MULTICAST ROUTING IN FIXED INFRASTRUCTURE AND MOBILE AD HOC WIRELESS NETWORKS WITH A MULTICAST GATEWAY

by

Wei Ding

A thesis submitted to the Faculty of Graduate Studies in partial fulfillment of the requirements of the degree of

Master of Science
Information and Systems Science

School of Computer Science
Carleton University
Ottawa, Canada

July 2002

The undersigned recommend to the Faculty of Graduate Studies and Research
the acceptance of the thesis

MULTICAST ROUTING IN FIXED INFRASTRUCTURE AND MOBILE AD HOC WIRELESS NETWORKS WITH A MULTICAST GATEWAY

Submitted by **Wei Ding**, M.Sc.
In partial fulfillment of the requirements for the degree of M.Sc. in
Information & System Science

____
Thesis Supervisor

____
Director, School of Computer Science

Carleton University

July 2002

ABSTRACT

Multicast is the transmission of datagrams to a group of zero or more hosts identified by a single destination group address. It provides a simple yet robust and efficient communication mechanism. Various categories of multicast routing protocols have been developed to perform the fixed wireline network multicasting and the wireless mobile ad hoc network multicasting separately. But less work has been done for the multicast routing between these two networks except for some work done with mobile IP for multicasting in fixed infrastructure cellular network, which consists of stationary base stations and one hop mobile endpoints. In this thesis, a multicast gateway (MGW) is designed and implemented to solve the challenge of multicast routing in the mixed network that consists of a fixed subnet and a wireless mobile multi-hop ad hoc subnet. Simulations were conducted on the network simulator *ns-2* to evaluate the performance of data delivery ratio and control overhead of protocol combinations of four fixed multicast protocols (in PIM-Sparse Mode or in PIM-Dense Mode) and two mobile ad hoc multicast protocols, i.e., Multicast Ad-hoc On-demand Distance Vector (MAODV) and On-Demand Multicast Routing Protocol (ODMRP), with the functionality of MGW by varying the sender and receiver numbers as well as scaling the subnet size. Our pioneer work of MGW has fulfilled the multicast data transmission for this mixed network. It also provides a model for the future study in this area.

ACKNOWLEDGEMENT

I would like to extend my thanks to my supervisor, Professor Thomas Kunz, for his guidance and direction for this thesis. His sharp mind and rigorous attitude towards the research have made my thesis research a beneficial as well as a pleasant experience.

I would like to thank all the faculty and staff at the School of Computer Science and the Department of Systems and Computer Engineering, as well as the School of Mathematics and Statistics in Carleton University, for everything I have learned here.

Finally, I give my special thanks to my husband Jiahong, my son Jingru, and my parents for their endless love and great support in my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

ACK	Acknowledgement
ADMR	Adaptive Demand-Driven Multicast Routing protocol
ARP	Address Resolution Protocol
CBR	Constant Bit Rate
CBT	Core-Based Tree
CM	Centralized Multicast
CTS	Clear-to-send
DARPA	The Defense Advanced Research Projects Agency
DCF	Distributed Coordination Function
DVMRP	Distance Vector Multicast Routing Protocol
DSSS	Direct-Sequence Spread-Spectrum
FG_FLAG	Forwarding Group Flag
ID	Identification
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IFq	Interface Queue
IGMP	Internet Group Membership Protocol
IP	Internet Protocol
IPv6	Internet Protocol Version 6
LL	Link Layer
MAC	Medium Access Control

MACT	Multicast Route Activation
MGW	Multicast Gateway
MANET	Mobile Ad-hoc Network
MAODV	Multicast Ad-hoc On-Demand Distance Vector Protocol
MoM	Mobile Multicast
NetIf	Network Interface
NS-2	Network Simulator (Version 2)
ODMRP	On-Demand Multicast Routing Protocol
OTcl	Object Tool Command Language
PIM	Protocol-Independent Multicasting
PIM-DM	Protocol-Independent Multicasting - Dense Mode
PIM-SM	Protocol-Independent Multicasting - Sparse Mode
RP	Rendezvous Point
RPF	Reverse Path Forwarding
RPT	Rendezvous Point Based Tree
RREP	Route Reply
RREQ	Route Request
RTS	Request-to-send
SPT	Shortest Path Tree
ST	Shared Tree
ST-WIM	Shared Tree Wireless Network Multicast
TTL	Time-To-Live

Chapter 1

1 Introduction

The idea of ad hoc networks of mobile nodes dates back to the DARPA packet radio network [JT87]. With the development of laptop computers, which have reduced weight, great power and long battery life, increased need has emerged for the communication among laptop computers, as well as among laptop computers and a fixed network.

Multicast is the transmission of datagrams to a group of zero or more hosts identified by a single destination group address. A multicast datagram is typically delivered to all members of its destination host group with the same reliability as regular unicast datagrams [Kunz02]. The key problem in multicasting is to enable efficient routing of packets from a sender to multiple receivers. The use of multicasting within a network has many benefits. Multicasting reduces the communication costs for applications that send the same data to multiple recipients. Instead of sending via multiple unicasts, multicasting minimizes the link bandwidth consumption, sender and router processing, and delivery delay [Paul98]. In addition, multicasting provides a simple yet robust communication mechanism whereby a receiver's individual address is unknown or changeable transparently to the source.

Several algorithms are available today for fixed network multicast routing [WZ01]. *Flooding* is the simplest algorithm that guarantees all group members that are reachable will receive the data. The principle is similar to broadcasting. No routing table is maintained. Upon receipt of the data, the router forwards it to all interfaces with the exception of the interface on which the data was received. Flooding can place a high internal load on a network, with many systems receiving data that is not addressed to them. An alternative algorithm uses *spanning trees* that reach all systems in the network; that is, it spans the entire network. However, since it is implemented without specific knowledge about group membership, broadcasting is used along the spanning tree. The next category of algorithms considers explicitly group membership during the setup of a distribution tree, called *multicast tree*. And three basic techniques are implemented in current protocols to address routing in this category:

- Source-based routing
- Steiner trees
- Trees (shared trees) with rendezvous points (RP)

The first and the third techniques are of practical importance. Source-based routing builds a multicast tree for each sender, while the third technique builds only one multicast tree (rooted at the RP) for each multicast group. A source-based scheme consists of broadcasting the packet from the source to all destinations along the source tree in a manner that avoids loops. This is accomplished by using “Reverse Path Forwarding” or RPF. In RPF, a router forwards a broadcast packet on its remaining interfaces if and only if the packet is received on an interface that is on the shortest path from the router to the source. Thus, only those packets are forwarded that arrive on the reverse shortest path from the router to the sender. Example protocols are Distance Vector Multicast Routing Protocol (DVMRP) [WPD88], Protocol-Independent Multicasting (PIM)-dense mode

(PIM-DM) [DEF99]. Source-based techniques are promising for smaller networks with high group density.

In the shared tree multicast scheme, all receivers join the multicast group by explicitly sending a JOIN message towards the RP. Senders send data to the RP, and the RP uses a single unidirectional-shared tree to distribute the data to the receivers. Algorithms with rendezvous points are attractive because they completely eliminate flooding, and are preferable for widely distributed groups where group members are not located in each subnetwork, that is, in the case of low group density. The problem is that rendezvous points (RP) constitute points of concentration for a group's traffic and represent a single point of failure. PIM-sparse mode (PIM-SM) [DEF98] and Core-based trees (CBT) [Bal97, Ball97] are examples that use this RP technique.

A mobile ad-hoc network (MANET) is a network that consists of a collection of dynamic nodes with sometimes rapidly changing multi-hop topologies that are composed of relatively bandwidth-constrained wireless links. For MANET, there are also three basic categories of multicast algorithms [Kunz02]. The first approach is to simply flood the network. And flooding a network acts like a chain reaction that can result in exponential growth. The second one, the pro-active approach, pre-computes paths to all possible destinations and stores this information in routing tables. To maintain an up-to-date database, routing information is periodically distributed throughout the network. Protocols like wireless extensions to DVMRP [CGZ99], Shared Tree Wireless Network Multicast (ST-WIM) (adapting PIM-SM to MANET)[CGZ97], and a modified version of the CBT multicast algorithm [CG97] as well as an Adaptive Shared Tree multicast [CGZ98] are based on this second approach. The third approach is to create paths to other hosts on demand. The idea is based on a query-response mechanism or reactive multicast. In the query phase, a node explores the environment. Once the query reaches the destination, the response phase is entered and the path is established. Multicast Ad-hoc On-Demand Distance Vector (MAODV) [RP99] and On-Demand Multicast Routing Protocol (ODMRP) [GL00] are two current popular on-demand multicast routing protocols. Recently, Adaptive Demand-Driven Multicast Routing protocol (ADMR) [JJ01], a new on-demand ad hoc network multicast routing protocol has been presented. ADMR attempts to reduce as much as possible any non-on-demand components within it.

