet. The search is performed by sending a RREQ packet assuming that the destination is within the ad hoc network at TTL values 1, 3, 5,7,30 and if no RREP is obtained the Default route is used. Time spent between the first RREQ and the last RREP\_I from the GW before starting traffic connection through the default route is measured for all the algorithms.

3.2 *Average Discovery Time*: The summation of all individual discovery time ( $DiscT$ ) within one simulation run divided by the number of discoveries(S) and averaged for the five sources. This shows the average time, a particular discovery algorithm takes to discover a gateway. The formula applied is shown below.

$$Avg.DiscT = \frac{\sum_{j=1}^S DiscT_j}{\sum_{i=1}^5 \left( \frac{S}{S} \right)} \quad (6.3)$$

3.3 *Percentage of Discovery Time*: The summation of all individual discovery times ( $DiscT_j$ ) within one simulation run in percentage out of the total

simulation time (300) and averaged for the five sources. This indicates the total time a particular GWDA spends for GW discovery purpose. The following formula has been applied. '5' stands for five sources, '300' is simulation time in seconds and 's' indicates number of discoveries.

$$DiscT\% = \frac{\sum_{i=1}^5 \sum_{j=1}^S DiscT_j}{5(300)} (100) \quad (6.4)$$

4. *Disconnection Time*: time spent from occurrence of error at the source node or at an intermediate node, until the source node finds another to the gateway. It can also be expressed as the summation of discovery time and error processing and propagation time.

4.1. *Error Processing and Propagation Time*: Time spent from, occurrence of error either at the source node or at an intermediate node, until the source node gets GWADV or until it sends a RREQ\_I after generating its own error message or receiving the error message from the intermediate nodes.

4.2. *Percentage of Disconnection Time*: shows the total time the source node stays disconnected from the GW in percentage out of the total simulation time. The following formula has been applied. '5' indicates number of sources, 'R' is number of errors that occurred and '300' is the simulation time in seconds. ' $DiscT_j$ ' refers to a MN'S one time discovery duration, while ' $ErrorPT$ ' refers a MN's Error Processing and Propagation Time.

$$DisconnT\% = \frac{\sum_{i=1}^5 (\sum_{j=1}^S DiscT_j + \sum_{r=1}^R ErrorPT_r)}{5(300)} (100) \quad (6.5)$$

5. *Handover Time*: time between the last packets received at one GW until the first packet is received at the other gateway.

5.1. *Percentage of Handover Time*: shows the total time the algorithm

spends for handover from one GW to the other, in percentage out of the total simulation time. The following formula has been used. 'Q' indicates number of handover, while  $ht_j$  represents a MN's one time handover time duration from the possible Q time handovers within one simulation run.

$$HT \% = \frac{\sum_{i=1}^5 \sum_{j=1}^Q ht_j}{5(300)} (100) \quad (6.6)$$

5.2. *Average Handover Time*: shows the average time required to

handoff from GW to the other. 'Q' indicates number of handover, while  $ht_j$  represents a MN's one time handover time duration from the possible Q time handovers within one simulation run.

$$Avg .HT = \frac{\sum_{i=1}^5 \left( \frac{\sum_{j=1}^Q ht_j}{Q} \right)}{5} \quad (6.7)$$

6. *Normalized Routing Load*: is the ratio of routing packets at the network layer to the sum of data packets received at the two gateways. NRL Indicates how many control packets has been applied to successfully deliver a data packet.

$$NRL = \frac{Total\_number\_of\_Control\_Pakcets\_at\_Network\_Layer}{Total\_Number\_Data\_Packets\_Received\_at\_the\_two\_Gateways} \quad (6.8)$$

7. *Routing Overhead*: shows the total number of routing packets at the network layer. Number of control packets (i.e. the number of RREQ, RREP, RREQ\_I, RREP\_I, and GWADV) is considered in this work.

#### **6.2.4 Mobility Model and Traffic Type**

The movement patterns of MNs should reflect the movement of people and accordingly a community based mobility model has been used, which is

mandatory with respect to the goal of this research work. To avoid the effect of mobility on higher layer protocols like TCP and for the sake of simplicity, a constant bit rate (CBR) traffic type with a UDP transport layer protocol used in the simulation.

### **6.2.5 Communication Model**

The communication is performed by five different MNs from the MANET to hosts in the wired network through the two gateways. Each MN generates traffic every 0.2 seconds and the size of each packet is 512byte. The traffic connection pattern is generated by using NS2's traffic generator tool.

### **6.3 Simulation Parameters**

The followings are the parameter setting of community based mobility model applied in the simulation. The variable parameters are written in italic font.

- **-n** (number of MNs)=30
- **-t** (Total Simulation Time in Seconds)=300Sec
- **-r** (Reconfiguration Interval in Seconds, interval for the formation of different social groups within the simulation time) = 100Sec.
- **s** (Lower bound speed of nodes in m/s) = 0m/s
- **S** (Upper bound speed of nodes in m/s) = 5m/s
- **-p** (Connection threshold,  $p = 0$  implies no interaction and  $p = 1$  means full or maximum interaction.) = 0.25

In this work  $p=0.25$ , this means when  $p$  is greater than 0.25 a social network connection between the nodes will be formed

- **-X**(Side length  $x$  coordinate in meter) = 1000m and 1200m
- **Y**(Side length  $y$  coordinate in meter) = 1000m and 1200m
- **R** (Number of rows) = 4
- **C** (Number of columns)=4
- **-T** (Transmission Range in meter)=250m
- **-w** (Rewiring Probability, indicates strength of social relationship)  $w=0.01$ (low social interaction),  $w=0.2$  (high social interaction)
- **-g** (Seed of the random number generator)=1,2,3,4,5,6,7,8,9,10  
10 different seed values have been used for 10 different runs.

- **-G** (Number of Groups, Initial number of isolated communities in the Grid area) =16.
- **-c**(Number of travelers)=21
- **-v** (Travelers Speed)=0,5,10,15,20,25,30m/s
- **a** (Girvan Newman algorithm, helps to identify communities from weighted graphs)=Off,
- **-d** (deterministic selection of nodes)=on
- **-A** (sets the collocation traces)=on
- **-b** (sets the communities traces ) =off

Other parameters used in the simulation are as follows

- GW Advertisement interval for proactive and hybrid GWDA=5Sec.
- Advertisement zone for hybrid GWDA=3 hops.
- Network Diameter=30 hops
- GW information life time=10Sec.
- Number RREQ\_I retries=4 times

Two-Ray Ground Reflection Approximation radio propagation model with range of 250m is used for all MNs. The radio model is based on the Lucent Technologies Wave-LAN 802.11 product, providing a 2Mbps transmission rate. The four scenarios are performed for ten runs with 300seconds for each in order to satisfy a 90% confidence interval.

## **6.4 scenarios**

For accomplishing the purpose of this thesis work, four scenarios has been designed and applied. The parameters that are varied for scenario designs are: the dimension of the simulation area, the travelling nodes' speed and the rewiring probability. The dimension of the simulation area should be determined in such a way that, internet connection through the GW nodes can be established using a number of hops. This facilitates to study, the effect of the mobility of the source nodes and also the the intermediate nodes on performance of the discovery algorithms. On top of that, by keeping the number of sources constant, incrementing the dimension will reduce the density of nodes in the simulation area. In such an environment, MNs can easily be outside the radio range of each other and can help to study the effect of mobility. Additionally, this helps to observe the effect of mobility beyond other factors like interference. The traveling nodes' speed is directly related to the movement of nodes in the simulation area. MNs moving at high speeds can easily move away from each other and result in failure of links between them. In the community based mobility model, there is a parameter called rewiring probability, which determines the strength of social (high and low) interaction among individuals and among communities. As the mobility of people is highly dependent on their social interaction and cooperation, this factor should be considered in designing the scenarios.

For all scenarios, the speed of travelling nodes is varied from 0m/s to 30m/s in steps of 5m/s. And the speed of MNs within their group ranges from 0m/s to 5m/s. This can accommodate all speed ranges from pedestrian to car movements.

**Scenario One:** The parameters shown in the following table are used for the first scenario. The number of compartments in the simulation area is incremented to allow movement of travelling throughout the simulation area.

| Parameter                    | Value         |
|------------------------------|---------------|
| Number of Nodes              | 30            |
| Simulation Area              | 1000m X 1000m |
| Simulation Time              | 300Sec        |
| Rewiring Probability         | w=0.01        |
| Travelers' Speed             | 0-30m/s       |
| Reconfiguration Interval     | 100Sec        |
| Number of Compartments       | 16            |
| Number of Travelling Nodes   | 21            |
| Speed of nodes within groups | 0-5m/s        |
| Initial Number of Groups     | 16            |
| Traffic Sources(travelling   | 5             |

*Table 6.1: Simulation parameter settings for scenario one.*

**Scenario Two:** Table 6.2 shows the parameter settings correspond to scenario 2. In this scenario, the case of high social interaction among people is represented by the MNs in the simulation area. This is achieved by using the maximum value of the rewiring probability, parameter of the community based mobility model.

| Parameter                    | Value         |
|------------------------------|---------------|
| Number of Nodes              | 30            |
| Simulation Area              | 1000m X 1000m |
| Simulation Time              | 300Sec        |
| Rewiring Probability         | w=0.2         |
| Travelers' Speed             | 0-30m/s       |
| Reconfiguration Interval     | 100Sec        |
| Number of Compartments       | 16            |
| Number of Travelling Nodes   | 21            |
| Speed of nodes within groups | 0-5m/s        |
| Initial Number of Groups     | 16            |
| Traffic Sources(travelling   | 5             |

*Table 6.2: Simulation parameter settings for scenario one.*

- 1. Scenario Three:** To analyze the performance of GWDAs with respect to social movement patterns, the parameters settings should be able to maximize the effect of mobility over other performance affecting factors, like interference. Because of this, in this scenario the dimension of the square simulation area is increased to 1200m keeping all other parameters similar to Scenario One. The density of nodes is lower than scenario one.
- 2. Scenario Four:** By applying 1200m dimension of the square simulation area, the density of nodes is made to be lower than the first two scenarios, and the social interaction is made high like scenario two using the maximum value for the rewiring probability.(i.e.  $w=0.2$ )

The simulation model applied together with, metrics of measurement and other different simulation environment related discussions have been presented in this chapter. In the next chapter the simulation based performance analysis with respect to the effect of mobility on the three GWDAs is presented.

## Chapter Seven: Simulation Results and Discussions

In this chapter, the simulation results on the effect of social mobility on the performance of AODV based internet GWDAs for all the scenarios (Section 6.1.3) and metrics (Section 6.3) are presented. Further, based on the results obtained, the comparative evaluation of the discovery algorithms is also discussed.

### 7.1 Results Obtained Using Scenario One

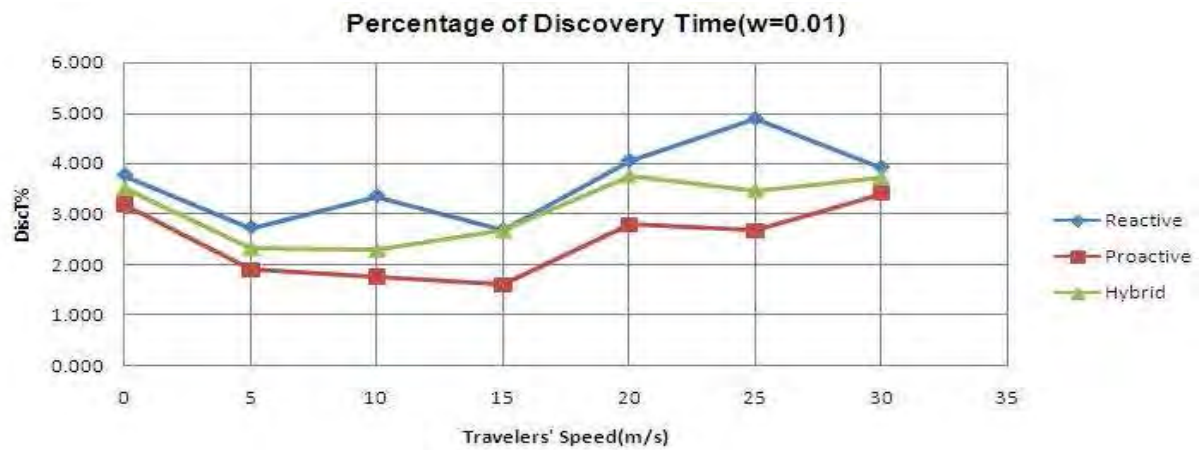
The first scenario represents a low social interaction mobility condition, where 30 nodes have been used in 1000m by 1000m simulation area. The speed of the travelers' node has been varied from 0 to 30m/s at steps of 5m/s. The metrics are calculated and the results are shown below from Fig 7.1 to Fig 7.10.



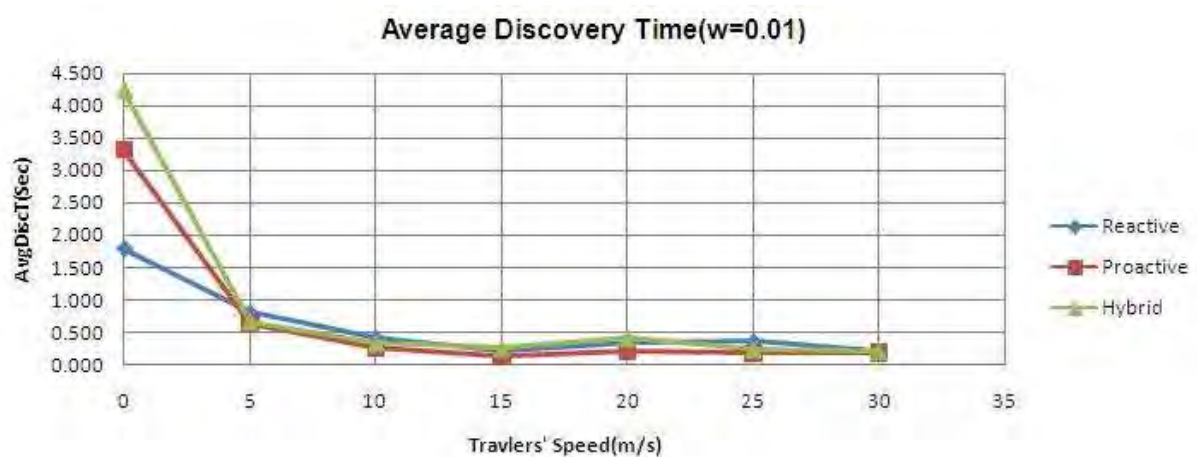
*Figure 7.1: Percentage of handover time vs. traveling nodes' speed*



*Figure 7.2: Average handover time vs. traveling nodes' speed*



*Figure 7.3: Discovery time in percentage vs. traveling nodes' speed*



*Figure 7.4: Average discovery time vs. traveling nodes' speed*

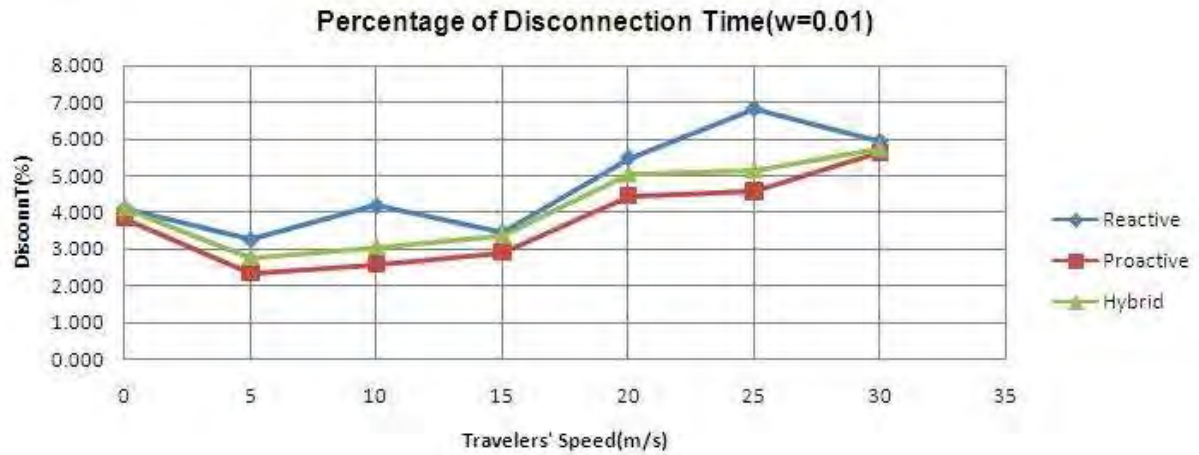


Figure 7.5: Percentage of disconnection time vs. traveling nodes' speed



Figure 7.6: Expanding radial-ring search time vs. traveling nodes' speed



Figure 7.7: Packet delivery fraction vs. traveling nodes' speed



Figure 7.8: Average end-to-end delay vs. traveling nodes' speed



Figure 7.9: Routing Overhead vs. traveling nodes' speed



Figure 7.10: Normalized Routing Load vs. traveling nodes' speed

*Percentage of handover time* in all the internet GWDAs has increased (figure 7.1) from around 1% up to 5% of the simulation time (300Sec) with increase in travelers' speed from 0m/s up to 30m/s. But the *average handover time* (figure 7.2) for all algorithms has increased up to 5m/s travelers' speed and decreased after 5m/s travelers' speed. The increase in the percentage of handover and decreasing trend of average handover time shows that, there is frequent handover with increase in travelers' speed. This is because, at higher speeds, MNs can easily and quickly move between the gateways nodes.

Comparatively, in all speed ranges, the proactive GWDA performed better in terms of the total time the algorithm spent for handover (percentage of handover) and the average handover time. This is because, in the proactive GWDA, the route to a GW is updated every five seconds by the GWADV packets that are broadcasted throughout the MANET. The reactive GW discovery has less performance than the other two algorithms. This is the result of the initial delay introduced from its reactive nature, whenever a handover is performed. The hybrid GWDA's performance is nearly the average of the reactive and the proactive discovery algorithms. Because, initial latency is introduced when traffic sources are outside 3 hops distance from the GW like the reactive GWDA. And, it takes advantage of GWADV packets, when nodes are near to the GW nodes (less than or equal to 3 hops distance).