1.1 Motivation

Maintaining group membership information and building optimal multicast trees is challenging in wired networks, and is particularly more challenging for MANET. For these two kinds of individual network domains, a number of multicast routing protocols have been implemented, validated and evaluated on different network simulators and even deployed onto real networks, including the Internet. However, less work has been done for the multicast routing between the fixed and the ad hoc multi-hop networks. Some work has been done for the multicasting in fixed infrastructure cellular network, which consists of stationary base stations and one hop mobile endpoints. Each base station is assigned a geographic area, so called a cell, and is responsible for connecting mobile endpoints to the wired network via one hop. Mobile IP [Per96, Per97] has been used to manage these kind of multicasting. Remote subscription multicast (foreign agent-based multicast) [SSK00], bi-directional tunneled multicast (home agent-based multicast) [BHW97], and mobile multicast (MoM) [BHW97] are instances of the Mobile IP applications. In this thesis, we extend the cellular world to build a multicast network that has a fixed domain and an ad hoc access part, which provides mobility with less infrastructure support. Figure 1.1 shows the generalized multi-hop wireless network with a fixed access point, the multicast gateway, to and from the wireline fixed network.

Traditional wireline multicast protocols like Protocol Independent Multicast – Sparse Mode (PIM-SM) [DEF96], Core Based Tree (CBT) [BFC93], and Distance Vector Multicast Routing Protocol (DVMRP) [WPD88] do not work well in a MANET. In a wireless environment, since the nodes are mobile, the frequent tree reorganization from these traditional multicast protocols can cause significant signaling overhead and frequent loss of datagrams. Tree reorganization is more frequent in MANET versus conventional static networks because the multicast protocols will need to deal with network topology changes as well as membership changes. Research [CGZ97, CG97, CGZ98, CGZ99] also shows that traditional wire-line multicast protocols which have been modified for the wireless environment are not suitable in terms of scalability and performance with increased mobility.

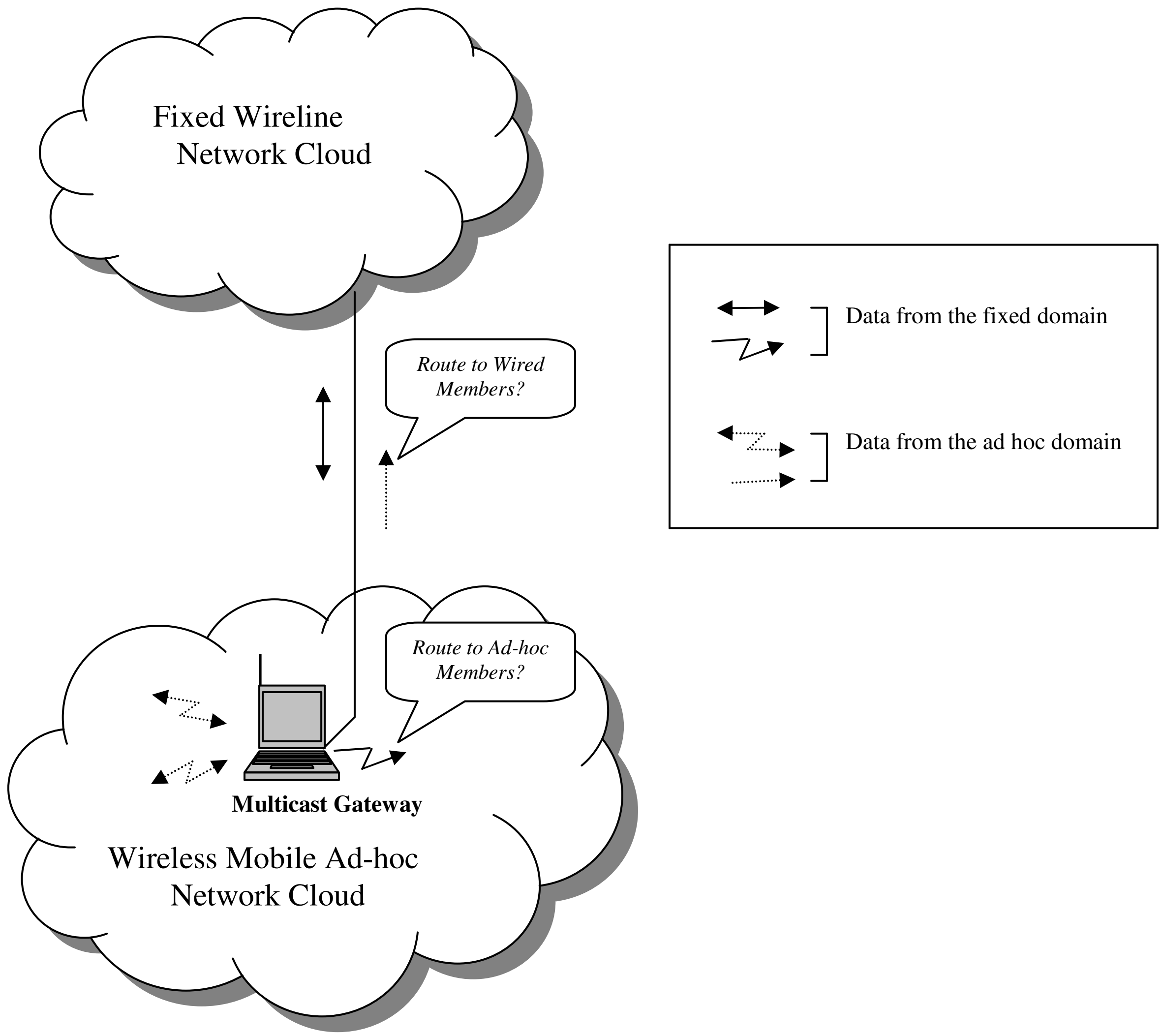

Figure 1.1: Multicasting Between the Fixed and the Wireless Mobile Ad hoc Networks with a Multicast Gateway as an Access Point

On the other hand, doing multicast routing on the fixed networks by modifying multicast protocols originally designed for mobile ad hoc networks might also raise some problems. By using table-driven wireless multicast protocols, periodic route updates need to be performed. This can generate great signaling traffic, which might be unnecessary and should be avoided in the fixed network that usually has little or no topology changes. All these control packets could introduce traffic overhead and inhibit data packet delivery for multicasting in the fixed network. This is also true for the on-demand protocols: MAODV broadcasts Group Hello periodically, and ODMRP periodically rebuilds the whole mesh by each source through broadcasting.

In this thesis, we propose a Wireline-Gateway-MANET model to handle multicasting between the wired network and a mobile wireless multi-hop ad hoc network. Basically, the idea is to use a certain fixed network multicast protocol for the wire-line domain multicasting, and use some existing on-demand MANET multicast protocol for the MANET domain. Between them, a special gateway is designated and implemented to fulfill the inter-domain multicast communication. Through the functionality of MGW, performance of different protocol combinations is explored for the mixed network under various network scenarios.

1.2 Research Contributions

This thesis provides the following contributions:

- Designed and implemented a multicast gateway (MGW) that transmits data packets between the fixed/wireline subnet and the MANET. The MGW uses different multicast protocol combinations for each side.
- A comprehensive evaluation for protocol combinations with MGW in terms of a set of performance metrics under various network scenarios.
- Insight into the future application and improvement in the mixed network multicasting.

1.3 Organization of this Thesis

This thesis is organized as follows: Chapter 2 surveys multicast routings on the Internet and MANET. They are the theoretical and practical bases for our mixed network multicasting. For better understanding of the design and implementation of the MGW, we also introduce the detailed node component configurations for a wireline multicast node as well as a wireless mobile multicast node in *ns-2*. Chapter 3 contains the analysis and design of the MGW in *ns-2*. Chapter 4 first shows eight protocol combinations that work with the MGW, it then explains the simulation environment with performance metrics and network parameters used in this study. Chapter 5 presents a discussion of the simulation results. Chapter 6 concludes this study and proposes future work directions.

Chapter 2

2 Background

2.1 Multicast Routing on the Internet

This section is devoted to some existing multicast routing protocols on the Internet. They belong to the categories of source-based routing and RP based shared tree routing. They all use Internet Group Membership Protocol (IGMP) as the basis for group management. The requirement in network areas with a shared medium (e.g., Ethernet segment) is that a designated router is informed of group membership [WZ01].