The *percentage of discovery time* (i.e. total GW discovery time in percentage out of the total simulation time) as shown in figure 7.3 decreased up to 15m/s speed. This is because, the low social interaction together with the lower speeds' of travelling nodes, makes it hard for the control messages to find intermediate nodes which can lead towards to the GW nodes. This is also shown by the decreasing *Average discovery time* in figure 7.4 up to 15m/s travelers' speed. But as the travelers' speed is increased beyond 15m/s, the average value stays almost constant (around 300msec) for all the algorithms, as the higher mobility results in distribution of nodes in the area. However, the percentage of discovery

time is increased by 1.5% to 2%. This is the result of frequent disconnection of MNs as they can easily move outside the radio range of each other at a high relative speeds.

The proactive GWDA is performed better than the other two discovery algorithms. On average, it spends 30% lesser time for discovery than the reactive GWDA. The proactive GWDA gets a quick update and also a quick reply from the periodic GWADV packets. But the reactive GWDA has to multicast RREQ\_I and waits for a RREP\_I message, every time the GW route fails. The average discovery time of the proactive GWDA up to 5m/s speed is higher than the reactive, as MNs are not yet distributed in the simulation area to facilitate easy transfer of GAWAD packets which leads to delay from the increased sizes of MN's queues. Including the initial latency problem of its reactive characteristic, the hybrid GWDA has the highest average discovery time when the travelers' speed is below 5m/s.

The percentage of time that each algorithm spends being completely disconnected (i.e. from time of error occurrence until a valid route is setup to the GW node) from the GW nodes out of the total simulation time (300Sec) verses travelers' speed is shown on figure 7.5. All the discovery algorithms showed high disconnection time especially after 15m/s speed and the proactive GWDA spent around 20% (on average) lesser time being disconnected from the GW than the least performing algorithm among the three( i.e. the reactive GWDA). The general trend of all the algorithms on the percentage of discovery time (figure 7.3) and percentage of disconnection time are similar; however, the 30% difference between the proactive and the reactive GWDA has become around 20%. This shows that, the error propagation time in case of the proactive GWDA is 10% higher than that of the reactive. This is resulted from the routing overhead in the network because of the periodic GW advertisements. Except this, the same justification stated for percentage of GW discovery time holds true here as well.

The *expanding radial-ring search time* (figure 7.6) decreased as mobility speed increased for all the three discovery algorithms. For this process at the beginning

of the simulation time, MNs at higher speed quickly distribute in the area and facilitate radial search and connection to the GW nodes in a shorter time. The reactive and the hybrid GWDAs spend nearly equivalent time but 31% smaller than the proactive GWDA. This is because; it takes advantage of the periodic GWADV packets broadcasted throughout the network to update its default route.

As the travelers' speed is increased from 0m/s to 5m/s, the PDF (figure 7.7) is also increased. This is because the increase in the speed helps to distribute MNs in the simulation area and facilitates connection to the GW nodes. But after the speed of 10m/s, the PDF decreases for all the three discovery algorithms. This is because, MNs at higher relative speeds can easily move outside each other's radio range and cause a link failure. Packets that are being transmitted on this link will be dropped and contributes to the lesser PDF value at higher speeds. As it can be seen in the figure, the proactive GWDAs' efficiency in terms of delivering packets successfully, in all speed ranges, is better than the reactive and hybrid GWDAs. This has resulted from the quick update and maintenance of its broken link to the internet using GWADV packets.

The average delay (figure 7.8) of the three discovery algorithms has increased after 15m/s travelers' speed, but the proactive discovery algorithm has the lowest average delay for all speed ranges. The increase in average delay after around 15m/s is because of the frequent route failure that in turn has resulted from the high mobility condition. In proactive GW discovery, GW advertisement packets are broadcasted throughout the network every 5Sec. This helps to update routes to the internet proactively and hence packets can be sent after spending relatively shorter time in MN's queue thereby reducing end-end delay and improving PDF.

The routing overhead (number of control packets at the network layer in figure 7.9) and the amount of control packets required in delivering a data packet successfully (NRL in figure 7.10) increased as the mobility speed is increased. This is due to the frequent disconnections and frequent control messages sent

over the network to re-discover the GW nodes and continue communication to the internet. Because of the advertisement packets frequently sent into the MANET, the proactive GWDA shows on average around a 36% higher routing overhead over the reactive GWDA. And the hybrid discovery algorithm is nearly the average of the two as the advertisement is made only for 3 hops distance from the GW nodes.

## 7.2 Results Obtained Using Scenario Two

Scenario two represents a high social interaction mobility condition of scenario one. The simulation results are shown from Fig 7.11 to Fig 7.20.

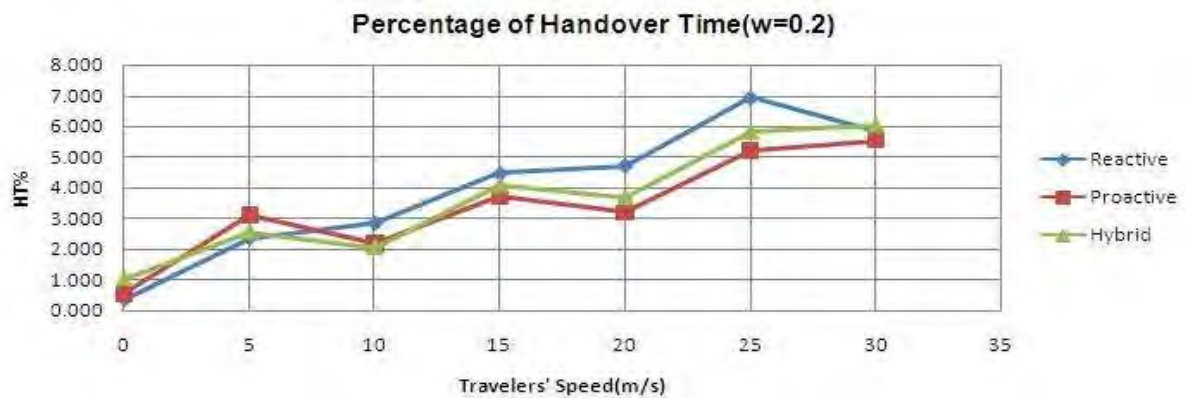
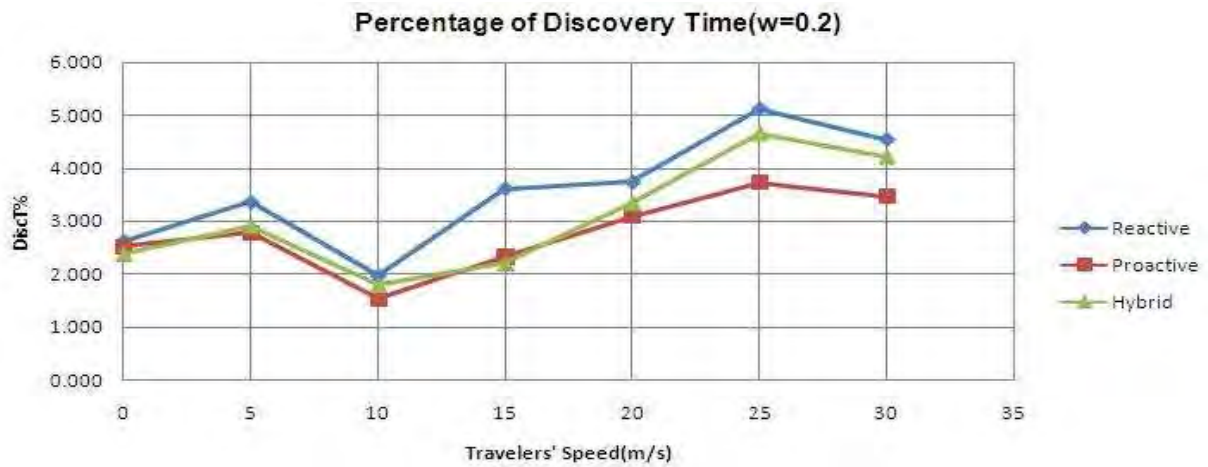


Figure 7.11: Percentage of handover time vs. traveling nodes' speed

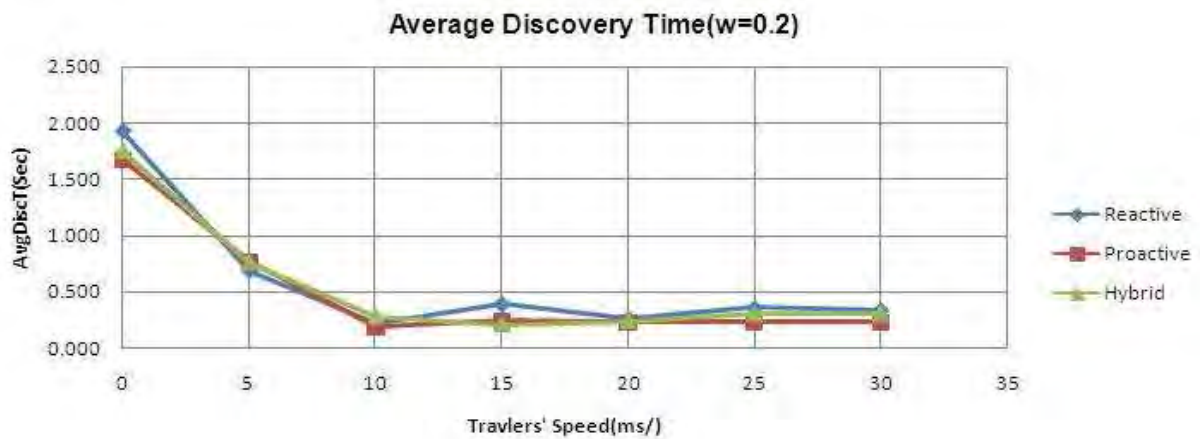


Figure 7.12: Average handover time vs. traveling nodes' speed

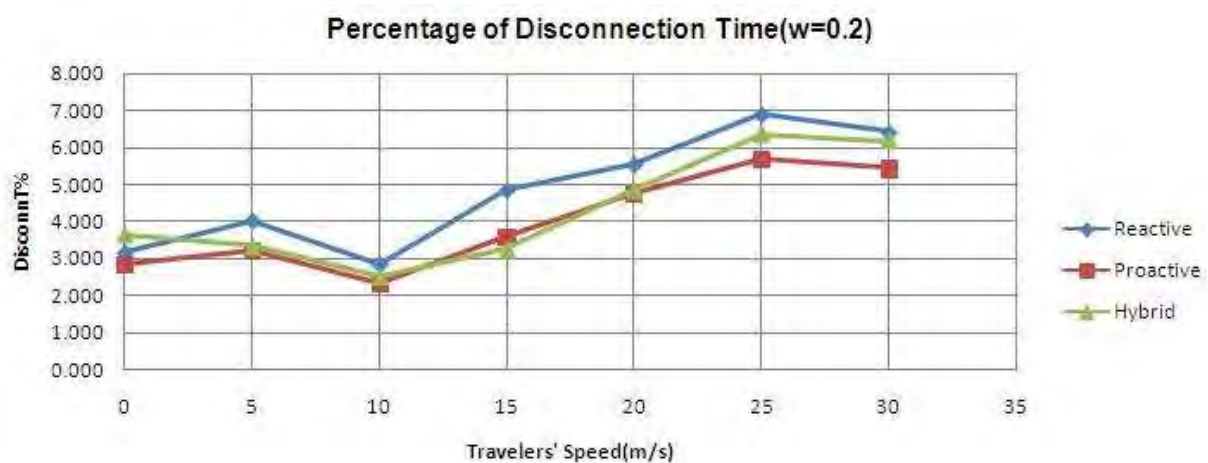
*Performance Evaluation and Nodes' Mobility Effect Analysis on AODV based Internet Gateway  
Discovery Algorithms in Social Networks*



*Figure 7.13: Percentage of discovery time vs. traveling nodes' speed*



*Figure 7.14: Average discovery time vs. traveling nodes' speed*

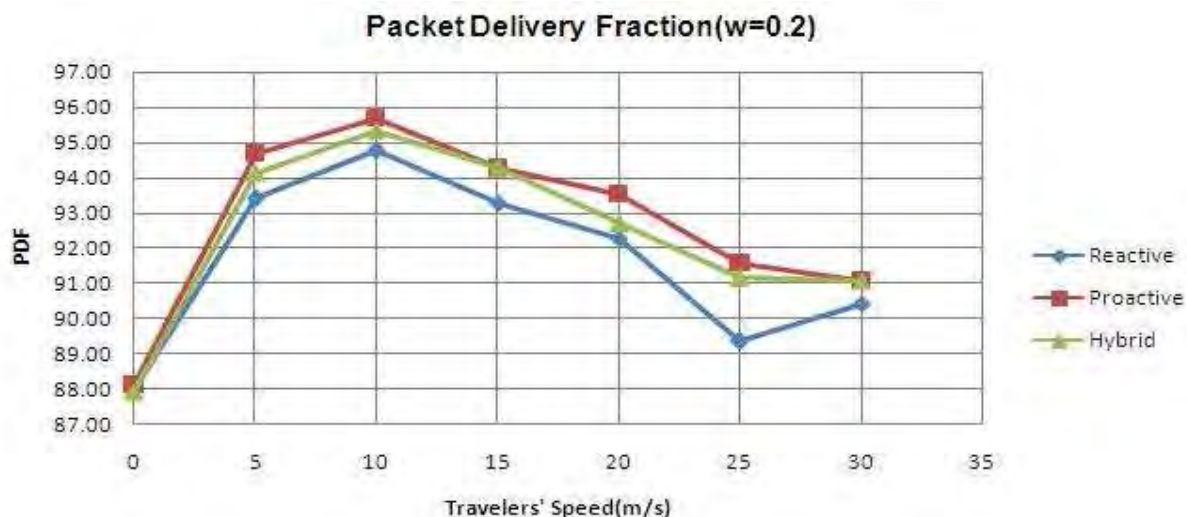


*Figure 7.15: Percentage of disconnection time vs. traveling nodes' speed.*



*Fi*

*Figure 7.16: Expanding radial search time vs. traveling nodes' speed*



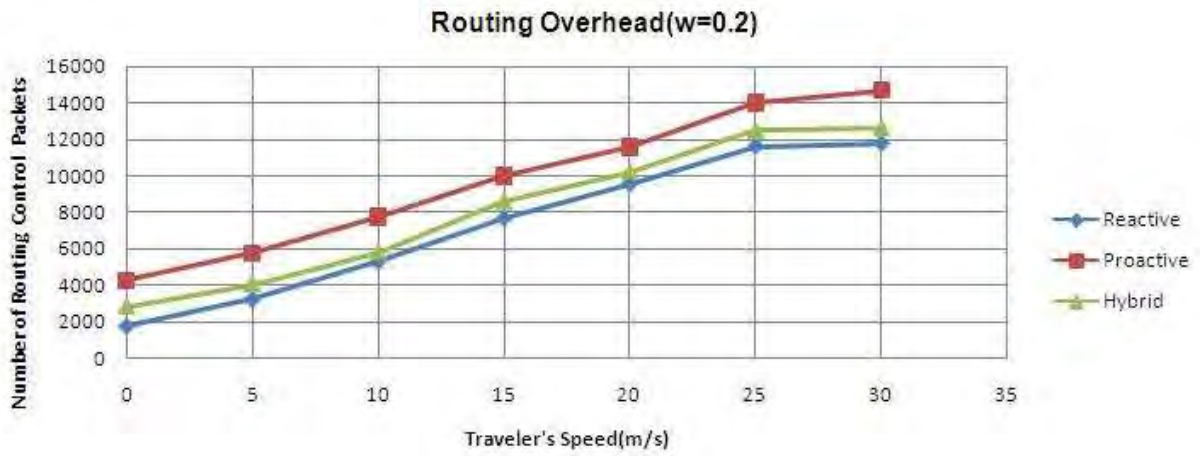
*Fi*

*Figure 7.17: Packet delivery fraction vs. traveling nodes' speed*

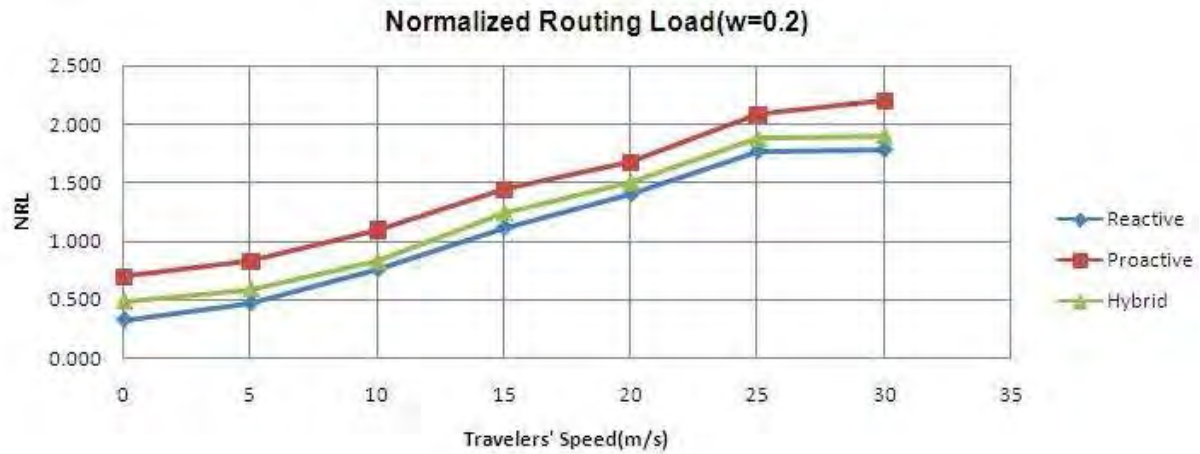


*Fi*

*Figure 7.18: Average end-to-end delay vs. traveling nodes' speed*



*Figure 7.19: Routing overhead vs. traveling nodes' speed*



*Figure 7.20: Normalized routing load vs. traveling nodes' speed*

The percentage of handover time (figure 7.11) and the average handover (figure 7.12) time show nearly a similar trend as in the case of scenario one. However, in the case of scenario one, the percentage of handover is more than 1% for all the three algorithms at 0m/s travelers' speed. But in scenario two it is below 1% for all the internet GWDAs. This is because, when there is high social interaction (like in the case of scenario two) nodes quickly spread throughout the simulation area and facilitates communication to the GW node through intermediate nodes even at lower speeds of traveling nodes. And the best performances are obtained at lower speeds than the case of low social interaction scenarios.