2.1.1 PIM-Sparse Mode

PIM-Sparse mode [DEF96, DEF98] is more efficient for geographically distributed group members in the network with low density. Two basic premises exist for this mode [WZ01]: Group membership is based on explicit join operations; and RPs are provided.

The explicit group join is designed to reduce the production of multicast data units by a sender at the beginning. Data is sent from the source to the RP. If members exist in a domain, this domain needs to register through explicit join to the group in order to get routed data from the RP. In real applications, more than one RP's are distributed across the network, although each group utilizes only one RP. When a receiver intends to join a group, it uses IGMP to signal in the subnet so that the designated router is aware of its group membership. After obtaining the RP information for this group, the router periodically sends an explicit join data unit (PIM join) to the RP.

The senders use tunneling to establish unicast paths to the RP, which is the root for the multicast tree (Figure 2.1). This tree is not necessarily optimal for the individual combinations of senders in a group [WZ01]. When a sender starts transmitting data to a group, its designated router forwards the data encapsulated in a unicast register data unit to the RP. There the data is decapsulated and routed as multicast data along the multicast tree rooted at the RP. The option of switching to a source-specific tree is provided, which relieves the overhead for encapsulation and decapsulation for high data rate streams, e.g., a significant number of data packets have been received in a certain period from a particular source. Only the RP or routers with local group members can initiate the transition from the shared tree to the source specific tree. As in Figure 2.2, PIM router 2 can initiate the transition from shared tree to specific tree for sender S1, while keep the shared tree for sender S2. It sends a pruning message to the RP, which generates an entry in the form (S, G) and sets appropriate bits indicating that the RP is part of the shared tree but not of the specific tree. The new route from sender S1 to the receiver includes PIM router 2 without any involvement of the RP. PIM router 2 periodically sends PIM join to the sender S1. After receiving data from the sender S1, PIM router 2 locally sets a bit to indicate specific tree transmission for this sender.

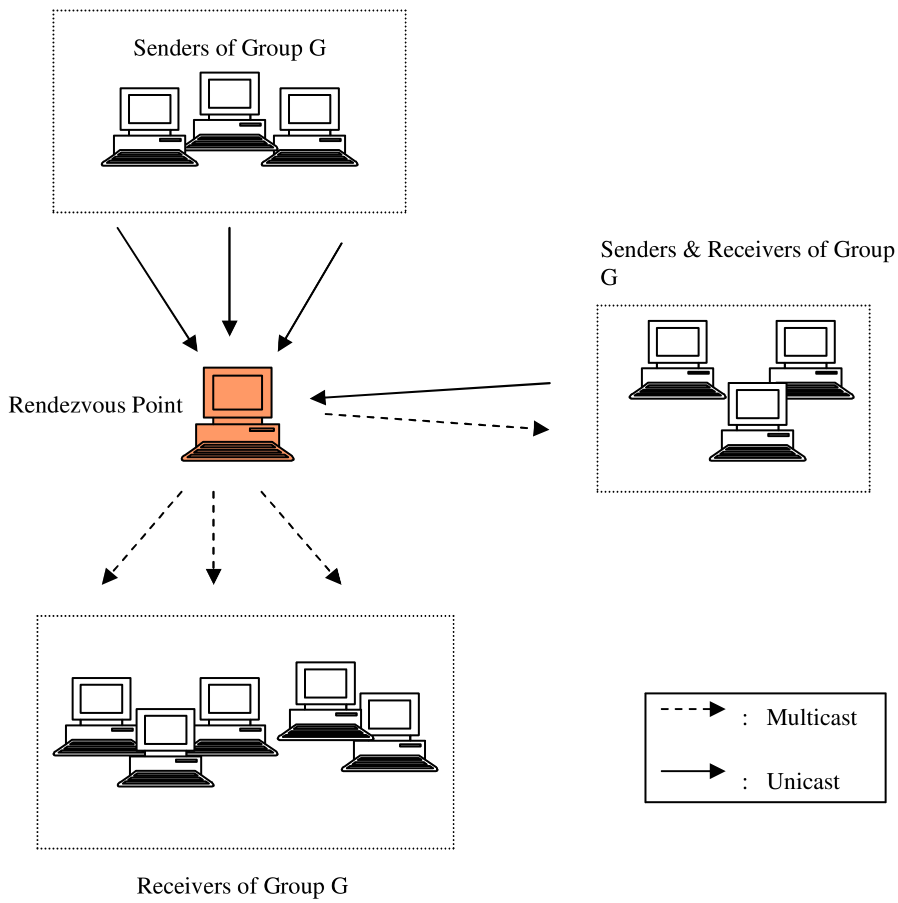

Figure 2.1: Shared Tree Multicast with a Rendezvous Point

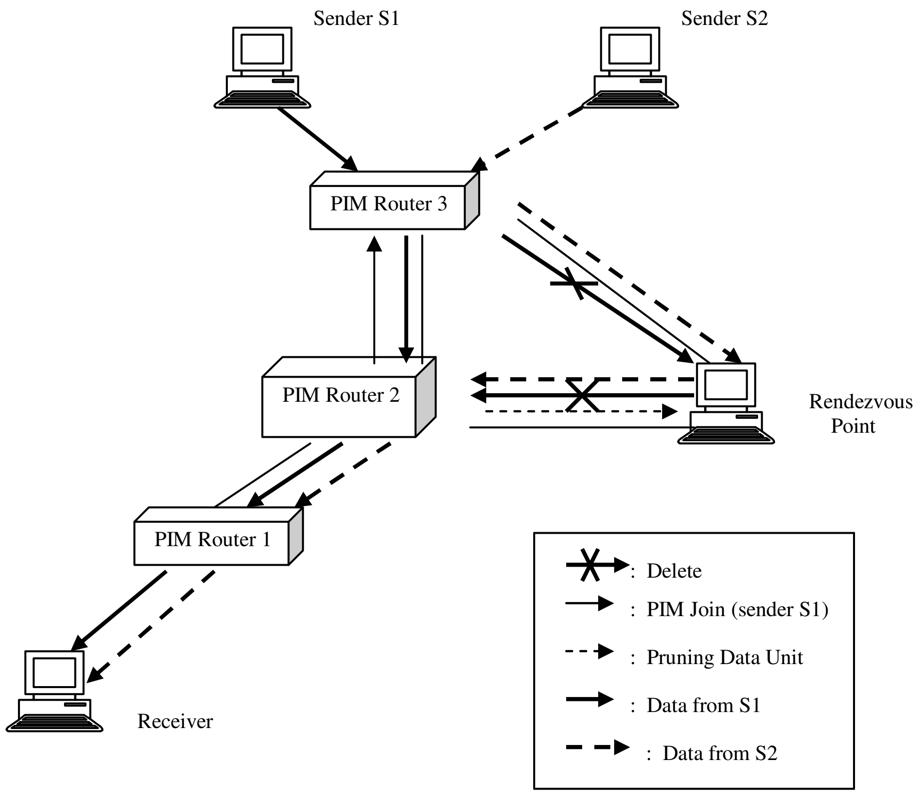

Figure 2.2: Source Specific Tree for Sender S1 and Shared Tree for Sender S2 in PIM-SM

Three situations exist for a router to keep a multicast routing entry when a receiver wants to join a group: No entry exists for group G; Entry for group G exists with unspecified source; and entry for group G exists with specified source S [WZ01]. If no multicast entry for group G exists yet in the router and a join data unit is received, a wildcard route entry (*, G) is created for the group. The wildcard stands for any source. After the router invokes a hash function to determine a RP, it sends a join data unit to this RP. If the group exists in the router in the form (*, G), then the data is delivered along the shared tree. And there is no need for the router to send another join to the RP for group G. If a special tree exists for sender S, this fact is noted in the router on the specific tree as (S, G), and no join data unit is sent to the RP. Data are forwarded on the source specific tree. A multicast routing entry in the router is not deleted as long as there is a group member and the router is required to forward data units to receivers in other subnet, i.e., dependent routers exist.

2.1.2 PIM-Dense Mode

PIM-dense mode [DEF99] is designed for multicast communication in small networks and high group density. No RP is used. The protocol builds a source specific multicast tree for each sender as soon as it starts sending data. Multicast routing entries in the form (S, G) are kept in related routers under the mechanism of timeout. At startup, it assumes that all subnets wish to receive data. Therefore, flooding and pruning are used. Graft data units are sent to support immediate integration of new group members into the multicast tree. PIM-DM routers periodically send Hello data units in order to become acquainted with their neighbours in the network.