The high social interaction has a negative impact on the performance by aggravating the frequent disconnections resulting from the high mobility of MNs. Consequently, the maximum value for percentage of handover in scenario two is higher than the maximum value for percentage of handover in scenario one. Difference is observed only in the maximum and minimum values of the two metrics of the 2 scenarios (i.e. Percentage of handover and average handover time). The same justification can be applied for the difference observed in the average handover time between the two scenarios. Especially, after 10m/s travelers' speed, the proactive GWDA performed better as in the case of scenario one.

The percentage of discovery time (figure 7.13) is better than scenario one up to the 10m/s travelers' speed. And all algorithms achieved their best value at 10m/s, unlike scenario one, where they should have waited up to 15m/s.

However, after 10m/s speed, all algorithms showed a high rate of increment unlike the case of scenario one, in their percentage of discovery, because of the high social interaction.

The average discovery time (figure 7.14) decreased rapidly up to the speed of 10m/s and remained nearly constant up to 30m/s speed as in the case of scenario one. But, the average discovery time below 10m/s for scenario two is very much lesser for proactive and hybrid GWDAs than the case in scenario one. This is because; the high social interaction at lower speeds has positive impact by distributing MNs in the simulation area and hence facilitating easy connection to the internet.

The percentage of total disconnected time (figure 7.15) shows an increasing trend especially after 10m/s speed for all the three GWDAs. The difference in percentage of discovery time between scenario one and two hold true here too. The comparative evaluation of the algorithms and the comparison of the percentage of discovery time and percentage of disconnection time is similar to the case in scenario one.

In the case of expanding radial-ring search time (figure 7.16), the time requirements of each discovery algorithm at travelers' speed range of 0m/s up to 5m/s is lower than the case of low social mobility condition in scenario one. This is because in this scenario, distribution of nodes in the simulation area at lower speeds is better than scenario one, because of the high social interaction among groups. The general trend and relative comparison among the GWDAs is the same as in the case of scenario one.

The PDF (figure 7.17) increased with nodes' mobility speed up to 10m/s like scenario one. And after the speed of 10m/s, it decreases almost smoothly for all the three discovery algorithms. The proactive GWDA is the best for all speed ranges with MNs' high social interaction. For the average delay (figure 7.18), unlike scenario one, all the three discovery algorithms achieved the minimum value earlier (i.e. at 5m/s speed) and increased for the later speeds. This is because; the high social interaction at lower speeds has positive impact in distributing nodes in the simulation area. The proactive discovery algorithm has the lowest average delay in all speed ranges.

The routing overhead (number of control packets at the network layer figure 7.19) and the amount of control packets required in delivering a data packet successfully (NRL in figure 7.20) shows similar trend as in the case of scenario one. The only difference between the two scenarios lies in the speed range from 0m/s up to 5m/s, where the proactive GWDA requires even a higher number of control packets in scenario one per a successfully delivered packet. Just like the case of scenario one, the proactive GWDA has the highest routing overhead than the other two discovery algorithms.

### **7.3 Results Obtained Using Scenario Three**

The third scenario represents a mobility condition at low social interaction and lower density (i.e. 30 MNs placed in 1200m by 1200m simulation area). The speed of the traveling nodes has been varied from 0 to 30m/s at steps of 5m/s.

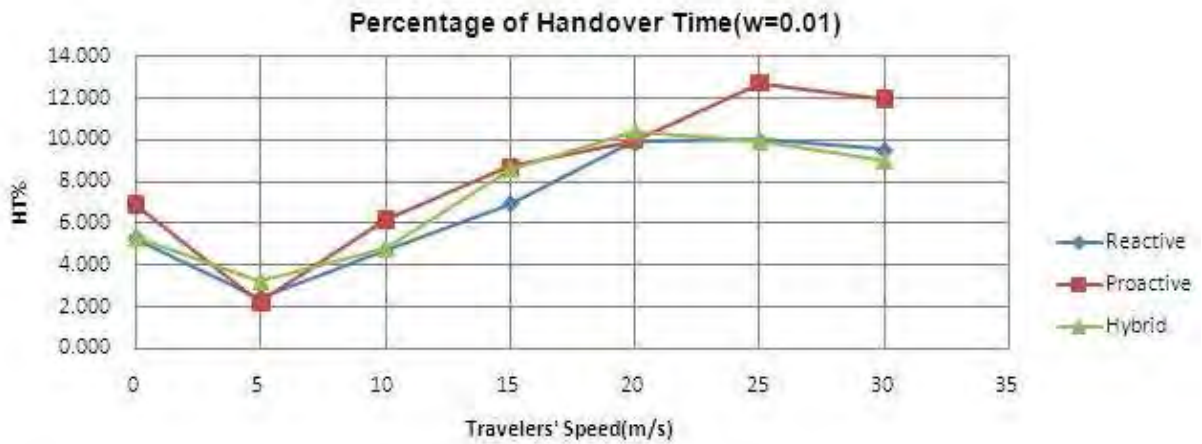


Figure 7.21: Handover time in percentage vs. traveling nodes' speed.



Figure 7.22: Average handover time vs. traveling nodes' speed

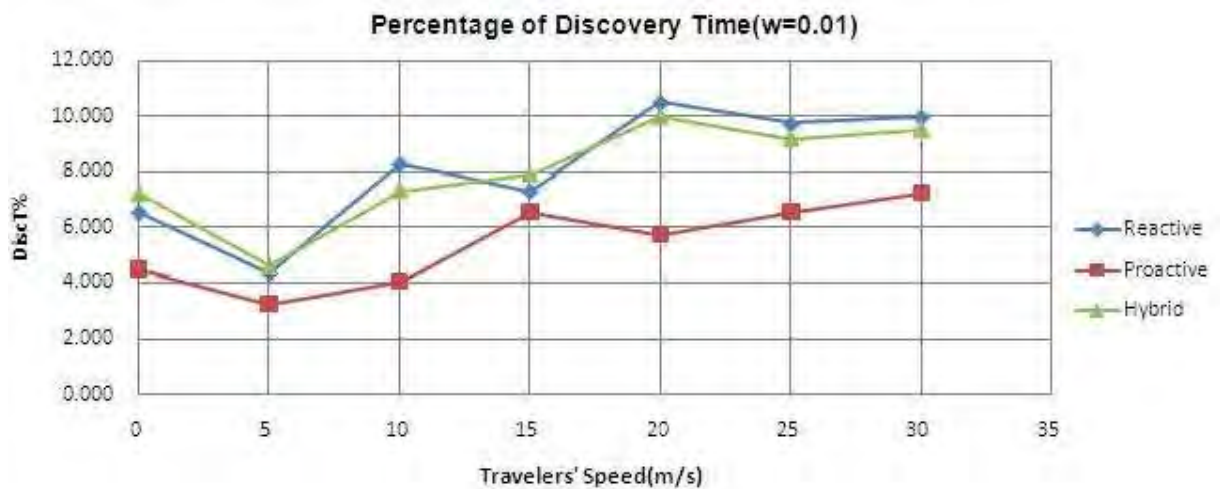


Figure 7.23: Percentage of discovery time vs. traveling nodes' speed

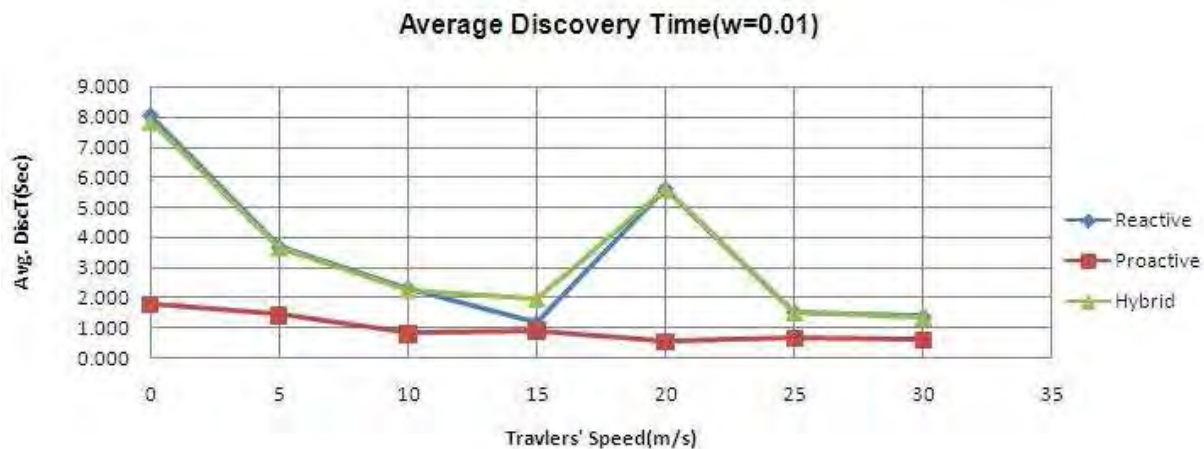


Figure 7.24: Average discovery time vs. traveling nodes' speed

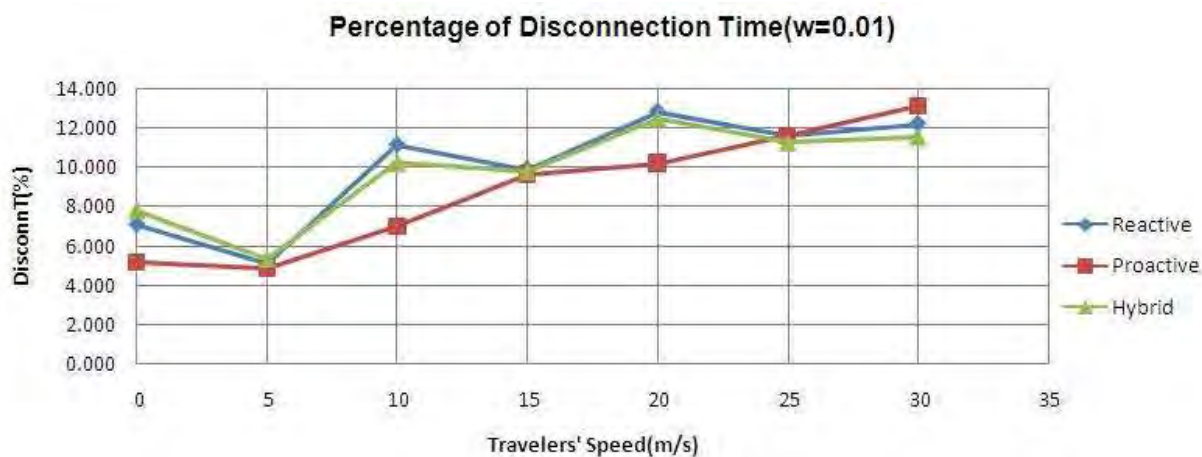


Figure 7.25: Percentage out of disconnection time vs. traveling nodes' speed

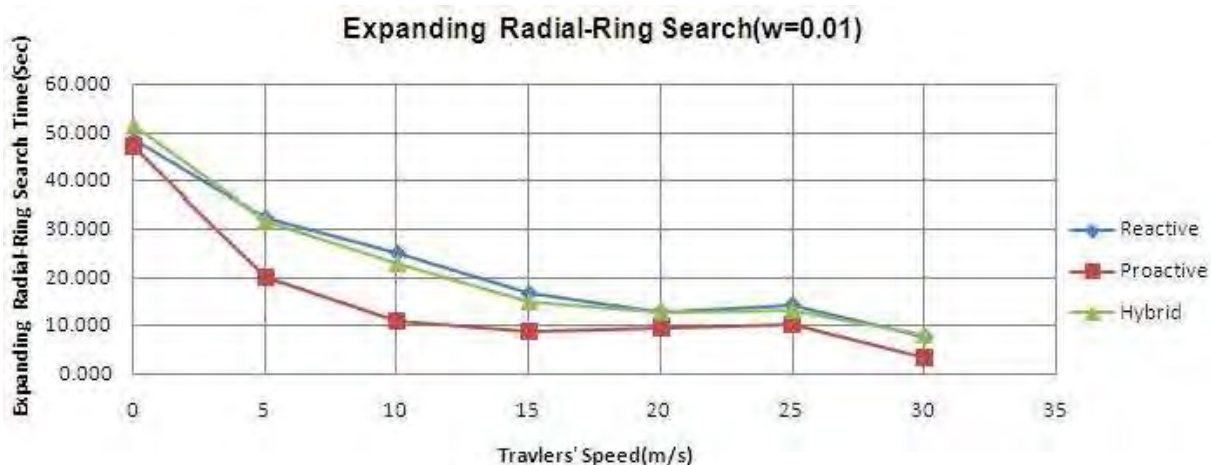
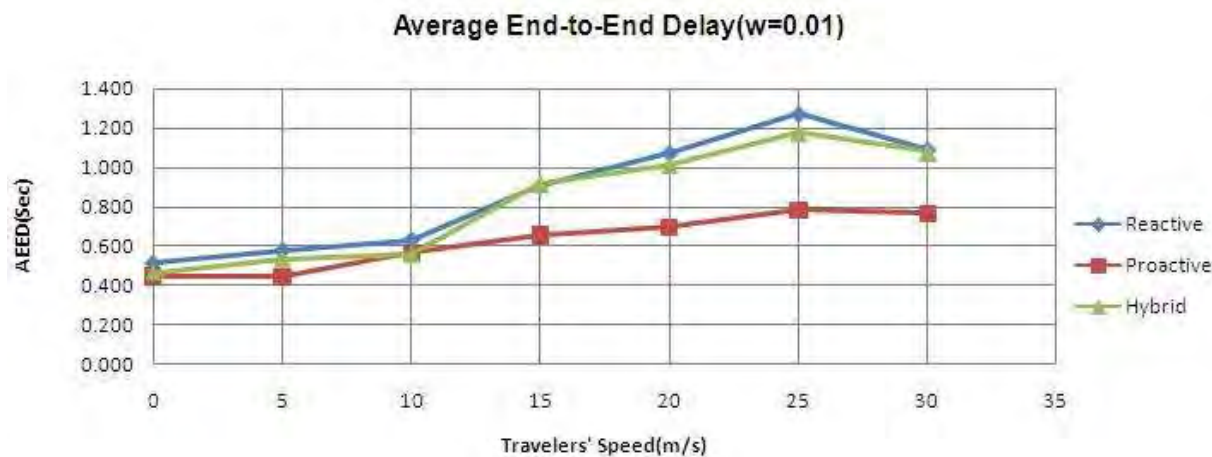


Figure 7.26: Expanding radial-ring search time vs. traveling nodes' speed



*Figure 7.27: Packet delivery fraction vs. traveling nodes' speed*

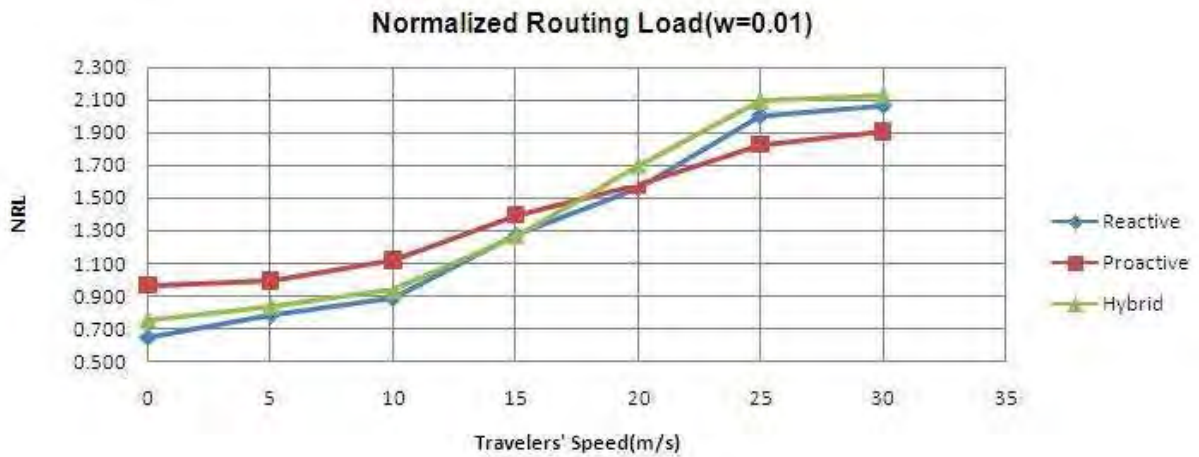


*Figure 7.28: Average end-to-end delay vs. traveling nodes' speed*



*Figure 7.29: Routing overhead vs. traveling nodes' speed*

*Fi*



*Figure 7.30: Normalized routing load vs. traveling nodes' speed*

All three internet GWDAs' percentage of handover time (figure 7.21) increased after 5m/s travelers' speed up to 12% of the simulation time unlike scenario one, where the maximum value is around 5%. This has resulted from the lesser MNs' density in the increased simulation area, where nodes can easily move outside the radio range of each other. And also as it can be seen on figure 7.22, the algorithms spends more time, than the case in scenario one, to handover between the GW nodes. The proactive GWDA spends more time for handover than the other two GWDAs. As can be seen from the average handover time on figure 7.22, considering both the results, one can say that, the percentage of handover time for the proactive GWDA has increased because of frequent handovers. This in turn is due to periodic updates, which brings options to the MNs to handoff to a GW with a shorter distance in terms of number of hops.