Due to flooding, data can end up being sent unnecessarily to network areas in which no group members are located. However, this is considered acceptable because it is assumed that the density of a group will be very high, the distance between members are short, and consequently the additional overhead will be low [WZ01].

2.2 Multicast Routing on the MANET

Multicast Ad-hoc On-Demand Distance Vector (MAODV) routing protocol [RP99] and On-Demand Multicast Routing Protocol (ODMRP) [GL00] are two examples of MANET inspired multicast protocols presented below. MAODV is a tree based on demand multicast protocol. ODMRP is a mesh based on demand multicast protocol. In this thesis, these two protocols are chosen for the ad hoc mobile subnet multicast routing.

2.2.1 Multicast Ad-hoc On-Demand Distance Vector (MAODV)

The MAODV routing protocol [RP99¸ Kunz02] discovers multicast routes on demand using a broadcast route-discovery mechanism. A mobile node originates a Route Request (RREQ) message when it wishes to join a multicast group (sets a join flag in RREQ), or when it has data to send to a multicast group without the knowledge of a route to that group (Figure 2.3 (a)). Only a member of the desired multicast tree or a group member on the tree may respond to a join RREQ. If the RREQ is not a join request, any node with a fresh enough route (based on a group sequence number) to the multicast group may respond. If an intermediate node receives a join RREQ for a multicast group of which it is not a member, or if it receives a RREQ and it does not have a route to that group, it rebroadcasts the RREQ to its neighbours.

As the RREQ is broadcast across the network, nodes set up pointers to establish the reverse route in their route tables. A node receiving a RREQ first updates its route table to record the sequence number and the next hop information for the source node. This reverse route entry may later be used to relay a response back to the source. For join

RREQs, an additional entry is added to the multicast route table. This entry is not activated unless the route is selected to be part of the multicast tree.

If a node receives a join RREQ for a multicast group, it may reply if it is a member for the multicast tree and its recorded sequence number for the multicast group is at least as great as that contained in the RREQ (Figure 2.3 (b)). The responding node updates its route and multicast route tables by placing the requesting node's next hop information in the tables, and then unicasts a Route Reply (RREP) back to the source node. As nodes along the path to the source node receive the RREP, they add both a route table entry and a multicast route table entry for the node from which they received the RREP, thereby creating the forwarding path.

When a source node broadcasts a RREQ for a multicast group, it often receives more than one RREP. The source node keeps the received route with the greatest sequence number and the shortest hop count to the nearest member of the multicast tree for a specified period of time, and disregards other routes. At the end of this period, it enables the selected next hop in its multicast route table, and unicasts an activation message (MACT) to this selected next hop. The next hop, on receiving MACT, enables the entry for the source node in its multicast route table. If this node is a member of the multicast tree, it does not propagate MACT any further. However, if this node is not a member of the multicast tree, it will have received one or more RREPs from its neighbours. It keeps the best next hop for its route to the multicast group, unicasts a MACT message to that next hop, and enables the corresponding entry in its multicast route table. This process continues until the node (member of the tree) that originated the RREP is reached. Nodes that generated or forwarded RREPs delete their entries for the requesting node if they do not receive a MACT in a certain time. Figure 2.3 (c) illustrates the procedure of multicast tree branch addition through MACT. The MACT message ensures that the multicast tree does not have multiple paths to any tree node. Nodes only forward data packets along activated routes in their multicast route tables.

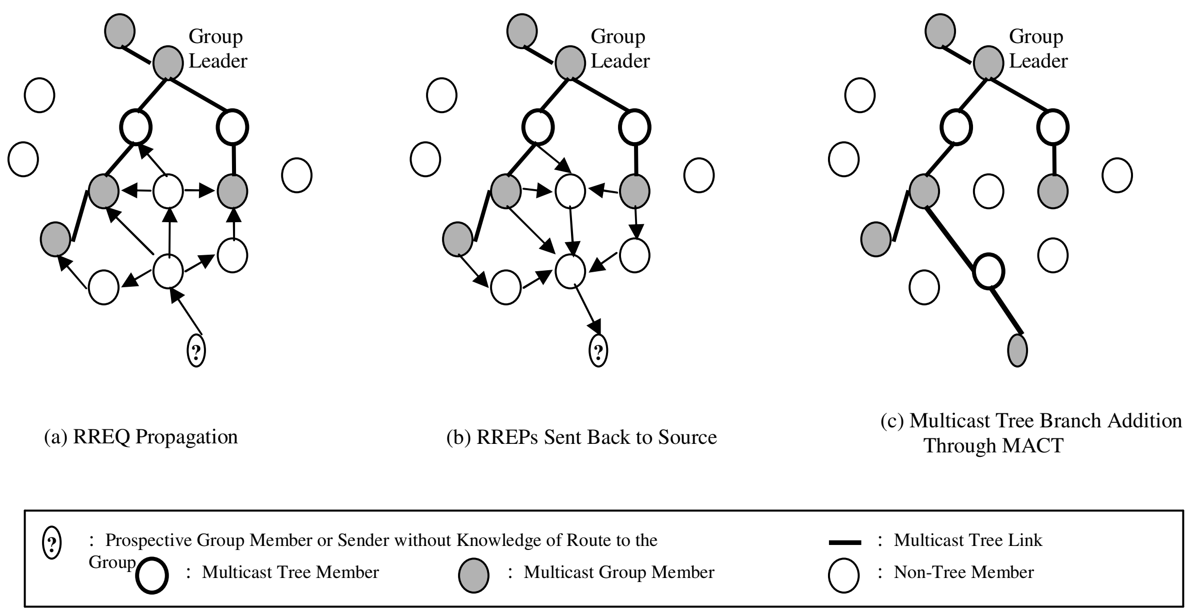

Figure 2.3: MAODV Multicast Join Operation [RP99]

The first member of the multicast group becomes the leader for that group. The multicast group leader is responsible for maintaining the multicast group sequence number and broadcasting this number to the multicast group. This is done through a Group Hello message. The Group Hello contains extensions that indicate the multicast group IP address and sequence numbers (incremented every Group Hello) of all multicast group for which the node is the group leader. Nodes use the Group Hello information to update their request table.

Since MAODV keeps hard state in its routing table, the protocol has to actively track and react to changes in this tree. If a member terminates its membership with the group, the multicast tree requires pruning. Links in the tree are monitored to detect link breakages. When a link breakage is detected, the node that is further from the multicast group leader (downstream of the break) is responsible for repairing the broken link. If the tree cannot be reconnected, a new leader for the disconnected downstream node is chosen as follows. If the node that initiated the route rebuilding is a multicast group member, it becomes the new multicast group leader. On the other hand, if it was not a group member and has only one next hop for the tree, it prunes itself from the tree by sending its next hop a prune message. This continues until a group member is reached. If these two partitions reconnect, a node eventually receives a Group Hello for the multicast group that contains group leader information that differs from the information it already has. If this node is a member of the multicast group, and if it is a member of the partition whose group leader has the lower IP address, it can initiate reconnection of the multicast tree.

2.2.2 On-Demand Multicast Routing Protocol (ODMRP)

ODMRP [BL00, GLS00, Kunz02] uses a flooding-based mesh and a forwarding group concept. A soft state approach is taken to maintain multicast group members without explicit group membership messages. Multicast routes and group membership are established and updated on demand. When a multicast source has data packets to send to a group without any knowledge of the route, it broadcasts a Join-Query control message (Figure 2.4). The Join-Query message is also periodically broadcast to update the membership and route information.

When an intermediate node receives the Join-Query, it checks and stores the source ID and the sequence number in its message cache to detect any duplicates. The routing table is updated with the appropriate node ID from which the message was received for the reverse path back to the source node. If the Join-Query is not a duplicate and the Time-To-Live (TTL) is greater than zero, it is then rebroadcast.

When the Join-Query reaches a group member, it generates and broadcast a Join-Reply as shown in Figure 2.4. When a node receives a Join-Reply, it checks if the next hop ID of one of the entries matches its own ID. If so, it knows that it is on the path to the source and thus is part of the forwarding group. It then sets the Forwarding Group Flag (FG_FLAG), and broadcasts its own Join Table built upon matched entries. The next hop node ID field is filled by extracting information from its routing table. In that sense, each forwarding group member propagates the Join-Reply until it reaches the multicast source via the selected shortest path. This process constructs the routes from source to receivers by building a mesh of forwarding group nodes.

After this, senders can multicast data packets to receivers via the forwarding group on the selected routes. When a node receives a multicast data packet, it forwards it only when it is not a duplicate and the setting of the FG_FLAG for this group has not expired. This reduces the traffic overhead and preventing sending packets through stale routes.