Percentage of total GW discovery time (figure 7.23) decreases up to 5m/s speed and shows an increasing trend after that speed for all the three algorithms. Compared to scenario one, the percentage of discovery time has increased up to 6% to 10% for the algorithms. However, in the case of scenario one, it has increased up to 3% to 5% of the total simulation time. The increased travelers' speed together with the decreased density of nodes has made it difficult for control messages to propagate and also made MNs to easily move outside the

radio range of each other. Hence, causing frequent discoveries and increase average discovery times (figure 7.24) than in the case of scenario one.

The proactive GWDA shows the best performance in terms of percentage of discovery time and average discovery time metrics than the other two discovery algorithms. The reactive GWDA is relatively the least in performance, because of the initial latency involved on every re-discovery of the GW node. The hybrid GWDA performed between the two, but much closer to the performance of the reactive GWDA. This shows that, the source nodes spend most of the simulation time beyond 3 hop distance from the GW node. As the effect of mobility is higher at lower density of MNs, the difference in performance among the algorithms has become more visible in the third scenario than the first two scenarios for the above two metrics.

The time the algorithms spend, being disconnected from the GW nodes in percentage out of the total simulation (figure 7.25) time increased for all algorithms above 5m/s travelers' speed. The percentage of disconnection time is almost twice the value of scenario one and two, because of the lesser density of MNs in the simulation area. The proactive GWDA spends lesser time being disconnected from the GW than the other two discovery algorithms.

The expanding radial-ring search time (figure 7.26) decreases as mobility speed increases for all the three discovery algorithms. But compared to the cases of first two scenarios, it has almost doubled because of the lesser density of nodes in the simulation area. At the beginning of the simulation time, the traveling nodes move at slower speed and hence there is even lesser distribution of nodes in the simulation area. This makes it difficult to get connection to the GW nodes. The proactive GWDA spends the lesser time among the three discovery algorithms by taking advantage of advertisement messages.

The PDF increased as mobility speed of nodes and as the result their distribution in the simulation area increases. But after the speed of 5m/s, the PDF decreased for all the three discovery algorithms. Comparing scenario one and three, the best PDF value has dropped from 96% which occurred at 15m/s in scenario one to 81.6% at 5m/s speed in this scenario (i.e. the third). This has resulted from the lower density of MNs. Figure 7.27 shows that, the proactive GWDAs' efficiency in terms of delivering packets successfully, in all speed ranges is better than the reactive and hybrid GWDAs.

The average delay (figure 7.28) of the three discovery algorithms increases as travelers' increases from 0m/s to 30m/s. The average delay has increased to a value, which is around three to four times the case in scenario one and two. Distribution of the advertisement packets throughout the MANET is not easy at lower mobility speeds, because of the less distribution of nodes in the area. In relation to this, the proactive GWDA performs like the other two GWAs, up to 10m/s travelers' speed. But after that, it starts to take advantage of the advertisement packets from the GW to minimize the delay of packets on MNs' queues.

Routing overhead (figure 7.29) of all the three GWDAs increased as mobility speed increased. The same is true for the normalized routing load (NRL in figure 7.30).

The amount of control packets at the network layer is lesser than the case of scenario one and two. The propagation of control packets between nodes is also a challenging task in this environment. But the proactive GW discovery has still the highest number of control packets at the network layer among the three discovery algorithms. This is because of the advertisement messages periodically broadcasted by the GW nodes. Unlike the case of scenario one and two, the amount of control packets required to deliver a data packet for proactive GWDA is lesser after about 17m/s travelers' speed. This is because, the algorithms capability to deliver more packets as shown in its PDF value. The hybrid GWDA

stays being the average of the two up to the speed of 17m/s and after that, showed least performance. It takes the disadvantages of the two algorithms, that is, a high initial latency from the reactive and high overhead from the proactive GWDAs.

#### 7.4 Results Obtained Using Scenario Four

Scenario four represents a mobility condition similar to scenario three except that with a high social interaction.

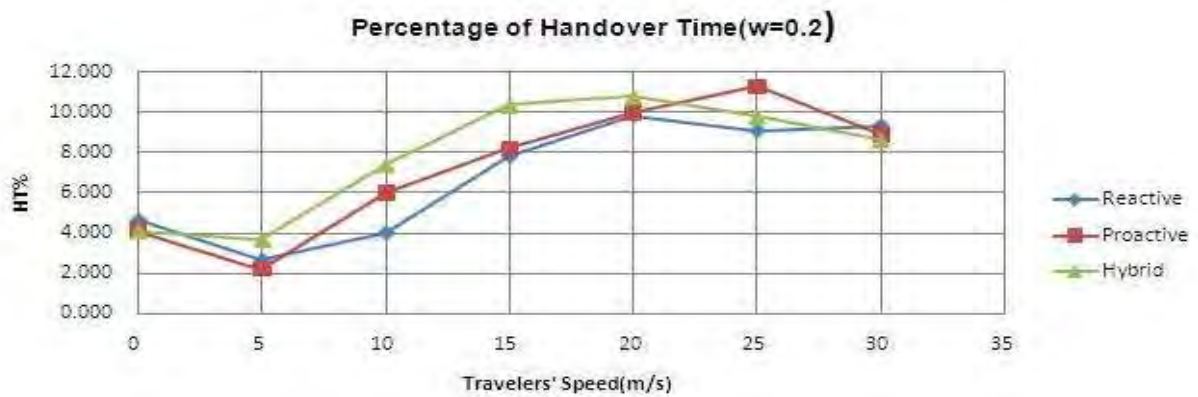


Figure 7.31: Percentage of handover time vs. traveling nodes' speed.



Figure 7.32: Average handover time vs. traveling nodes' speed.

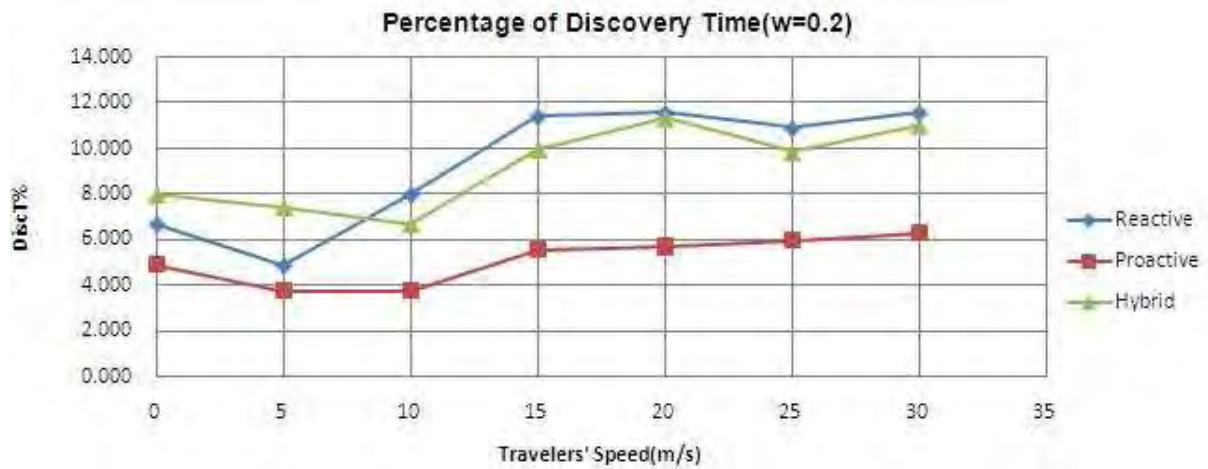


Figure 7.33: Discovery time in percentage vs. traveling nodes' speed.

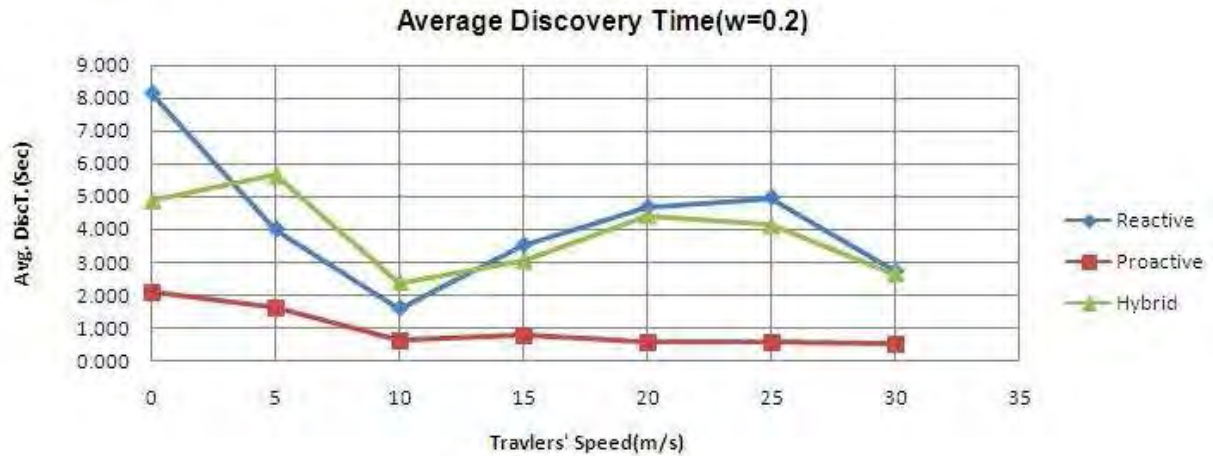


Figure 7.34: Average discovery time vs. traveling nodes' speed

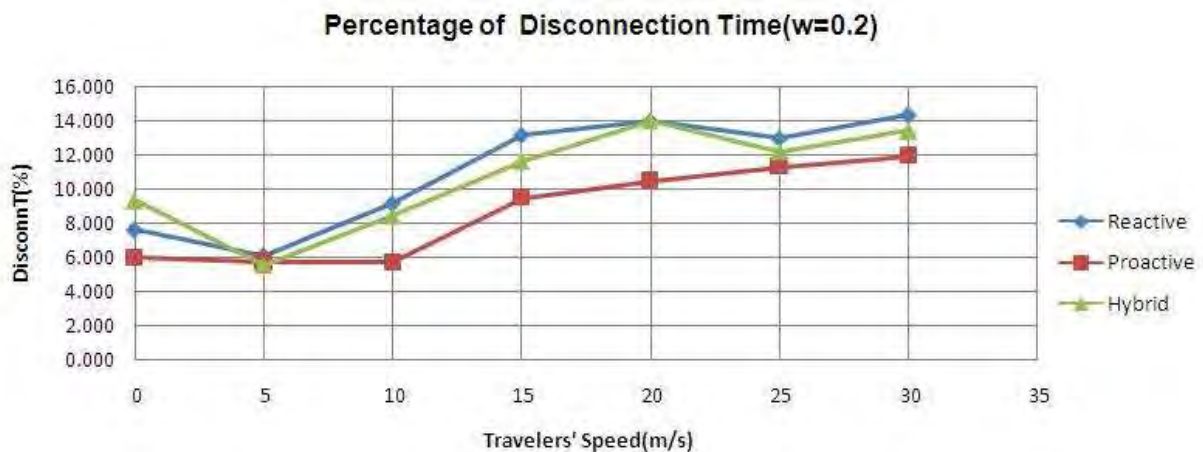


Figure 7.35: Total Disconnection time in percentage vs. traveling nodes' speed

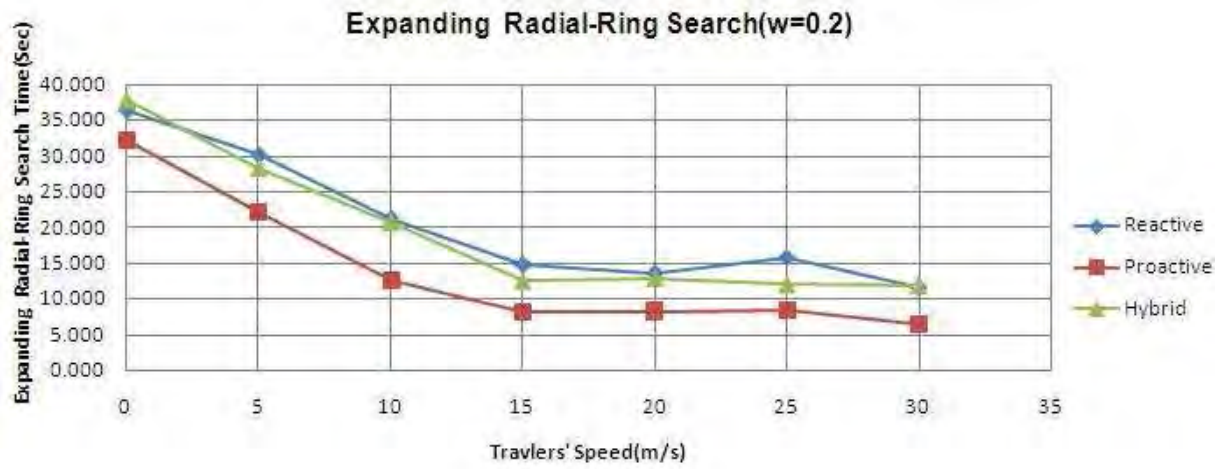


Figure 7.36: Expanding radial-ring search time vs. traveling nodes' speed

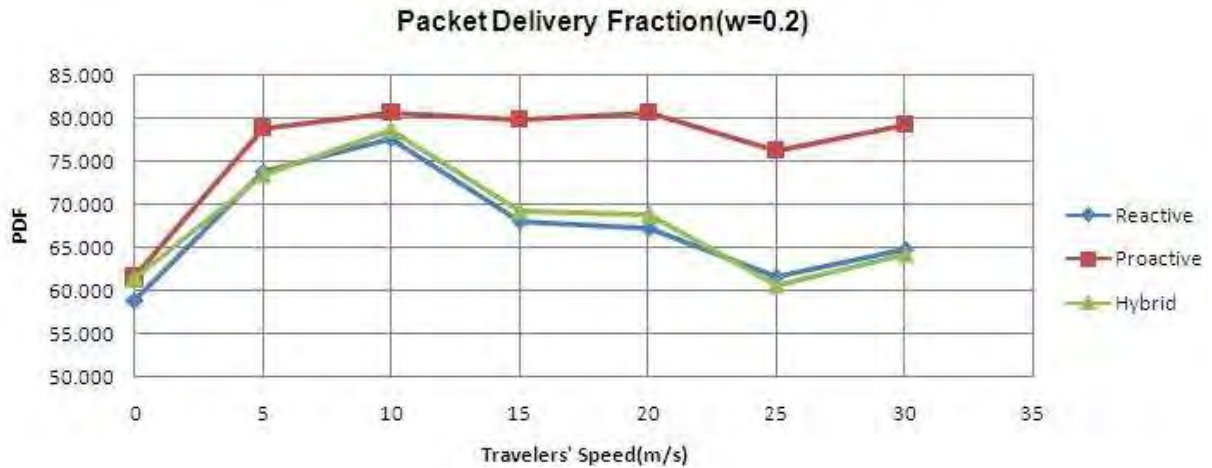


Figure 7.37: Packet delivery fraction vs. traveling nodes' speed.

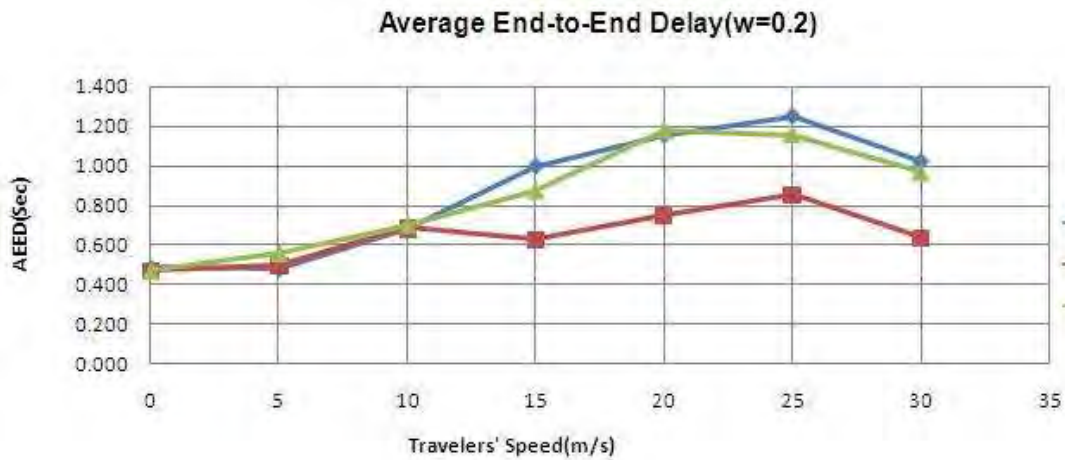


Figure 7.38: Average end-to-end delay vs. traveling nodes' speed.



. Figure 7.39: Routing overhead vs. traveling nodes' speed

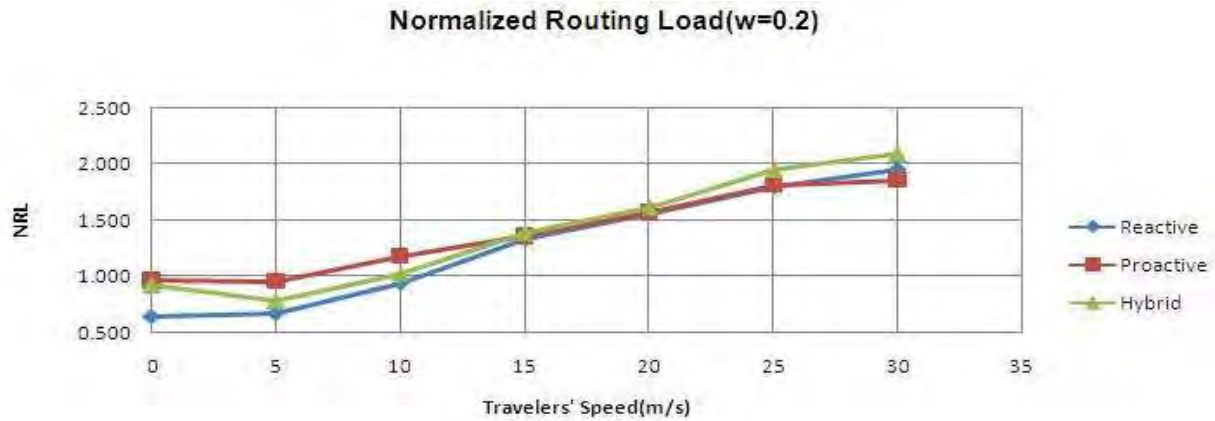


Figure 7.40: Normalized routing load vs. traveling nodes' speed.

All three internet GWDAs' percentage of handover time (figure 7.31) shows increasing trend after 5m/s travelers' speed. This is because of the frequent disconnection at higher speeds. The hybrid GWDA spends more time for handover than the other two GWDAs. As can be seen from the average handover time on figure 7.32, the hybrid GWDA has higher average handover time than the proactive discovery algorithm. The high social interaction facilitates the arrival of the advertisement messages to the MNs connecting to the internet, with better options of distance to a GW in terms of number of hops. The high average handover time together with the frequent handover makes the algorithm to have the least performance among the three GWDAs in respect to the total time the algorithm spends for handover during the simulation time.

Percentage of discovery time (figure 7.33) decreases up to 5m/s speed and increases after that speed for all the three algorithms. Compared to the low social mobility condition in scenario three, the reactive and the hybrid GWDAs are highly affected. This is the result the frequent disconnection and re-discovery of the GW node and the initial latency of the algorithms on every discovery. The higher speeds of traveling nodes combined the high social interaction causes a high mobility condition. And hence, leads to frequent link failure. The proactive GWDA has nearly equivalent performance with the case in scenario three; this is the result of a compromise between easy propagation of periodic advertisement messages and frequent disconnections.

The average discovery time (figure 7.34) for the proactive GWDA out performs the other two GWDAs using its periodic advertisement messages. The reactive and hybrid discovery algorithms achieves their minimum value earlier (10m/s speed) unlike the case in scenario three and are highly affected by the high social mobility condition afterwards. This is the result of the initial latency from the nature of the reactive GWDA. The hybrid GWDA performance is very much close to that of the reactive one, as the source nodes stay mostly outside 3 hops distance from the GW node. This in turn is the result of the increase in dimension of the simulation area. The proactive GWDA shows the best performance in terms of percentage of discovery time and average handover time metrics than the other two discovery algorithms.

The time each algorithms spends, being disconnected from the GW nodes in percentage out of the total simulation (figure 7.35) shows nearly a similar trend for all the three GWDAs as in the case of scenario three. In this scenario also, it is the proactive GWDA that spends lesser time being disconnected from the GW than the other two discovery algorithms.

The expanding radial-ring search time (figure 7.36) at the beginning of the simulation is decreased from 50% up to around 35% for all algorithms, when compared to the case in scenario three. This is the result a better distribution of

nodes as a result of the high social interaction at lower speeds. But otherwise the general trend and the comparative evaluation of the algorithms are nearly similar to the observation made in scenario three.

The packet delivery fraction (figure 7.37) at higher social interaction shows almost the same trend as the case in scenario three. The same is true for the average delay (figure 7.38).

Routing overhead (figure 7.39) shows similar trend like the case of scenario three. However, the amount of control packets required to deliver a data packet successfully (NRL in figure 7.40) for the proactive GWDA is lesser than the other discovery algorithms starting from 15m/s travelers' speed.

## **7.5 Summary of Results**

The results obtained by the simulation study show that, the proactive GWDA's performance is less affected by nodes' mobility compared to the other two GWDA's. But it has the highest routing overhead. Generally, at lower speeds (i.e. less than 10m/s) the general trends for all the algorithms shows improvement for each metric. This is because; the increase in speed and also high social interaction has a positive impact for distributing the MNs in the simulation area at lower travelers' speed. However, as travelers' speed is increased to 30m/s, MNs can easily move outside the radio range of each other and hence the performance of all the algorithms is degraded. For the same reason, the high social interaction is observed to aggravate the performance degradation of all the algorithms at higher speeds. The change in the simulation area resulted in a less dense environment and all algorithms have shown reduced performance. This is because, in this environment, nodes can easily move outside the radio range of each other, which resulted in the difficulty of transferring data and control messages to the GW nodes.

The proactive GW discovery is observed to be more resistive to the effects of mobility than the other two GWDA's. But it has the highest routing overhead among the three. For this algorithm, the periodic advertisements from the GW nodes throughout the network were the main advantageous in updating and maintaining routes to the GW nodes proactively.

The next chapter presents discussion on the design, implementation and testing of a solution to enhance the reactive GWDA so that, it can have a better performance with lower routing overhead than the proactive GWDA.

## **Chapter Eight: The Proposed Solution**

### **8.1 Design of the Solution**

The solution proposed in this section introduces additional packets in to the network, and for this reason it is not appropriate for the proactive GWDA, as it has the highest overhead among the three. However, the reactive GWDA has the least routing overhead and so, it is appropriate to be modified for better performance with respect to the impact of mobility.

As it is shown in the previous chapter, the main cause for the performance degradation of the GWDAs is the disconnection of the route to the internet. This failure of GW route occurs as a MN which is part of the active route, goes out of the reach of the neighboring node on the same route. When the GW route fails, packets which are being exchanged with the internet through the GW are dropped, contributing to lower PDF values. After receiving error message, a new GW discovery is initiated. The total time spent, for the frequent error message propagation (Error occurrence time until MN sends RREQ\_I after receiving the ERROR message) and rediscovery process (from RREQ\_I message until RREP\_I or GWADV message is obtained) increases. This also increases the overall end to end delay of the message exchange. The movement of intermediate nodes and the source node itself may also cause frequent handoff to occur between GW nodes. This situation is aggravated as the speed of travelling nodes is increased and as the density of nodes in the area decreases. The high social interaction also aggravates the situation at higher speeds. To alleviate the above problems, it is necessary to replace the suspected GW route before it breaks. This requires, performing GW discovery process in advance (before the GW route breaks). The following section presents the predictive GW discovery.

#### **8.1.1 Predictive Gateway Discovery**

GW discovery can be performed before the GW route to the internet fails. And hence it can be replaced before disconnection to the internet occurs. This can be made using prediction techniques based on the signal power [25], [26], [27],

[28].The signal power is assumed to drop in proportion to  $1/r^4$  with distance, more specifically, the received power is given by

$$P_{received} = K \frac{P_o}{r^4} \quad (8.1)$$

Where  $P_o$  is a constant for each transmitter/receiver pair, based on 'K' (i.e. antenna gain and height). This is the base of the algorithm A, for predicting if two nodes on the GW to the internet are moving outside the radio range of each other or not. To start the predictive GW discovery, the time of route failure needs to be computed using the power level of the received unicast packets from a neighboring node. The steps to perform predictive GW discovery is shown in section (8.1.1.1). The node starts to store the power level or RX level (p) of each received packet from a neighboring node on the GW route, if it is below a warning threshold power level set (power level calculated at a distance of 220m away from the source node). After storing the three power values, the link failure time is predicted if  $p_3 < p_2 < p_1$ , as shown in step 7 of the algorithm. The calculation of the time follows the same formulas as in [26]. The formulas are shown in equations 8.2 to 8.6. However, here whether there is enough time to start GW discovery or not, is also predicted rather than using just 1Sec default value. This can be carried out by storing the difference of packet send and receive times and by computing the exponential moving average of the stored three consecutive times values at the moment, when the criteria  $p_3 < p_2 < p_1$  is met. The prediction of time required to make GW discovery helps to reduce congestion in the network as it can help to avoid unnecessary predictive GW discovery process.

If the GW route failure time is less than or equal to the predicted GW discovery time, a warning message is sent to the source node and the sender node stores warning information on the entry for that link. If the source node has data to be sent to the destination in the fixed network (internet connection) on the GW route with the warning, a GW discovery will be started.

### 8.1.1.1 Algorithm A: Predictive Gateway Discovery

START

1. Read the signal power of the received data packet.
2. If the signal power is less than or equal to the power level at a distance of 220m (i.e.  $P_{\text{threshold}} = 6.09 \times 10^{-10}w$ ), assuming that minimum received signal power at 250m distance is  $(PS) = 3.652 \times 10^{-10}w$ . This is based on NS2's default value.

Then go to step 3.

Else go to step 11.

3. Store the data packet's power.
4. Store the time of arrival for the data packet stored.
5. Store the time difference between send and receive time of the packet (Send time is stored in the packet itself)
6. If number of received packet is equal to three, then go to step 7.  
Else go to step 11.
7. If  $p_3 < p_2 < p_1$

Then predict the time for GW route failure ( $t_p$ ).

Where,  $t_p$  is calculated as follows.

$$t_p = \frac{\sqrt{b^2 - 4ac} - b}{2a} \quad (8.2)$$

$$a = t_2 \sqrt{p_2 p_s} \beta \quad (8.3)$$

$$b = \sqrt{p_s} ((\sqrt{p_1} - \sqrt{p_2}) - t_2^2 \beta \sqrt{p_2}) \quad (8.4)$$

$$c = t_2 \sqrt{p_2 p_s} - t_2 \sqrt{p_1 p_2} \quad (8.5)$$

$$\beta = \frac{(\sqrt{p_1 p_2 t_2} + \sqrt{p_2 p_3 t_3} - \sqrt{p_1 p_3 t_3} - \sqrt{p_2 p_3 t_3})}{(t_2 t_3^2 - t_3 t_2^2) \sqrt{p_2 p_3}} \quad (8.6)$$

8. Predict the GW discovery ( $P_{dt}$ ) time using Exponential moving average (EMA) on the stored three consecutive send to reply time ( $\Delta t_1, \Delta t_2, \Delta t_3$ ) of the data packets.

$$P_{dt} = \alpha (\Delta t_m + (1 - \alpha)^2 \Delta t_{m-1} + (1 - \alpha)^3 \Delta t_{m-2}) \quad (8.7)$$

$$\alpha = \frac{2}{N+1}, N=3 \quad (8.8)$$

9. If  $t_p \leq P_{dt}$

Then send warning message to the previous node.

Else go to step 2, assuming there will not be enough time to make GW discovery and replace the GW route before it breaks.

10. At the source node

If there are more data to be exchanged to the internet

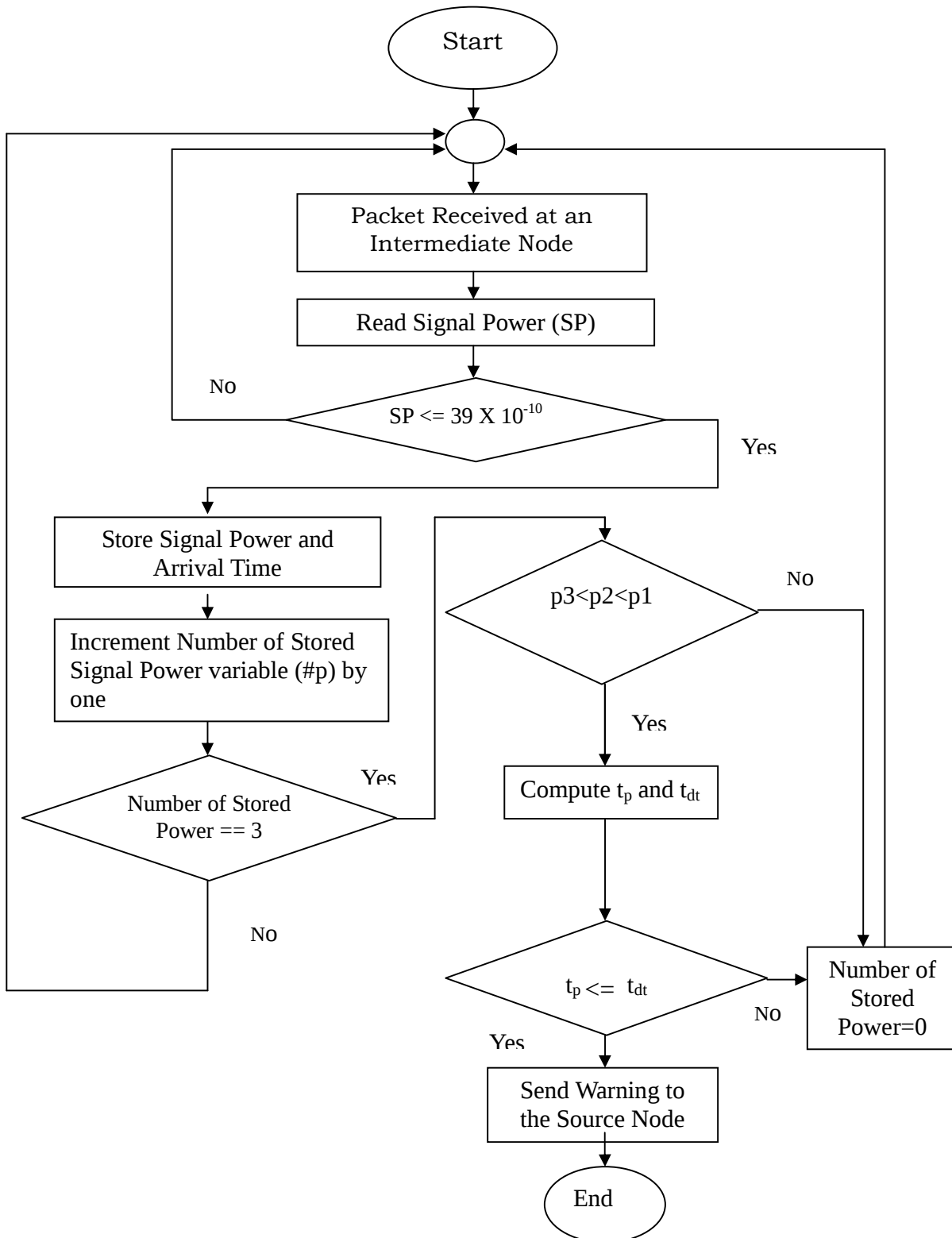
Then initiate a new GW discovery.

Else drop the warning packet and do nothing.

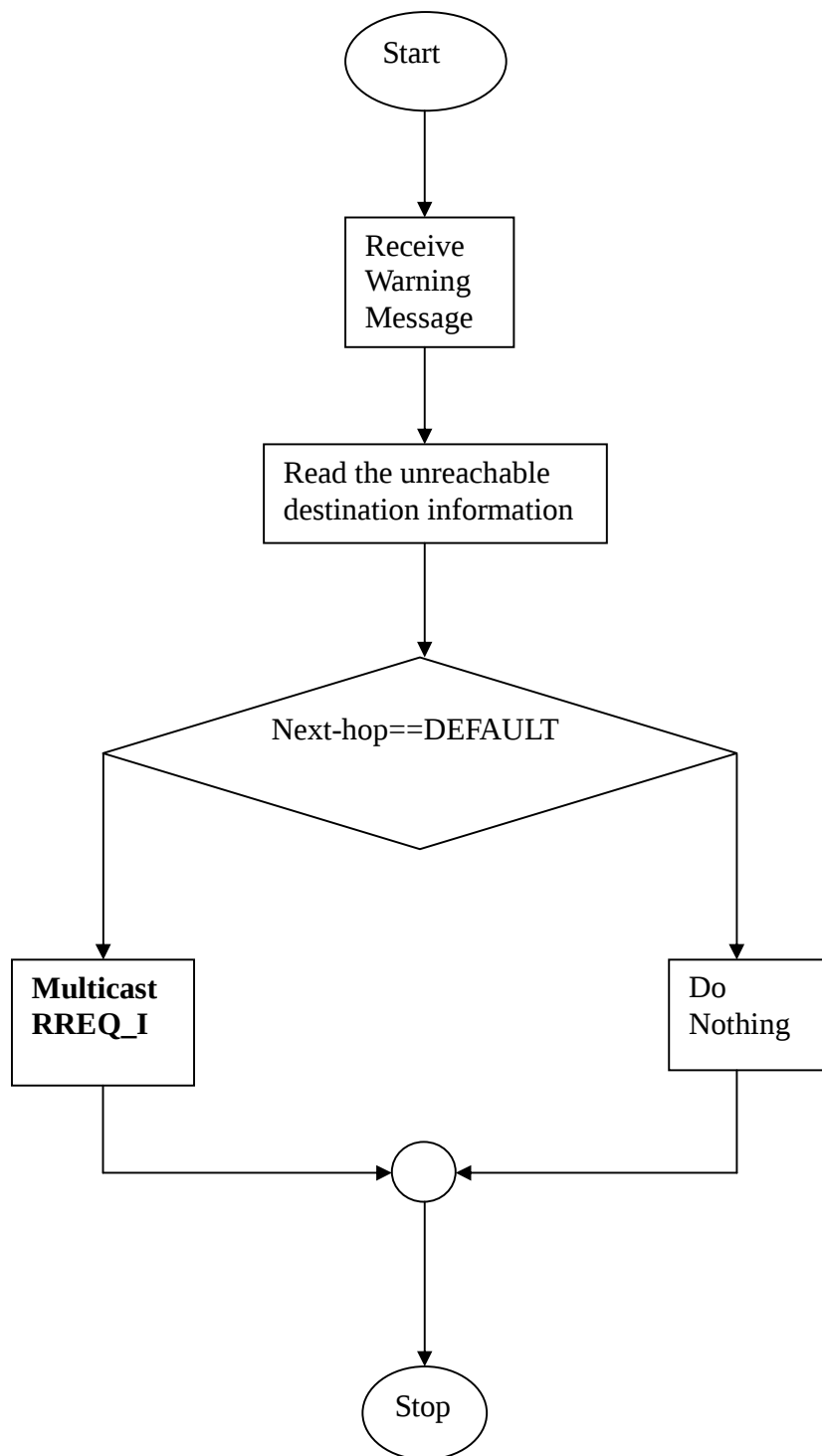
11. If a new packet is arrived

Then go to step 1.