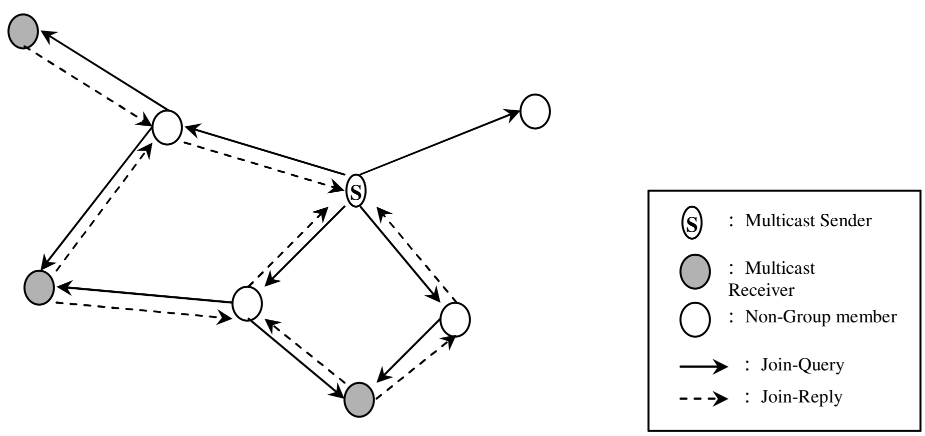

Figure 2.4: ODMRP Membership Setup and Maintenance

In ODMRP, the soft state mechanism is applied. A sender stops broadcasting Join-Query if it has no more data to send to the multicast group. When a receiver wants to leave a group, it simply does not send Join-Replies for that group. Nodes in the forwarding group are denoted to be non-forwarding nodes if they are not refreshed (by receiving Join Replies) before they timeout. The timer values for Join-Query refresh interval and forwarding group timeout interval can have an impact on ODMRP performance [GLS00]. One should choose these values after concerning the network environment. The forwarding group timeout value must be larger (e.g., 3 to 5 times) than the value of the Join-Query refresh interval.

2.3 Wireline Multicast Node Configuration in *ns-2*

Figure 2.5 shows the internal structure of a typical fixed multicast node in the network simulator *ns-2* [FV01]. This node has some physical links to be connected with some other fixed nodes. The node entry switches the arriving packet to either the *classifier_* or the *multicastclassifier_* depending on whether the packet destination is a specific node address (unicast) or is a multicast group address. According to the source and group address value, the multicast packets are sent to the corresponding replicators. Each replicator replicates and forwards the packets towards the links as well as the receiving agent if the node is a group member. In case of RP based shared tree multicasting, when the node is a source for multicast data packets, an encapsulator object will be created and targeted by the source agent so that any data to the multicast group will be tunneled only to the decapsulator object inside the RP node before it is distributed to other group members.

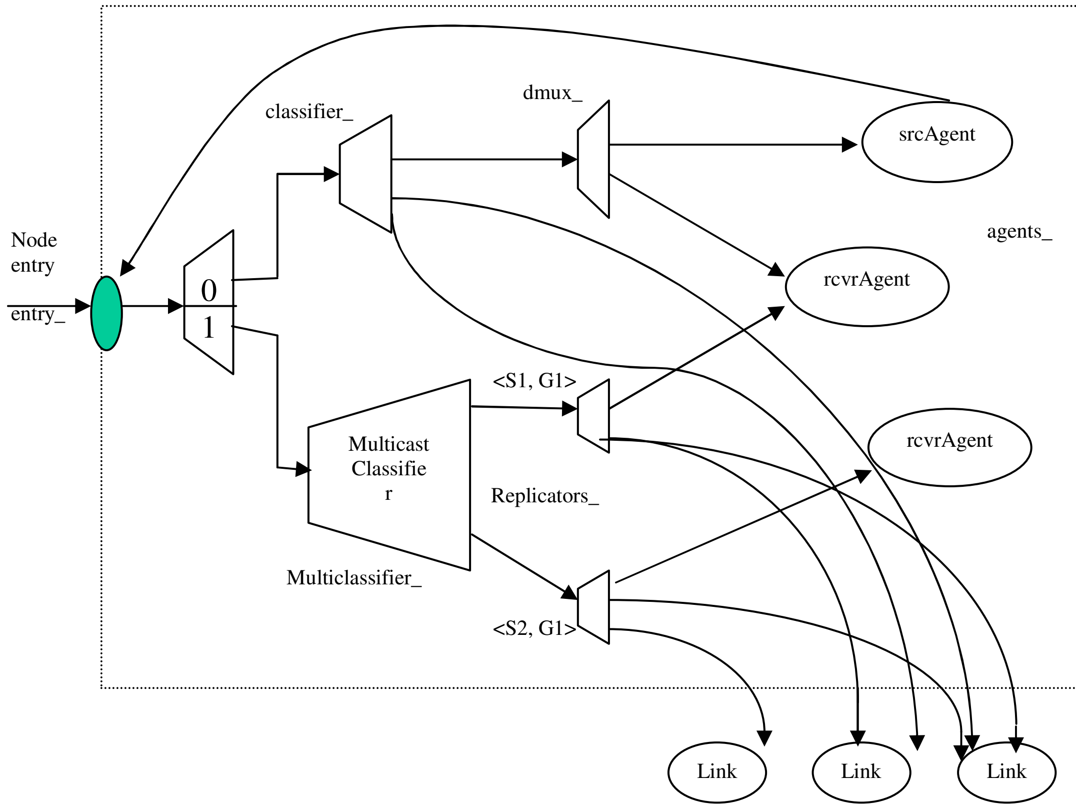

Figure 2.5: Internal Structure of a Multicast Fixed Node in *ns-2*

2.4 Wireless Mobile Multicast Node Configuration in *ns-2*

Figure 2.6 is schematic of a mobile node in the network simulator *ns-2*. The mobile node has a MAODV or ODMRP routing agent configured inside. It consists of the network stack of a link layer (LL), an ARP module, an interface queue (IFq), a MAC layer (MAC), and a network interface (NetIf). All of these get connected together down to the channel. These network components are created and combined in OTcl [FV01]. The routing agent receives both control packets for routing and data packets.