END



*Figure 8.1: Prediction algorithm operation at intermediate node*



*Figure 8.2: Prediction algorithm operation at the source node*

## **8.2 Implementation of the solution**

The predictive GWDA is incorporated in reactive GW discovery protocol. The discovery process by multicasting RREQ\_I message is started before receiving error message from a broken link on the GW route. Instead, prediction whether the link breaks or not will be made according to the algorithm in section 8.1.1.1, before the link breaks and hence reduce the end to end message delay. The number of packets dropped because of link failure also reduces. The modified functions of the enhanced AODV routing protocol for reactive GW discovery is presented below.

1. In AODV.cpp

*rt\_resolve (Packet \*p)*

The transmission of data packets is handled in this class. Reading the rx signal power from the data packet, storing and prediction of link failure time are performed here. The prediction of discovery time using exponential moving average is also performed in this modified class. The sendwarning function is also called from here.

*recv\_AODV (Packet \* p)*

The 'case' for switch statement, *case AODVTYPE\_WARNING* is included to call *recvwarning (p)* function.

*sendwarningprev (aodv rt entry \*rt0, nsaddr t id)*

Sends warning message to the source node when called from the *rt\_resolve* function.

*Recvwarning (packet \*p)*

Receives the warning message and forwards to the source if it is an intermediate node. If it is the source node, it check if the route is a route to the internet and initiates the reactive GW discovery by multicasting a RREQ\_I to the GW nodes.

## 2.In AODV.h

The *sendwarningprev (aodv\_rt entry \*rt0, nsaddr\_t id)* and *receivewarning (Packet \*p)* functions are declared as member functions.

## 3.In AODV packet.h

The AODVTYPE\_WARNING packet type and its header (*hdr\_aodv\_warn*) are defined. The header contains 8 bits for Packet type, 32 bits for the warning source IP address and 32 bit for the destination address to which the path to the internet is suspected to be broken.

## 4.In the AODV\_rt\_table.h

### AODV Neighbor

Two, three element arrays to store power and arrival time of data packets in *rt\_resolve (packet \*p)* are defined. Additionally, a variable to count the number of received signal power is added.

### AODV\_rt\_entry

To monitor the status of the route, whether it is suspected to fail or not, a variable is added in this class.

## 8.2.Result and Discussion

In this sub section the simulation result of the predictive GW discovery compared to the reactive GW discovery is presented. The PDF, NRL, ROUTING OVERHEAD and AVERAGE END-TO-END DELAY metrics are used for evaluation. Scenario 3 and 4 are used for testing the proposed discovery algorithm. However, if not for time factor, it is recommended to check using all the metrics and the scenarios that are described in chapter 6.

### 8.2.1 Results Obtained Using Scenario three

The third scenario represents a mobility condition at low social interaction and lower density (i.e. 30 MNs placed in 1200m by 1200m simulation area). The speed of the traveling nodes has been varied from 0 to 30m/s at steps of 5m/s.



Figure 8.3: Packet delivery fraction vs. traveling nodes' speed



Figure 8.4: Average end-to-end delay vs. traveling nodes' speed

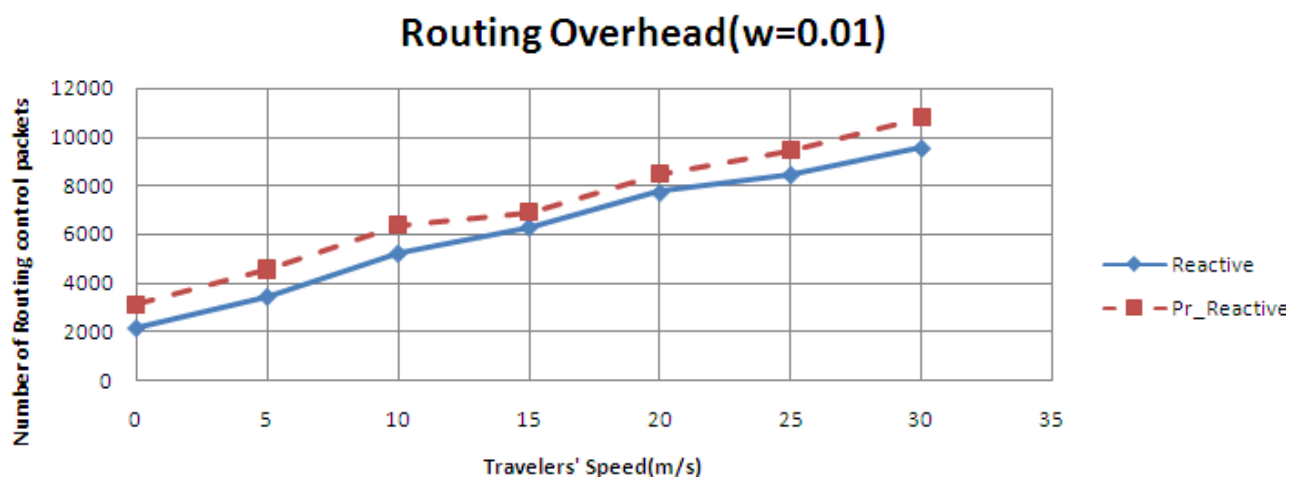


Figure 8.5: Routing overhead vs. traveling nodes' speed

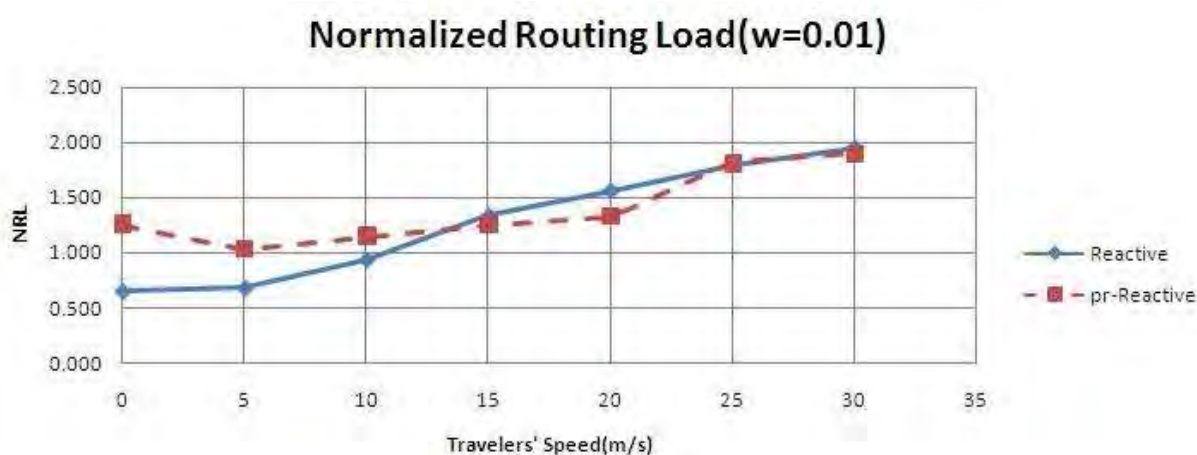


Figure 8.6: Normalized routing load vs. traveling nodes' speed

The predictive GWDA's PDF value is observed to be up to a maximum of 22% better than the original reactive GWDA as shown in figure 8.3. This is the result of link failure prediction and performing discovery in advance to maintain the route to the internet. The predictive GW discovery has shown a nearly equivalent performance in terms of PDF value when compared to the proactive GWDA in the previous chapter (figure 7.27). The average end-to-end delay is also improved up to a maximum value of 36%. This is because, starting discovery in advance reduces the time spent from error occurrence until a new route to the internet is discovered and hence the time packets spent waiting in MNs' queues. The routing overhead is higher by 21% on average over all speed ranges, because of the additional warning messages introduced in the network. The proactive GWDA's routing overhead shown in figure 7.29 was 38% greater than the reactive one. In this regard, the predictive GW discovery is able to cut the routing overhead by 7% compared to the proactive GWDA. The NRL of the predictive GWDA is high up to 15m/s travelers' speed but after that it is nearly equivalent to the reactive GWDA, corresponding to the relative increase in the PDF of the predictive GWDA.

### 8.2.2 Results Obtained Using Scenario three

Scenario four represents a mobility condition at high social interaction and lower density as scenario three. The speed of the traveling nodes has been varied from 0 to 30m/s at steps of 5m/s.



Figure 8.7: Packet delivery fraction vs. traveling nodes' speed



Figure 8.8: Average end-to-end delay vs. traveling nodes' speed

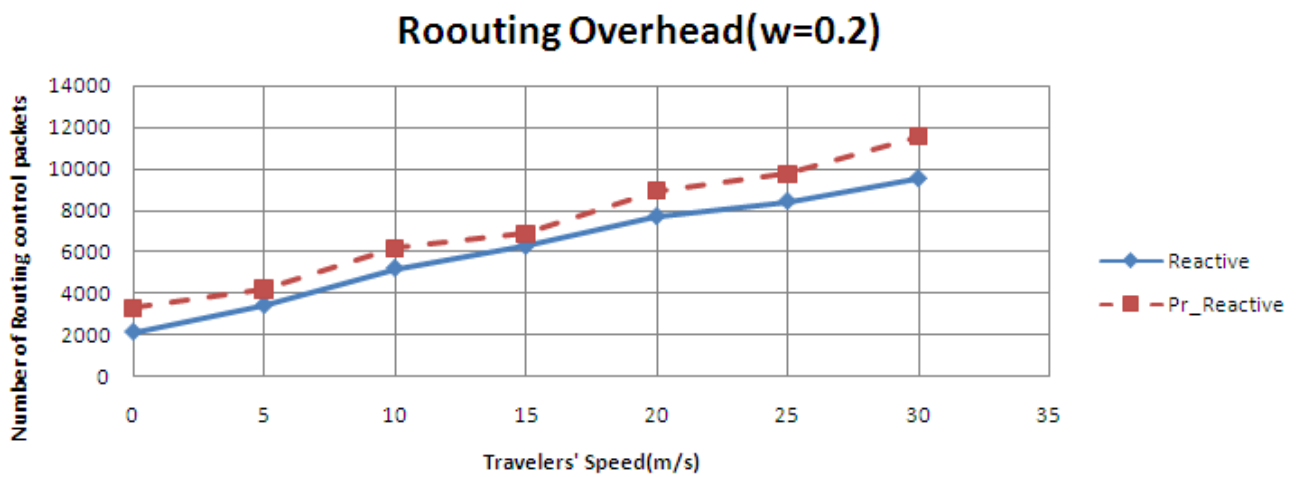


Figure 8.9: Routing overhead vs. traveling nodes' speed

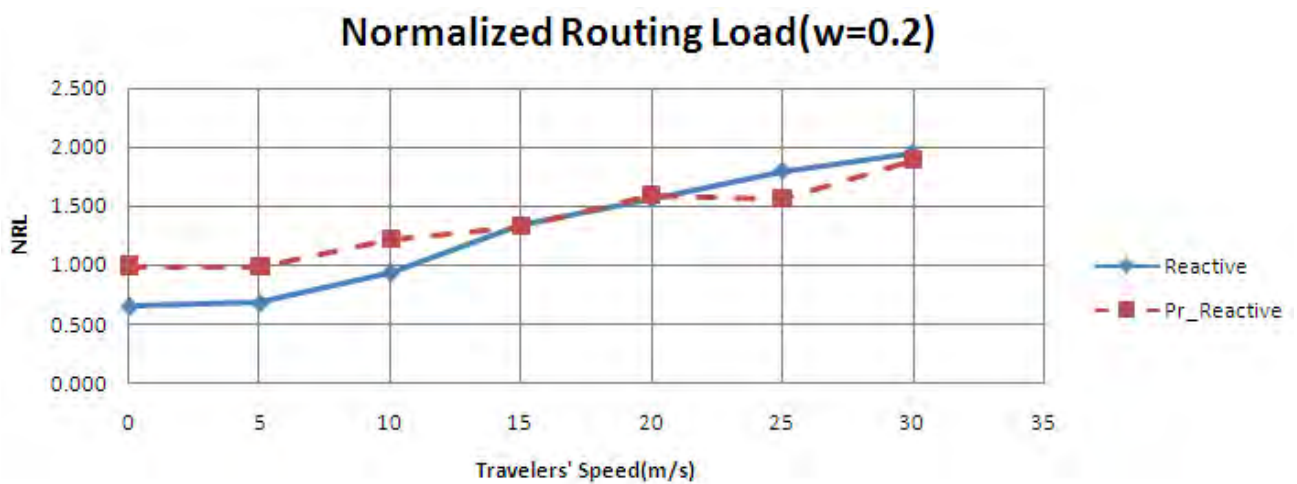


Figure 8.10: Normalized routing load vs. traveling nodes' speed

The PDF value is observed to be up to a maximum of 16% better than the original reactive GWDA in the high social interaction environment relative to scenario 3, as shown in figure 8.7. The improvement is the result of link failure prediction and performing discovery in advance to maintain the route to the internet. The decline in the maximum value is because of the reactive GWDA's improvement for travelers' speed less than 15m/s. This is also the positive impact of the high social interaction for distributing nodes in the simulation area in such less dense environment (30 nodes per 1200m square area). The predictive GW discovery has shown a nearly equivalent performance in terms of PDF value when compared to the proactive GWDA in the previous chapter (figure 7.37). The average end-to-end delay is also improved up to a maximum value of 36% like the case in scenario 3. Similar justification holds true here as well. The routing overhead is higher by 22% on average over all speed ranges. 1% increase from scenario 3. This is a result of the additional warning messages introduced in the network and the negative impact of the high social interaction at higher speeds. The proactive GWDA's routing overhead shown in figure 7.29 was 37% greater than the reactive one. In this regard, the predictive GW discovery is able to cut the routing overhead by 5% compared to the proactive GWDA. The NRL of the predictive GWDA is high up to 15m/s travelers' speed but after that it is nearly equivalent to the reactive GWDA, corresponding to the relative increase in the PDF of the predictive GWDA like scenario three. But the NRL value is lesser than the case in scenario three for lower speeds (less than 10m/s) because of the fact that the high social interaction helps to distribute MNs in the simulation area and hence facilitate propagation of control messages and data packets exchange to the internet through the GW nodes.

### **8.3 Summary of the Results**

This section presented the summery of the simulation results for the proposed solution in the previous section.

The predictive GWDA's PDF value is observed to be up to 16%-22% better than the original reactive GWDA as the result of link failure prediction and performing discovery in advance maintaining the route to the internet. The average end-to-end delay is also improved up to 36% as a result of starting discovery in advance that can minimize the disconnection time from internet. This also helps to shorten the time packets spent waiting in MNs' queues. The routing overhead is higher by 21-22% on average over all speed ranges, because of the additional warning messages introduced in the network. The NRL of the predictive GWDA is high up to 15m/s travelers' speed but after that it is nearly equivalent to the reactive GWDA, corresponding to the relative increase in the PDF of the predictive GWDA.

Compared to the proactive GWDA, the predictive GWDA has shown a nearly equivalent performance, but with 5%-7% lesser routing overhead.

The next chapter presents the conclusion of this thesis work and the recommended future works.

## **Chapter Nine: Conclusion and Recommendation**

### **9.1 Conclusion**

Performance investigation of the three AODV based internet GWDA is performed in this thesis work. The best one in terms of resisting nodes' mobility effect is also identified. The community based mobility model is used as the movement of the MNs needs to be more realistic. This work also proposed and also tested a GWDA which is based on prediction of a link failure on the route to the internet.

The performance evaluation of the reactive, proactive and hybrid GWDA is performed by using four scenarios and about 10 metrics. The social interaction (high/low) is directly related to the movement of MNs in the simulation area and hence it is considered for the scenario design. The density of nodes is also another factor considered to design the scenarios as the impact of mobility is high in less dense environment. This is because nodes can easily move outside the radio range of each other causing frequent GW route failures. The travelers' speed is also varied from 0m/s up to 30m/s to study how each GWDA responds for high speed mobility condition. Handover time (total in percentage and average) between GW nodes, discovery time( total in percentage and average), expanding radial search time, disconnection time, PDF, Average end-to-end delay, NRL and routing overhead metrics are used for a comprehensive performance analysis.

For all the scenarios, as speed of the travelling nodes is varied from 0m/s up to 30m/s at steps of 5m/s, the algorithms performance shows improvement for lower speed ranges. In the case of high social interaction, lower speed is less than 5m/s and for low social interaction it could be up to 10m/s. At these speeds algorithms attain their best performances. After that, their performance generally declines. At lower speeds the increase in travelers' speed and the high social interaction have a positive impact by distributing MNs in the simulation area and hence facilitating data and control messages exchange through the GW nodes.

But higher speeds (especially after 10m/s) and high social interaction aggravates the impact of mobility causing frequent default route failures. The less dense environment (30 MNs per 1200m square area) is also observed to decline the performance of all algorithms.

When the algorithms' relative performance is considered, the proactive GWDA is the best one among the three. It is less affected by the increase in the speeds of the travelling nodes. However, it has a relatively higher routing overhead (up to a maximum of 38%) compared to the least performing reactive GWDA. The hybrid GWDA has shown an average performance in the first two scenarios and performed more like the reactive GWDA in the less dense environments (the third and fourth scenarios).

The predictive GWDA proposed and tested in this thesis work is based on improving the reactive GWDA by performing prediction and GW discovery before GW route fails.