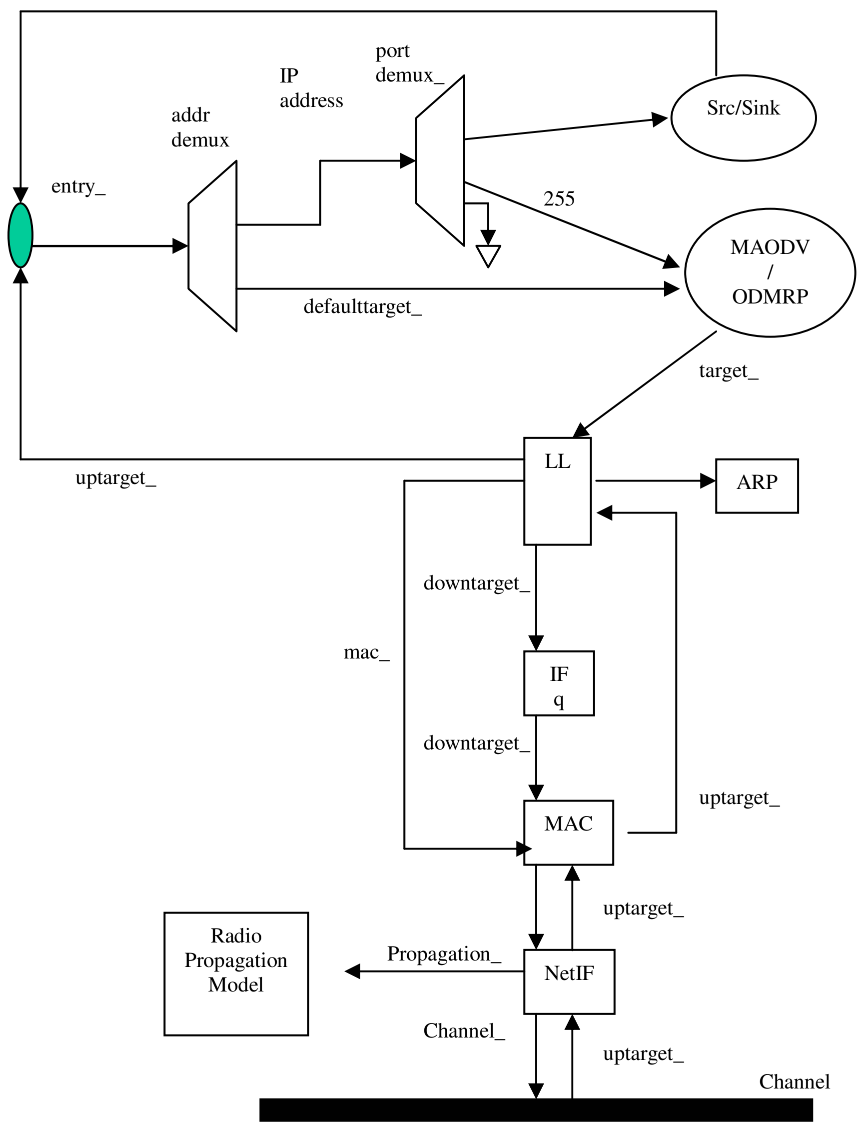

Figure 2.6: Schematic of a Multicast Wireless Mobile Node in *ns-2*

The link layer (LL) simulates the data link protocols. Many protocols can be implemented within this layer such as packet fragmentation, reassembly, and reliable link protocol. Another important function of the link layer is setting the MAC destination address in the MAC header of the packet. This function has been implemented as two separate issues: finding the next-hop-node's IP address (routing) and resolving this IP address into the correct MAC address (ARP). For simplicity, the IP addresses are reused at the MAC layer. In Figure 2.6, the LL has an ARP module connected to it which resolves all IP to hardware (MAC) address conversions. Normally all outgoing packets are handed down to the LL by the routing agent. The LL hands down packets to the interface queue. For all incoming packets, the MAC layer hands up packets to the LL which is then handed off at the node *entry_* point.

The Address Resolution Protocol (ARP) module receives queries from LL. If the ARP has the hardware address for the destination, it writes it into the MAC header of the packet. Otherwise, it broadcasts an ARP query, and caches the packet temporarily. For each unknown destination, there is a buffer for a single packet. In case additional packets to the same destination are sent to ARP, the earlier buffered packet is dropped. Once the hardware address of a packet's next hop is known, the packet is inserted into the interface queue.

The class *PriQueue* is implemented as an interface priority queue which gives priority to routing protocol packets, inserting them at the head of the queue. It supports running a filter over all packets in the queue and removes those with a specified destination address.

For the MAC layer, the IEEE 802.11 distributed coordination function (DCF) MAC protocol has been implemented. MAC layer handles collision detection, fragmentation and acknowledgements. It uses a request-to-sent/clear-to-send/DATA/acknowledgement (RTS/CTS/DATA/ACK) four-way handshaking mechanism for all unicast packets and simply sends out DATA for all broadcast packets. To reserve the channel before unicast data transmission, the source first transmits a RTS frame. Following the reception of the RTS, the destination will transmit a CTS frame. Only after the source receives the CTS correctly, it can start sending its data packet. And a positive MAC ACK is transmitted by the destination to confirm the successive packet delivery.

The network Interface layer serves as a hardware interface that is used by mobile nodes to access the channel. This interface is subject to collisions and the radio propagation model receives packets transmitted by other node interfaces to the channel. The interface stamps each transmitted packet with the meta-data related to the transmitting interface like the transmission power, wavelength etc. This meta-data in the packet header is used by the propagation model in the receiving network interface to determine if the packet has minimum power to be received and/or captured and/or detected (carrier sense) by the receiving node.

The Radio Propagation Model uses Friss-space attenuation ($1/r^2$) at near distances and an approximation to Two Ray Ground ($1/r^4$) at far distances. The approximation assumes specular reflection off a flat ground plane. The wireless channel duplicates packets to all mobile nodes attached to the channel except the source itself. It is the receiver's responsibility to detect if it can receive the packet.

2.5 Summary and Correlation to Our Work

Sections 2.1 and 2.2 have gone through the protocol definitions of PIM-SM and PIM-DM for the fixed wireline multicasting, as well as the MAODV and ODMRP protocol definitions for the MANET multicasting. Sections 2.3 and 2.4 described the internal structure of a fixed and a mobile multicast node in *ns-2* respectively. As shown in Chapter 3, our MGW is designed based on these two types of node components under the circumstance of different pairs of multicast protocols utilized for each subnet. The protocol performance for the mixed networks will also be tested with various protocol combinations chosen from above (As shown in Chapter 4 & 5).

Chapter 3

3 Analysis and Design of the Multicast Gateway (MGW) for the Mixed Network Multicasting in *ns-2*

For the mixed network multicasting, the MGW is designed according to different multicast routing protocols used in each subnet. It is built with both the fixed multicast node components and the mobile multicast node components in *ns-2*, together with our modifications as shown below.

3.1 Combination of Fixed Side Rendezvous Point (RP) Based Multicast Protocols with MANET Multicast Protocols

In order to understand both wireline and MANET multicast protocols for multicast communication, the MGW node needs to have two types of network interfaces installed, one is the interface to the physical links, the other is the interface to the wireless channel. Figure 3.1 illustrates our detailed design of the MGW node in *ns-2*. No explicit sending agent exists in MGW for the MANET domain. In the beginning of the simulation, let the MGW join multicast groups on both sides. Although different group addresses are generated for each side according to the definitions in *ns-2*, the MGW treats them as one group. Only data packets will pass through the MGW from one domain to the other. And the MGW follows the appropriate protocols for each side multicast routing. Detailed processing is provided as follows when data packets arrive at the MGW node.

(I) Data packet forwarding from the fixed domain to the MANET domain

Let the MGW join the multicast group in the fixed domain with a receiver agent inserted to all the replicators. Meanwhile, let the routing agent also be inserted to the replicators so that whenever the MGW receives a data packet from the fixed side multicast group by its attached fixed side receiving agent, its routing agent can also get a copy of this packet. In Figure 3.1, the thick lines with arrows represent the insertions of routing agents to the replicators. After getting a copy of the fixed side multicast data, the routing agent checks that this is not a looped back data (this situation happens when RP based protocol is involved, i.e., some data packet flows from a mobile source -> MGW -> RP => MGW, since the MGW is also a group member in the fixed domain). Then it repacks the packet by changing some packet headers, e.g., resetting the data source to be the MGW node ID, changing the destination to be the multicast group in the MANET, setting the TTL value to be the initial value set in the ad hoc domain. After this, the repacked datagram is ready to be forwarded to the ad hoc domain through the routing or forwarding procedures in M*AODV* or *ODMRP*.

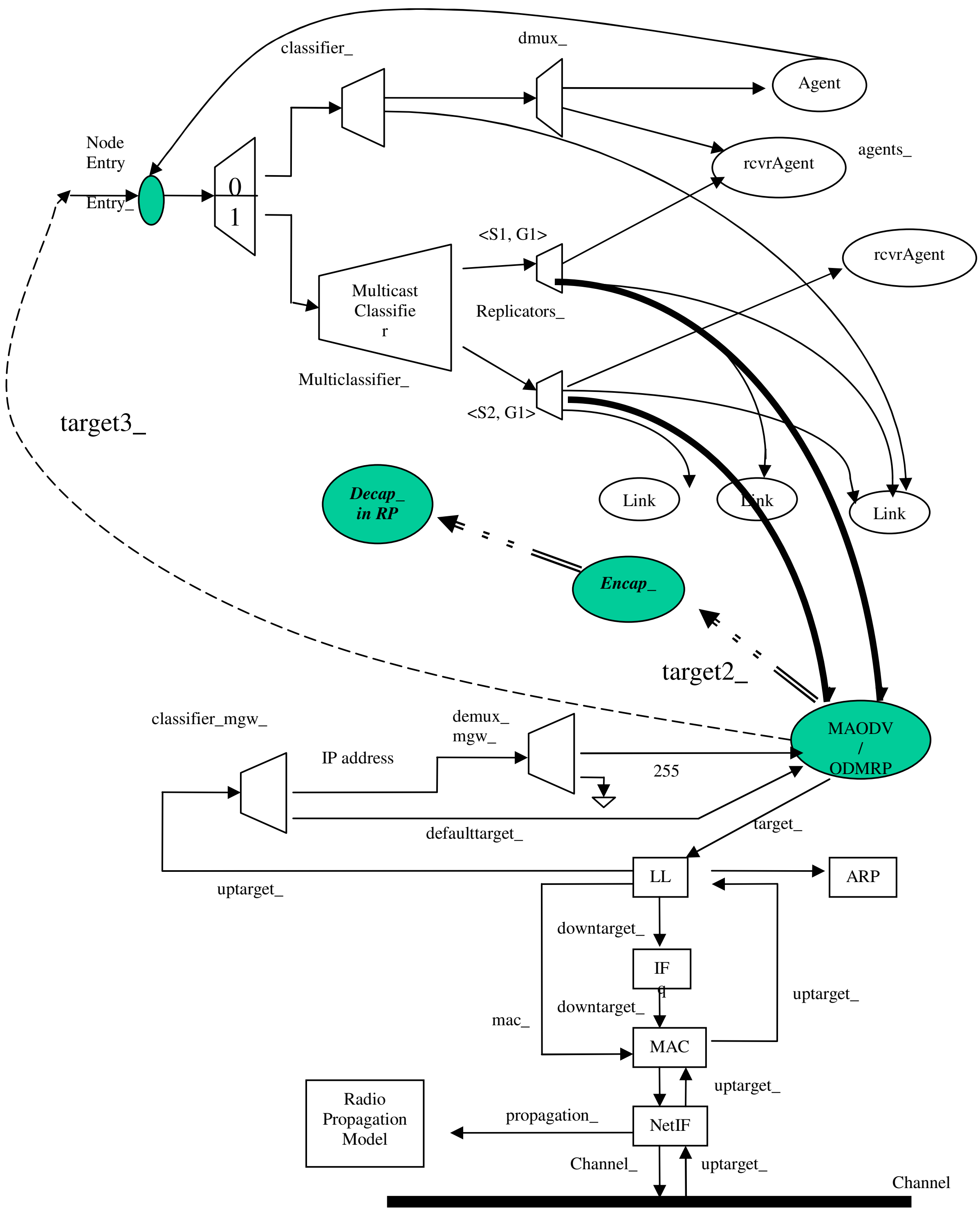

Figure 3.1: Schematic of a Multicast Gateway (MGW) for Wireline-MANET Multicast Routing

(II) *Data packet forwarding from the MANET to the fixed domain that runs RP based multicast routing protocol*

By receiving data packets from the MANET, and having to forward it to the fixed domain, the MGW indirectly acts as a sender in terms of the fixed domain. In order to fulfill this task under the RP based shared tree protocols, an encapsulator object is created in the MGW, and it is connected to the decapsulator object in the RP node of the fixed domain. In Figure 3.1, the double dashed lines represent this connection. The encapsulator object is targeted by the routing agent through the assignment of this object to an instance variable *target2_* in the routing agents' implementations. After the MGW receives (in its routing agent) a data packet from a source in the MANET with a non-zero TTL value, it creates a copy of this packet. Besides routine ad hoc side packet forwarding or routing in the MANET, the MGW's routing agent checks to make sure that this packet is not a duplicate, it then repacks the packet copy (e.g., changes the data copy's source ID to be its own and lets the data copy's destination to be the fixed side group address) and lets the encapsulator object receive it. Therefore, this data packet could be further distributed by the RP to the fixed side multicast group members.

3.2 Combination of Fixed Side Source Specific Tree Based Multicast Protocols with MANET Multicast Protocols

The design of the MGW for the combination of a source specific tree based fixed/wireline multicast protocol and the MANET protocols is similar to Section 3.1, except when the data packet is forwarded from the MANET to the fixed domain. As shown in Figure 3.1, the single dashed line targets the routing agent directly to the MGW's node entry (which is assigned as the *target3_* in routing agents' implementations). When a data packet arrives at the MGW in its routing agent, the routing agent generates a copy of the packet and forwards the original to the ad hoc domain by using the MANET routing protocol. If this data packet is not a duplicate, the MGW repacks the data copy and forwards it to the fixed side along the source specific multicast tree that is rooted at the MGW.

Chapter 4

4 Experimental Design

After the validation of the MGW implementation, the MGW model is used for protocol performance evaluations under different mixed network scenarios with different protocol combinations for each subnet. The performance evaluation is done on the network simulator *ns-2*.

4.1 Related Resources in *ns-2* and Protocol Combinations

For the wire-line network, *ns-2* provides the following multicast route computation strategies:

- Dense Mode (DM): a PIM-DM-like or DVMRP-like implementation.
- Centralized multicast (CM): a sparse mode implementation similar to PIM-SM. It implements two types of multicast trees, the default one is the "RPT" tree with rendezvous point (RP), the other is "SPT" tree that consists of a source specific shortest path tree. In the beginning of the simulation, the user can choose between these two tree types.
- Shared Tree Mode (ST): a simplified sparse mode implementation of the shared-tree (with RP) multicast protocol.

MAODV and ODMRP for MANET multicast routing in *ns-2* have been implemented and evaluated in our lab [Cheng01]. They are used in this thesis research.

Eight protocol combinations have been implemented with the MGW functionality. They are as follows: dm-maodv, dm-odmrp, st-maodv, st-odmrp, cmspt-maodv, cmspt-odmrp, cmrpt-maodv, cmrpt-odmrp. All the above eight combinations were investigated in this thesis for the protocol performance evaluations with the multicast gateway. For dense mode (DM), a PIM-DM-like implementation was used for fixed side multicasting. Since all the fixed multicast group members join the group in the very beginning and do not leave, and there is no network dynamics, we set one of DM's parameters, *pruneTimeout*, to be as large as the simulation duration in order to minimize control messages.

4.2 Simulation Environment

Similar to [Cheng01], the performance simulation environment we used is based on ns-2, a network simulator that was originally developed by the University of California at Berkeley for the VINT project [FV99]. In previous work [BMJ98], the MONARCH research group in CMU developed support for simulating multi-hop wireless networks complete with physical, data link and IEEE 802.11 MAC layer models in *ns-2*.

The environment consists of 51 nodes, split between wire-line nodes, mobile nodes, and one MGW node. Mobile nodes are roaming in an L meters * L meters flat space for a certain simulated time. The area size depends on the number of mobile nodes and is scaled to provide constant node density (excluding MGW). For fifty mobile nodes, L equals to 1000 meters, whereas for forty, thirty, twenty or ten mobile nodes, L equals 894 meters, 776 meters, 632 meters or 447 meters. The radio transmission range is 250 meters. A free space propagation channel is assumed. Group scenario files determine the mobile node locations and node movement. Specific traffic pattern/connection files are used to define mobile nodes as receivers and sources. For wire-line nodes, their traffic patterns and connections are defined in the test script. Each data point represents an average of at least six experiments with identical traffic models for each side subnet (which means both senders and receivers are identical for the ad hoc subnet, while only receivers are identical for different repetitions in the fixed subnet), but different randomly generated mobility scenarios for mobile nodes, and different data packet senders in the fixed subnet. For fairness, identical mobility and traffic scenarios were used for the simulations that use different protocol combinations of multicast routing protocols (see Section 4.1 above) for fixed network and MANET. Only one multicast group and one MGW are simulated for this mixed networking. The MGW is located in the middle of the L meters * L meters flat space, it has only one link to the fixed network. In the beginning of the simulation, the MGW joins each side multicast groups on each network, which are considered as the same group although they use different group addresses according to the *ns-2* definition. The average degree for the fixed node connection is three. A sender can also be a receiver. In terms of a multicast tree with RP, the RP is not a sender but can be a receiver. The number of senders and receivers in each subnet are scaled according to their subnet sizes.

Mobile nodes move randomly at a preset average speed according to a “random waypoint model”. Each node starts its journey from a random location to a random destination with a randomly chosen speed (uniformly distributed between 0 – some maximum speed).

Once the destination is reached, another random destination is targeted after a predefined pause time. The maximum speed is set to 1m/s. A zero pause time is used to create a harsher mobility environment [Cheng01].

A constant bit rate (CBR) at four packets (with size 512 bytes each) per second and per sender was chosen to simulate a uniform traffic pattern for both network domains.

Table 4.1 and Table 4.2 list the values for the essential parameters used in MAODV and ODMRP simulations.

Table 4.1: MAODV Simulated Parameter Values

Number of allowed Hello Losses	3
Group Hello interval	2 sec
Hello interval	1 sec
Time to wait to receive a MACT after sending a RREP	5 sec
Time to keep reverse route entries	5 sec
Life time of route table entry	50 sec
Max number of RREQ retransmission	3
Max time to wait for a RREP	5 sec
Time-To-Live (TTL/Network Diameter)	20

Table 4.2: ODMRP Simulated Parameter Values

Join-Query refresh interval	3 sec
Forwarding group timeout	9 sec
Time-To-Live (TTL)	20

4.3 Performance Evaluation Metrics

According to [QA01, Cheng01, CM99], the following quantitative metrics are to be used to evaluate the multicast gateway performance under different multicast protocol combinations in the fixed network and the MANET.

Packet Delivery Ratio: ratio of the number of packets actually delivered without duplicates to the destinations versus the number of data packets supposed to be received. This number represents the effectiveness and throughput of a protocol in delivering data to the intended receivers within the network. The number of data packets supposed to be received is a theoretical number projected from the multicast group member size and the number of packets sent from multicast sources. For some simulations, delivery ratio for each subnet was also analyzed, i.e., number of data packets originated and also received in that subnet versus data packets theoretically originated in that subnet. Data packets received by the MGW are not counted.