This algorithm has shown up to a maximum of 22% improvement in PDF compared to the reactive GWDA. However, its performance relative to the proactive GWDA is nearly equivalent but has 5-7% lesser routing overhead.

Therefore, the work concludes, the proactive GWDA is less affected with mobility than the reactive and hybrid GWDAs and the predictive GWDA is a better solution considering its lesser routing overhead.

## **9.2 Future Work**

Some of the points can be pointed out as future outlooks; one of the areas for future work is to incorporate the effect of fading, noise and interference with appropriate model and study their relationship with the mobility of nodes' effect on the performance of the algorithms.

Modification of the GWDAs to store alternative paths to the internet and update them using unicast messages. So that, when the current link fails, it can use the alternative path. This approach could be interesting to develop a more mobility effect resistive GWDA.

Another open area, related to this work is to change the GW selection criteria from number of hops to route stability criteria, which also considers the signal strength of the alternative RREP\_I packet or the GWADV packets from the GW nodes. This can help to obtain a more efficient handoff from one GW to the other and can help to reduce the frequent handover and disconnection from internet connection.

Finally, in this work the simulation based analysis for studying the impact of nodes' mobility on AODV based internet GWDAs has been performed. And a predictive GWDA is proposed and tested. Performing analysis on the predictive GWDA using all the scenarios and metrics indicated in chapter-6 and also conformation of the outcomes found on real test bed is another important work.

## References

- [1]. Stephano Basagni. Marco Conti, Silvia Giordano and Ivan Stojmenovic', "Mobile Ad Hoc networking" by IEEE press 2004
- [2]. Carlos de Moraes cordeiro Dharma Prakash Agrawal. "Ad Hoc & Sensor Networks Theory and Application", World scientific publishing 2006.
- [3]. M.Rosenschon et al. "Gateway Discovery Algorithm for Ad-Hoc Networks using HELLO Messages". IWWAN 2005, London, May 2005.
- [4]. Krishna Gorantala. "Routing Protocols in Mobile Ad-hoc Networks", Master's Thesis, Ume°a University, SWEDEN, June 15, 2006
- [5]. Matthias Rosenschon. "Internet Gateway Discovery for Mobile Ad Hoc Networks", CITY University London, 2 October 2007.
- [6]. Ali Hamidan. "A Study of Internet Connectivity for Mobile Ad Hoc Networks in NS 2", LUND University London, January 2003
- [7]. Charles E. Perkins and Pravin Bhagwat. The "*ns Manual*", A Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC, December 19, 2008
- [8]. Ashenafi Yadessa, Dr. Tamrat Behayle, Dr. Kumdha Raimond, "Impact of node mobility on the performance of mobile ad hoc network routing protocols". AAU computer Engineering Dpt., March, 2008
- [9]. S. Gupta. "Performance Evaluation of Ad-Hoc network routing protocols". Available at: <http://www.cs.utk.edu/~gupta/Adhoc.doc>. Accessed (August, 2009)
- [10]. M. Rosenschon et al. "Performance Comparison of Gateway Discovery Algorithms in Ad Hoc Networks with Mobile Nodes", IWWAN, London, April 2005.
- [11]. Sehrish Abrejo et.al, "Analysis of MANET Routing Protocols using Scenario Based Mobility Models", Israel University, Hyderabad. Pakistan, May 2007.
- [12]. Rakesh Kumar et.al, "An Efficient Gateway Discovery in Ad Hoc Networks for Internet Connectivity", Department of Electronics and

Computer Engineering Indian Institute of Technology Roorkee,  
Uttarakhand, India- 247667, 2007

- [13]. Mirco Musolesi and Cecilia Mascolo, "A Community Based Mobility Model for Ad Hoc Network Research" In *Proc. of the 2nd ACM/SIGMOBILE International Workshop on Multi-hop Ad Hoc Networks: from theory to reality (REALMAN'06)*, Florence, Italy, May 2006.
- [14]. <http://www.cs.ucsb.edu/~ebelding/txt/globalv6.txt> Viewed on Nov 10, 2009.
- [15]. Francisco J. Ros et al. "Performance Evaluation of Interconnection Mechanisms for Ad Hoc Networks across Mobility Models", Dept. of Information and Communications Engineering, University of Murcia, Spain, JNW 1(2): 9-17 (2006).
- [16]. R. Wakikawa, J. Malinen, C. Perkins, A. Nilsson, and A. Tuominen, "Global Connectivity for IPv6 Mobile Ad Hoc Networks," Internet-Draft "draft-wakikawa-manet-globalv6-03.txt". Oct. 2003.
- [17]. C. Jelger, T. Noel, and A. Frey, "Gateway and Address Auto configuration for IPv6 Ad Hoc Networks," Internet-Draft "draft-jelger-manet-gateway-Autoconf-v6-02.txt", Apr, 2004.
- [18]. S. Singh, J. Kim, Y. Choi, K. Kang, and Y. Roh, "Mobile Multi-GW Support for IPv6 Mobile Ad Hoc Networks," Internet-Draft, "draft-sinhg-manet-mmng-00.txt", June 2004.
- [20]. D. Cavalcanti, C. Cordeiro, D. Agrawal, B. Xie and A. Kumar, "Issues in Integrating Cellular Networks, WLANs, and MANETs: A Futuristic Heterogeneous Wireless Network", *IEEE Wireless Communications Magazine*, Vol. 12, No. 3, pp. 30-4, April 2005.
- [21]. Bin Shen et.al, "Performance Comparison and Analysis of Three GW Discovery Protocols of Internet Connectivity for Ad Hoc Network", Wuhan institute of technology, Wuhan 430073, China, 2006.
- [22]. Jahanzeb Farooq, "Mobility and Internet Connectivity in Mobile Ad Hoc Networks", Department of Computing Science Umeå University,

Sweden, June 2005.

- [23] Jin Xi et.al, "Wireless Multi-hop Internet Access: Gateway Discovery, Routing and Addressing", n Proc. Intern. Conf. on Third Generation Wireless and Beyond (3Gwireless), (San Francisco, USA), May 2002.
- [24] Pedro M.Ruiz et.al,"Adaptive Gateway Discovery Mechanisms to Enhance Internet Connectivity for Mobile Ad Hoc Networks", *Espinardo, Murcia, Spain*, March 3, 2005.
- [25]. T. Goff, N. B. Abu-Ghazaleh, D. S. Phatak, R. Kahve-cioglu. "Preemptive routing in ad hoc networks" , In Proc. ACM MOBICOM 2001, Rome, Italy, 2001, pp.43—52.
- [26]. Q. Liang, T. Kunz, "Increasing packet delivery ratio in DSR by link prediction", IEEE, proceedings of the 36<sup>th</sup> Annual Hawaii International Conference on System science, Vol.9, Issue, 6-9 Jan. 2003, pp-10-19
- [27]. Sathyaraj B.H, Doss R.C., "Route maintenance using mobility prediction for mobile ad hoc networks", mobile Adhoc and sensor system conference, IEEE international conference pp-6, 2005.
- [28]. S. Boukli Hacence, A. Lehireche, A. meddahi, "Predictive preemptive Ad hoc on-demand distance vector Routing", Malaysia journal of computer science, vol. 19(2), 2006,pp. 189-195.
- [29]. Kuo-Hsing Chaing, Shenoy, N., "A 2-D random-walk mobility model for Location management studies in wireless networks". Vehicular 424. Technology, IEEE Transactions on Volume 53, Issue 2, March 2004 Page(s):413-424
- [30]. X. Hong, M. Gerla, G. Pei, C.-C. Chiang, "A group mobility model for ad hoc wireless networks", in Proc. ACM Intern. Workshop on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM), August 1999.
- [31]. C. Bettstetter, "Smooth is better than sharp: a random mobility model for simulation of wireless networks", in Proc. ACM Intern. Workshop on Modeling, Analysis, and simulation of wireless and Mobile systems

- (MSWiM), Rome, Italy, July 2001.
- [32]. M. de Gergg, M. van Kreveld, M. Overmars and O. Schwarzkopf, “Computational Geometry: Algorithms and Applications”, Springer Verlag, 2000.
- [33]. A Jardosh, E. M. Belding-Royer, K.C. Almeroth, S. Suri, “Towards Realistic Mobility Models for Mobile Ad Hoc Networks”, ACM Mobicom 2003, San Diego, CA, September 2003.
- [34]. Camp, T., Boleng, J., and Davies, V., “A survey of Mobility Models for Ad Hoc Networks Research, Wireless Communications & Mobile Trends and Computing (WCMC)”, Special issue on Mobile Ad Hoc Networking: Research, Applications, 2002, p. 483-502.
- [35]. Fan Bai, Narayanan Sadagopan, Ahmed Helmy, “The IMPORTANT Framework for analyzing the impact of mobility on performance of routing Protocols for Ad hoc network”, Elsevier Ad Hoc Networks, 2003, P.383- 403.
- [36]. Minkyong Kim, David Kotz and Songkuk Kim, “Extracting mobility model from real user traces”, IEEE InfoComm 2006.
- [37]. D. J. Watts, “Small Worlds: The Dynamics of Networks between Order Protocols for Ad Hoc Mobile Wireless Networks”, IEEE Personal Communications, Vol. 6, No. 2, pp. 46-55, April 1999.

## **Appendix A                      TCL Script for Simulation**

```
# =====
# Define options
# =====
set opt(namfile)          out.nam
set opt(tracefile)        out.tr
set opt(x)                1200;          #x dimension of the topography
set opt(y)                1200;          #y dimension of the topography
set opt(wirelessNodes)    30;            #mobile nodes
#-----Traffic srcs-----
set opt(cp)               "./traffic/traffic-srcs"
#-----Mobility-----
#set opt(sc)              "./mobility/w0.01-1000-v0-G16-c21-s1"
#set opt(sc)              "./mobility/w0.2-1000-v0-G16-c21-s1"
set opt(sc)               "./mobility/w0.01-1200-v0-G16-c21-s1"
#set opt(sc)              "./mobility/w0.2-1200-v30-G16-c21-s1"
#-----
set opt(wiredNodes)       4;             #hosts and routers
set opt(gatewayNodes)     2;             #gateways
set opt(stop)             300.0;         #simulation time
set opt(gw_discovery)     reactive;      #GWdiscovery method
# =====
#Initialize Global Variables
#=====
#create a simulator object
set ns_ [new Simulator]
$ns_ color 0 Brown
$ns_ use-newtrace
#Define The Hierachial Topology Structure
#=====
$ns_ node-config -addressType hierarchical
#Nbr of domains
#=====
AddrParams set domain_num_ 4
#Nbr of clusters (=subdomains) in each domain
#=====
AddrParams set cluster_num_ {1 1 1 1}
#Nbr of nodes in each cluster
#=====
AddrParams set nodes_num_ {2 2 13 13}
#create trace objects for ns and nam
#=====
set nstrace [open $opt(tracefile) w]
$ns_ trace-all $nstrace
set namtrace [open $opt(namfile) w]
$ns_ namtrace-all-wireless $namtrace $opt(x) $opt(y)
#create a topology object and define topology
#=====
set topo [new Topography]
$topo load_flatgrid $opt(x) $opt(y)
#Choose method for GWdiscovery
#=====
if {$opt(gw_discovery) == "proactive"}
    {Agent/AODV set gw_discovery 0}
```

Performance Evaluation and Nodes' Mobility Effect Analysis on AODV based Internet Gateway  
Discovery Algorithms in Social Networks

```
if {$opt(gw_discovery) == "hybrid"}
  {Agent/AODV set gw_discovery 1}
if {$opt(gw_discovery) == "reactive"}
  {Agent/AODV set gw_discovery 2}
if {{$opt(gw_discovery) == "predictive"}}
  {Agent/AODV set gw_discovery 3}
#create God (General Operations Director)
#=====
set god_ [create-god [expr
$opt(wirelessNodes)+$opt(gatewayNodes)+$opt(wiredNodes)]]
#create wired nodes
#=====
  set router1 [$ns_ node 0.0.0]
  set host1 [$ns_ node 0.0.1]
  set router2 [$ns_ node 1.0.0]
  set host2 [$ns_ node 1.0.1]
#Configure for GWand Mobile Nodes
#=====
#Use hierarchical addresses for GWs and MNs
#=====
#AODV is folder name of the enhanced AODV
#=====
$ns_ node-config -adhocRouting AODV
$ns_ node-config -llType LL
$ns_ node-config -macType Mac/802_11
$ns_ node-config -ifqType Queue/DropTail/PriQueue
$ns_ node-config -ifqLen 50
$ns_ node-config -antType Antenna/OmniAntenna
$ns_ node-config -propType Propagation/TwoRayGround
$ns_ node-config -phyType Phy/WirelessPhy
$ns_ node-config -topoInstance $topo
$ns_ node-config -channel [new Channel/WirelessChannel]
$ns_ node-config -agentTrace ON
$ns_ node-config -routerTrace ON
$ns_ node-config -macTrace ON
$ns_ node-config -movementTrace OFF
#configure for gateways
#=====
$ns_ node-config -wiredRouting ON
#create gateway
#=====
  set gw1 [$ns_ node 2.0.0]
  set gw2 [$ns_ node 3.0.0]
#set initial coordinates
#=====
$gw1 set X_ 250.0
$gw1 set Y_ 200
$gw1 set Z_ 0.0
$gw2 set X_ 850.0
$gw2 set Y_ 200
$gw2 set Z_ 0.0
#=====
$ns_ at 0.00 "$gw1 setdest 200 200 0"
$ns_ at 0.00 "$gw2 setdest 800 200 0"
```

```
#configure for mobile nodes
#=====
$ns_ node-config -wiredRouting OFF
#create mobile nodes in the same domain as gw(0)
#=====
set temp {2.0.1 2.0.2 2.0.3 2.0.4 2.0.5 2.0.6 2.0.7 2.0.8 2.0.9 2.1.0
          2.1.1 2.1.2 2.1.3 2.1.4 2.1.5\3.0.1 3.0.2 3.0.3 3.0.4 3.0.5
          3.0.6 3.0.7 3.0.8 3.0.9 3.1.0 3.1.1 3.1.2 3.1.3 3.1.4 3.1.5}
for {set i 6} {$i < $opt(wirelessNodes)-9} {incr i} {
    set node_($i) [$ns_ node [lindex $temp [expr $i-6]]]
    $node_($i) base-station [AddrParams addr2id [$gw1 node-addr]]}
for {set i 21} {$i < $opt(wirelessNodes)+6} {incr i} {
    set node_($i) [$ns_ node [lindex $temp [expr $i-6]]]
    $node_($i) base-station [AddrParams addr2id [$gw2 node-addr]]}
# give color for hosts and the gateway
#=====
$router1 color blue
$router2 color blue
$host1 color blue
$host2 color blue
$gw1 color red
$gw2 color red
$gw1 shape box
$gw2 shape box
$router1 shape square
$router2 shape square
$host1 shape hexagon
$host2 shape hexagon
#Give label for hosts, the GW and the nodes
#=====
$ns_ at 0.0 "$router1 label \"ROUTER\""
$ns_ at 0.0 "$router2 label \"ROUTER\""
$ns_ at 0.0 "$host1 label \"HOST\""
$ns_ at 0.0 "$host2 label \"HOST\""
$ns_ at 0.0 "$gw1 label \"GATEWAY\""
$ns_ at 0.0 "$gw2 label \"GATEWAY\""
for {set i 6} {$i < $opt(wirelessNodes)+6} {incr i} {
    $ns_ at 0.0 "$node_($i) label \"MN $i\""
}
# Load traffic model from file indicated by opt(cp)
#=====
puts "Loading traffic model..."
source $opt(cp)
# Load mobility model from file indicated by opt(sc)
#=====
puts "Loading mobility model..."
source $opt(sc)
#Set Nodes' Initial Position
#=====
for {set i 6} {$i < $opt(wirelessNodes)+6} {incr i} {
    $ns_ initial_node_pos $node_($i) 30}
#Connection between routers and hosts
#=====
$ns_ duplex-link $router1 $router2 100Mb 2ms DropTail
$ns_ duplex-link-op $router1 $router2 orient right
$ns_ duplex-link $router1 $host1 100Mb 1.80ms DropTail
$ns_ duplex-link-op $router1 $host1 orient down
```

```
$ns_ duplex-link $router2 $host2 100Mb 1.80ms DropTail
$ns_ duplex-link-op $router2 $host2 orient down
$ns_ duplex-link-op $router1 $router2 queuePos 0.5
#Connection between gateways and routers
#=====
$ns_ duplex-link $router1 $gw1 100Mb 2ms DropTail
$ns_ duplex-link-op $router1 $gw1 orient up
$ns_ duplex-link $router2 $gw2 100Mb 2ms DropTail
$ns_ duplex-link-op $router2 $gw2 orient up
#Tell Nodes When The Simulation Ends
#=====
for {set i 6} {$i < $opt(wirelessNodes)+6} {incr i} {
    $ns_ at $opt(stop).01 "$node_($i) reset";}
    $ns_ at $opt(stop).01 "$router1 reset";
    $ns_ at $opt(stop).01 "$host1 reset";
    $ns_ at $opt(stop).01 "$router2 reset";
    $ns_ at $opt(stop).01 "$host2 reset";
    $ns_ at $opt(stop).01 "$gw1 reset";
    $ns_ at $opt(stop).01 "$gw2 reset";
    $ns_ at $opt(stop).0001 "stop"
    $ns_ at $opt(stop).0002 "puts \"NS EXITING...\" ; $ns_ halt"
    proc stop {} {
        global ns_ nstrace namtrace opt
        $ns_ flush-trace
        close $nstrace
        close $namtrace
        puts "Running nam with $opt(namfile)..."
        exec nam $opt(namfile) &
        exit 0}
    puts "Starting simulation..."
    $ns_ at 0.0 "$ns_ set-animation-rate 5ms"

$ns_ run
```

## **Appendix B, Script for Reading trace Files**

```
#PDF, NRL, AEED and ROUTING OVERHEAD
#=====
BEGIN{send=0;G1recvs=0;G2recvs=0;routing_packets=0.0;droppedBytes
=0;noerrors=0;droppedPackets=0;highest_packet_id=0;sum=0;avghop=0;
recvnum=0;dropq=0;} time = $3; packet_id = $41;if
(($1=="s")&&($19=="AGT" )&&($35=="cbr")) send++;
if (($1=="s")&&($35=="cbr")&&($19=="MAC"))avghop++;
if (($1 == "r")&&($5=="4")&&($19=="MAC")&&($35=="cbr" ))
G1recvs++;if(($1=="r")&&($5=="5")&&($19== "MAC")&&($35=="cbr"))
G2recvs++;if(($1 == "d")&&($19=="IFQ"))dropq++;
if(start_time[packet_id]==0)start_time[packet_id]=time;
if(($1=="r")&&($5=="4")&&($19=="MAC")&&($35=="cbr"))
{end_time[packet_id]=time;}else if (($1=="r")&&($5==
"5")&&($19=="MAC")&&($35=="cbr" )) {end_time[packet_id] = time; }
else{end_time[packet_id]= -1;}if(($1=="s"||$1=="f")&&$19=="RTR" &&
$35 == "AODV") routing_packets++;#packets at network layer
if(($1=="f")&&$19=="RTR"&&$35=="AODV"&&$37=="40")noerrors++;
if(($1=="d")&&($35=="cbr")&&($37>0))
{droppedBytes=droppedBytes+$37; droppedPackets=droppedPackets+1;}
#find the number of packets in the simulation
if (packet_id > highest_packet_id)highest_packet_id = packet_id;}
END{for ( i in end_time ){start = start_time[i];end =
end_time[i];packet_duration=end -start;if(packet_duration>0){ sum
+= packet_duration;recvnum++;}} if(recvnum>0){delay=sum/recvnum;
NRL = routing_packets/recvnum;} # NRL
if(send>=0){PDF = ((G1recvs+G2recvs)/send)*100;}
else {printf("NO PACKET HAS BEEN EXCHANGED\n");}
printf("send..= %.3f\n",send);printf("G1recv ..=
%.3f\n",G1recvs);printf("G2recv..= %.3f\n",G2recvs);
printf("routingpkts..=%.3f\n\n",routing_packets);
printf("PDF ..= %.2f\n",PDF);printf("Average e-e delay(Sec)..=
%.3f\n\n",delay);printf("Delay ..= %.3f\n\n",sum);
printf("NR(Normalized routing load) ..= %.3f\n",NRL);
printf("No of errors ..= %.2f\n",noerrors);# /35
printf("No. of dropped data (packets) = %d\n",droppedPackets);
printf("No. of dropped data b/c full queue(bytes) = %d\n",dropq);
printf("avrage hop count..= %d\n",avghop);
#printf("Total Number of packets generated..%d\n",
highest_packet_id);}

#AVERAGE AND PERCENTAGE HANDOVER PER-NODE
#=====
BEGIN{GW7=0;r7=0;PacketSentTime7=0;LastReceiveTime7GW1=0;LastRecei
veTime7GW2=0;CondOne7=0;CondTwo7=0;tTwo7=0;tOne7=0;count7=0;TotalH
andoverTime7=0;packetID7H=-1;rcv7=0;}
```

```

{if(($1=="s")&&( $5=="7")&&($19 == "AGT")&&($35=="cbr")&&($3 <300))
{ PacketSentTime7=$3;packetID7H=$41;}
if (($1=="r")&&( $5== "4")&&($41==packetID7H)&&($19=="MAC")&&(
$35=="cbr" )&&( $37 == "532")&& ($3 < 300)){GW7=4;
if(rcv7==0){CondOne7=1;GW7=4;}rcv7=rcv7+1;LastReceiveTime7GW1=$3;
if(rcv7<2) printf("CondOne7= %.d\n",CondOne7);}
else if(($1=="r")&&( $5=="5")&&( $19=="MAC")&&($41== packetID7H)
&&( $35=="cbr" )&&( $37 == "532")&& ($3 < 300)){GW7=5;if(rcv7==0)
{CondTwo7=1;GW7=5;}rcv7=rcv7+1;LastReceiveTime7GW2=$3;if(rcv7<2)
printf("CondTwo7= %.d\n",CondTwo7);}else { } if(rcv7!=0)
{if((GW7 == 4) && (CondOne7==1)) {tOne7=LastReceiveTime7GW1;}
else if((GW7==5)&&(CondTwo7==1)){tOne7=LastReceiveTime7GW2;}
else {if(CondOne7==1){tTwo7=LastReceiveTime7GW2;CondTwo7=1;
CondOne7=0;printf("7's GW1(Sec)= %.4f\n",tOne7);
printf("7's GW2(Sec)= %.4f\n",tTwo7);count7++;}
else{tTwo7=LastReceiveTime7GW1;CondTwo7=0;CondOne7=1;
printf("7's GW2(Sec)= %.4f\n",tOne7);
printf("7's GW1(Sec)= %.4f\n",tTwo7);count7++;}
printf("H-TIME(Sec)= %.4f\n", (tTwo7-tOne7));r7++
TotalHandoverTime7=TotalHandoverTime7+(tTwo7-tOne7);}}}
END{ printf("-----NODE-7-----\n");
printf("\nTotal Handover Time(7)..HT= %.4f", (TotalHandoverTime7));
printf("\nNumber of Handovers(7).....NoH= %d",count7);
if(count7>0) AvgHT7=(TotalHandoverTime7/count7);else AvgHT7=0;
if(count7>0) printf("\nAvg. Handover time(7)Avg-HT= %.4f",AvgHT7);
printf("\nHandover Time out of Total Simulation Time in
percentage(7).....HTin%=.4f\n", (TotalHandoverTime7*100)/300);
}
#AVERAGE AND PERCENTAGE DISCOVERY, DISCONNECTION AND EXPANDING
#RING SEARCH TIMES PER-NODE
#=====
BEGIN
{IdleC7=0;ReqTime7;TotalIdleTime7=0;ReplyTime7=0;r7=0;TDiscTime7=
0;DiscC7=0;RadialDSent7=0;errorT7;rr7=0;PkDrop7=0;node7=0;radrq7=0
;radrp7=0;RadialRequ7=0;thirtyhop7=0;DradialStartTime7=0;RadialReq
uLast7=0;Idle7=1;GenC7=0;DGenStarted7=0;DataRadial7=1;RadialGWData
RecvTime7=0;k7=1;}
#-----Advertisement-----
if (( $1 == "s") && ( $5 == "5" )&&( $19 == "MAC" ) && (
$35=="AODV" )&&( $37 == "100")&&($61=="GWADV")($3<300))
{printf("GW2(Node_5) Advertised at(Sec)= %.7f\n", $3);}
else if (( $1 == "s") && ( $5 == "4" )&&( $19 == "MAC" ) && (
$35=="AODV" )&&( $37 == "100")&&($61=="GWADV")&&($3<300))
{printf("GW1(Node_4) Advertised at(Sec)= %.7f\n", $3);}

```

```
#-----Radial Search-----
if (($1=="s")&&($5=="7")&&($19=="AGT")&&(GenC7==0)&&($35=="cbr"
)&&( $37 == "512")&&($31 == "8388610.0")&&($3<300))
{GenC7=GenC7+1;DGenStarted7=1;DradialStartTime7=$3;packetID7=$41;
printf("-----Node 7 started generating Data at
Time(Sec)=%.7f\n", $3)}
if (($1=="s")&&($5=="7")&&($19=="MAC")&&(RadialDSent7!=1)&&(
$35=="AODV" )&&( $37 == "100")&&($31 == "8388610.255")&&($3<300))
{if(($43=="1")&&(DGenStarted7==1)&&(k7==1)&&($61=="REQUEST"))
{printf("Node 7's Radial Search Started at time(Sec)=
%.7f\n", $3);RadialRequ7=$3;radrq7=1;radrp7=0; k7=k7+1}
if(($43=="30")||($43=="28")||($43=="27"))&&($61=="REQUEST")&&(DGe
nStarted7==1))
{printf("Node 7's radial search at max hop Request(Sec)=
%.7f\n", $3);radrq7=1;radrp7=0;thirtyhop7=1;DGenStarted7=0;}
if(($43 >= 2)&&($43<=29)&&(thirtyhop7 ==
0)&&(DGenStarted7==1)&&($61=="REQUEST"))
{printf("Radial search continued(Sec)= %.7f", $3);
printf("-----hop=%d\n", $43);radrq7=1;radrp7=0;}}
if(($1=="r")&&($5 == "7")&&($19=="RTR")&&($35=="AODV")&&($33==
"8388610.255")&&($61=="RREP_I")&&( $37 == "44")&&($3<300))
{if(radrq7 == 1){printf("Radial RREP_I received at time
(Sec)=%.7f\n", $3); radrq7=0;radrp7=1;
if(thirtyhop7==1){RadialRequLast7=$3; printf("Initial Radial
Search latency(Sec)=%.7f\n", RadialRequLast7-RadialRequ7);}}}
if(($1=="r")&&(DataRadial7==1)&&( $5 == "7" )&&( $19 == "RTR" ) &&
( $35=="AODV" )&&($33 == "8388610.255")&&( $37 == "44")&&($3<300))
{ printf("last RREP_I for radial search of node 7 is
(Sec)*****=%.7f\n", $3);radialreplytime7=$3; }
if ( ( $1 == "r")&&(($5 == "5")||($5 == "4"))&&($41 ==
packetID7)&&($19 == "MAC" )&&(DataRadial7==1)&&($35=="cbr" )&&(
$37 == "532")&&($31 == "8388610.0")&&( $3 < 300))
{printf("7's radialreplytime7...=%.3f\n", radialreplytime7);
printf("7's RadialRequ7...=%.3f\n", RadialRequ7);
printf("7's last RREP_I-Radial REQUEST
Time(Sec).....*****.....=%.3f\n", (radialreplytime7-
RadialRequ7));printf("-----Node 7's Data Reached the
gateway(Sec)=%.7f\n", $3);DataRadial7=DataRadial7+1;RadialGWDataRec
vTime7=$3;RadialDSent7=1;}
#-----Error-----
if ( ( $1 == "s" ) && ( $5 == "7" )&&( $19 == "AGT" ) && (
$35=="cbr" )&&( $37 == "512")&&( $3 < 300)) #generated
{packetID7=$41;# printf("Node 7's generated packet id=
%d\n", packetID7);} if ( ( $1 ==
"d")&&($35=="cbr")&&($41==packetID7)&&($21 == "RET")&&($31 ==
"8388610.0")&&($3<300)) #Its packet dropped?
```

```

{node7=$5;pkssent=0;errorT7=$3;rr7=1;DSent7=0;DataRcvcc7=1;Adv7=1;A
dveRcv7=0;Idle7=1;AnotheRep7=0;
if(7==node7){printf("Node 7's Packet drop because of itself-
-->node= %d...",node7);printf("at time=--%.7f\n", $3);}
if(7!=node7){printf("Node 7's Packet drop because of intermediate
node--->=%d",node7);printf("at time=--%.7f\n", $3);}
if (($1=="r")&&(($5=="5")||($5=="4"))&&($41==packetID7)&&($19
=="MAC")&&(rr7==1)&&(DataRcvcc7==1)&&($35=="cbr")&&($37==
"532")&&($31=="8388610.0")&&($3<300))
{if ((rq7==0)&&(rr7==1)&&(AdveRcv7==1))
{if(AdvRcvTime7>=errorT7)
{printf("7's Idle Time==>(GWADV-errorT)(Sec)=%.3f\n",AdvRcvTime7-
errorT7);TotalIdleTime7=TotalIdleTime7+(AdvRcvTime7-errorT7);
IdleC7=IdleC7+1;Idle7=0;}else {printf("AdvRcvTime7<=errorT7\n");}
printf("7's Disc-Time==>(reply-reqT)(Sec)=%.3f\n\n",0);
TDiscTime7=TDiscTime7+0;DiscC7=DiscC7+1;rr7=0;}else
if((rq7==1)&&(rr7==1)&&(AdveRcv7==1))
{if(AdvRcvTime7>=ReqTime7){printf("7's Disc-Time==>(GWADV-
reqT)(Sec)=%.3f\n\n",AdvRcvTime7-ReqTime7);
TDiscTime7=TDiscTime7+(AdvRcvTime7-ReqTime7);
DiscC7=DiscC7+1;rr7=0;rq7=0;} else
{printf("AdvRcvTime7<=ReqTime7\n");}}
else if ((rq7==0)&&(rr7==1)&&(AdveRcv7==0))
{printf("7's Idle Time==>(Sec)=%.3f\n",0);
TotalIdleTime7=TotalIdleTime7+0;IdleC7=IdleC7+1;Idle7=0;
printf("7's Disc-Time==>(Sec)=%.3f\n",0);
TDiscTime7=TDiscTime7+0;DiscC7=DiscC7+1;rr7=0;}
else if ((rq7==1)&&(rr7==1)&&(AdveRcv7==0)&&(AnotheRep7==
1)){if(AnotheRep7Time>=ReqTime7){printf("7's Disc-
Time==>(AnotheRep-reqT)(Sec)=%.3f\n",AnotheRep7Time-ReqTime7);
TDiscTime7=TDiscTime7+(AnotheRep7Time-ReqTime7);
DiscC7=DiscC7+1;rr7=0;rq7=0;AnotheRep7=0;}
else {printf("AnotheRepTime7<=ReqTime7\n");}}
else{printf("NOTHING TO COMPUTE AT 7'S DATA DELIVERY\n");}
printf("_____Node 7's Data Reached the
gateway(Sec)=%.7f\n", $3);DataRcvcc7=DataRcvcc7+1;GWDataRecvTime7=$
3;DSent7=1;rq7=0;}if (($1=="r")&&($5=="7")&&($19=="RTR"
)&&(rr7==1)&&(Adv7==1)&&($61=="GWADV")&&($3<300)) #delay
{printf(".....Node 7 Recived Advertisement After Error at
Time(Sec)=%.7f\n", $3);Adv7=Adv7+1;AdvRcvTime7=$3;AdveRcv7=1;}
#error-reply=disc is wrong! purpose of error? error-
gwadv=processing=idle and disc=0 in this case
#-----Multicasting RREQ_I-----
if (( $1=="f") && ( $5=="7" )&&( $19=="RTR" ) && (
$35=="AODV" )&&( $37=="48")&&($31=="8388610.255")&&($3<300))
#sending RREQ_I {if(($61=="RREQ_I")&&(rr7==1)&&(Idle7==

```

```

1)&&(DSent7==0)){printf("Node 7 Multicast RREQ_I after its
error at time(Sec)= %.7f\n", $3);ReqTime7=$3;rq7=1;
if(ReqTime7>errorT7) {printf("7's Idle Time==>(ReqT-
errorT)(Sec)=%.3f\n", ReqTime7-errorT7);
TotalIdleTime7=TotalIdleTime7+(ReqTime7-errorT7);
IdleC7=IdleC7+1;Idle7=0;}}
#-----Receiving RREP_I-----
if (( $1 == "r" ) && ( $5 == "7" )&&( $19 == "RTR" ) && (
$35=="AODV" )&&($33 == "8388610.255")&&( $37 == "44")&&($3<300))
{if((rr7==1)&&(rq7 == 1)&&(DSent7==0))
{printf("Node-7 Got RREP_I message after error at time(ms)=
%.7f\n", $3)ReplyTime7=$3;if(ReplyTime7>ReqTime7) {printf("7's
Disc-Time==>(reply-reqT)(Sec)=%.3f\n\n", ReplyTime7-ReqTime7);
TDiscTime7=TDiscTime7+(ReplyTime7-ReqTime7);
DiscC7=DiscC7+1;rq7=0;rr7=0;Idle7=1;}}
if (( $1 == "r")&&( $5 == "7" )&&( $19 == "RTR" ) && ( $35=="AODV"
)&&($61 == "RREP_I")&&( $37 == "44")&&($3<300))
{if((rq7 == 1)&&(AdveRcv7 == 0)&&(rr7 == 1)&&($33 !=
"8388610.255"))
{printf("Node-7 Got RREP_I of another node after its request, at
time(ms)= %.7f\n", $3);
AnotheRep7=1;AnotheRep7Time=$3;}}
END {printf("7's Radial Search Latency
Time(Sec).....=%.3f\n", (radialreplytime7-RadialRequ7));
printf("7's Difference b/n first data reached the a gateway and
Radial search start time(Sec)=%.3f\n", (RadialGWDataRecvTime7-
RadialRequ7));printf("7's Sum of all
Discoverytimes(Sec).....=%.3f\n", TDiscTime7);
printf("7's Number of Discoveries.....=%d\n", DiscC7);
printf("7's Sum of all Idle times(Sec).req-
error.....=%.3f\n", TotalIdleTime7);
printf("7's Number of Idle Times after error ....=%d\n\n", IdleC7);
if(DiscC7>0) AvgDiscT7=TDiscTime7/DiscC7;else AvgDiscT7=0;
if (DiscC7>0) printf("7's Average Discovery Time(Sec)..Avg(rep-
req).....=%.4f\n", AvgDiscT7);
if(IdleC7>0) AvgIdleT7=TotalIdleTime7/IdleC7;else AvgIdle7=0;
if (IdleC7>0) printf("7's Avg Idle time (Sec)..Req-
error.....=%.4f\n", AvgIdleT7);
printf("\n7's Total Disconnection Time
(Sec).....=%.4f\n", (TDiscTime7+TotalIdleTime7));
printf("7's Time for discovery out of total simulation
time(%).....=%.4f\n\n", (TDiscTime7*100)/300
);printf("7's Idle Time out of the total simulation time(%)...req-
error.....=%.4f\n\n", (TotalIdleTime7*100/300));
printf("7's Disconnection Time out of the total simulation time in
(%).....=%.4f\n\n", ((TotalIdleTime7+TDiscTime7)*100/300)
);}

```