Number of control packets transmitted per data packet delivered (without duplicates): shows the efficiency as number of control packets expended in delivering a data packet to an intended receiver (group member).

4.4 Network Parameters

- *Simulation time*

Simulations were run from 100 seconds to 1500 seconds in steps of 100 seconds for different protocol combinations to explore how the data delivery ratio and the control overhead change with simulation time. Twenty fixed nodes and thirty mobile nodes were used. The number of senders was set as five (two for the fixed side and three for the ad hoc side). Twenty multicast group members (eight versus twelve for each side) were simulated. After these experiments, an appropriate simulation time was chosen for the following experiments.

- *Number of senders*

One to twenty senders (at steps of 5) were simulated. Two cases were conducted for one-sender simulations, i.e., placing the one sender either in the fixed subnet or in the ad hoc subnet. There were 20 fixed nodes and 30 mobile nodes. The distribution of senders in the fixed and the ad hoc subnet was scaled at 2:3. The number of multicast group members was fixed at 20 (eight versus twelve at a ratio 2:3 for each side).

- *Number of multicast group members*

With twenty fixed nodes and thirty mobile nodes, ten to fifty multicast group members (at 2:3 ratio for each side) were applied with the number of senders set at five.

- *Subnet size*

By varying the fixed and the ad hoc network sizes (10 vs. 40 nodes, 20 vs. 30 nodes, 30 vs. 20 nodes, and 40 vs. 10 nodes), we can measure the performance of the mixed network with MGW functionality. A total of 5 senders and 20 group members were introduced. The number of senders and receivers in each subnet were scaled according to their subnet sizes.

Chapter 5

5 Simulation Results

5.1 Simulation Time

The multicast gateway (MGW) transmits data packets between fixed and ad hoc subnets. When the MGW receives a data packet from the fixed side and the MGW is not the data source, except for forwarding data packet copies to all attached replicators, it also sends a copy to the routing agent that was configured in the MGW node. On the other side, when MGW receives a data packet from the ad hoc side that is not originated (either directly or indirectly) from itself, it does two things: send a copy to the fixed side by using the fixed side multicast protocols, and perform ad hoc side routing by using MAODV or ODMRP protocol. The order in which to perform these steps might impact the results. One preliminary study we did was to study the impact of MGW forwarding order on the mixed network performance after it receives data packets from the ad hoc side.

The experiments were done at the same time when we were studying the simulation time. Figure 5.1 and Figure 5.2 show the data delivery ratio for the fixed side, while the MGW forwarded a copy of the data packet to the ad hoc side first in the former one, and it sent a copy to the fixed side first in the latter. We can see that different MGW forwarding orders have no influence on the fixed side data delivery ratio. All eight protocol combinations showed greater than 99% data delivery ratios. The wireline multicast routing got stabilized quite early with perfect delivery ratio in this static subnet.

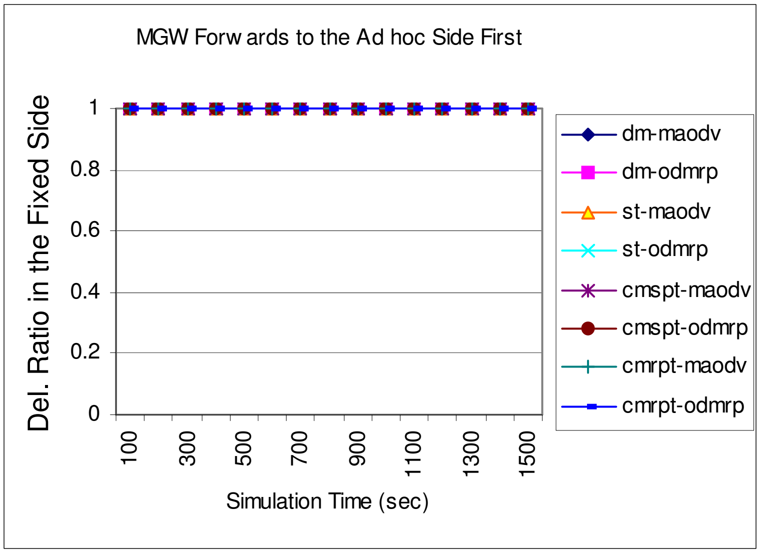

Figure 5.1: Data Packet Delivery Ratio in the Fixed Subnet as a Function of Simulation Time (MGW Forwards to the Ad hoc Side First)

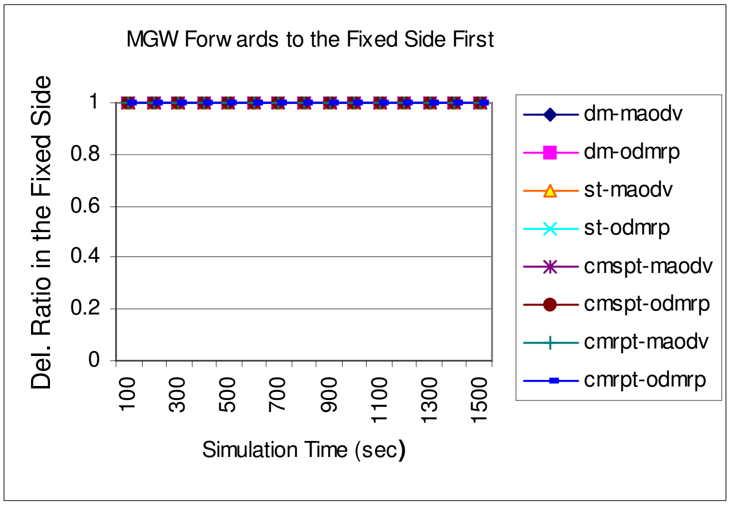

Figure 5.2: Data Packet Delivery Ratio in the Fixed Subnet as a Function of Simulation Time (MGW Forwards to the Fixed Side First)

Contrary to the fixed side data delivery ratio, the ad hoc side delivery ratio was affected (Figure 5.3 and Figure 5.4) by the forwarding order of the MGW node after it received data packets from the ad hoc side, especially for MAODV participated routings. During the 1500 seconds simulations, simulations using protocol combinations with MAODV outperformed or were not worse than those that used ODMRP. The MGW forwarding orders had little or no influence on the ODMRP participated multicasting, and trivial differences exist inside the ODMRP group (four protocol combinations). The mesh data structure in ODMRP is more robust and quicker in response. All MAODV combinations showed the highest delivery ratio around 200 – 400 seconds of simulation time, and degraded afterwards. Of the four MAODV combinations, cmspt-maodv showed superior delivery ratio over all the other combinations in both situations. In Figure 5.3, although the absolute delivery ratios in the ad hoc side are about 2 - 5% lower for dm-maodv and cmspt-maodv than in Figure 5.4, in which MGW forwards packets to the fixed side first, the variance among MAODV categories is smaller when the MGW forwards packets to the ad hoc side first (Figure 5.3). Based on these results, for the following simulations, we chose this configuration.

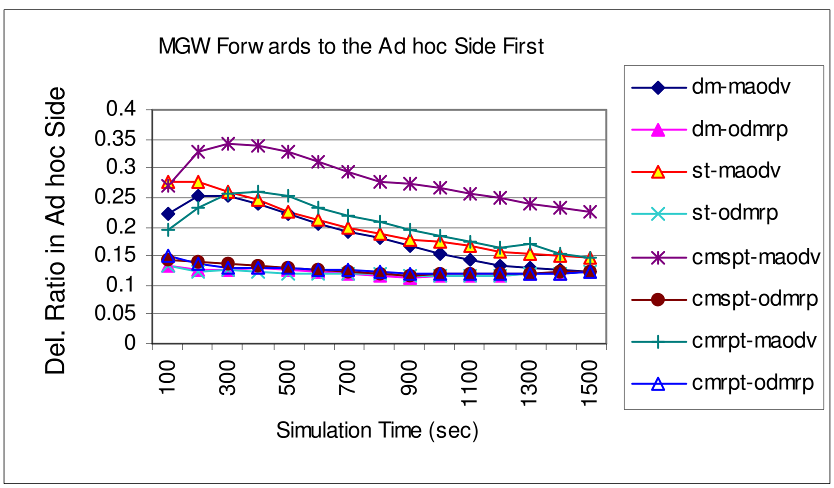

Figure 5.3: Data Packet Delivery Ratio in the Ad hoc Subnet as a Function of Simulation Time (MGW Forwards to the Ad hoc Side First)

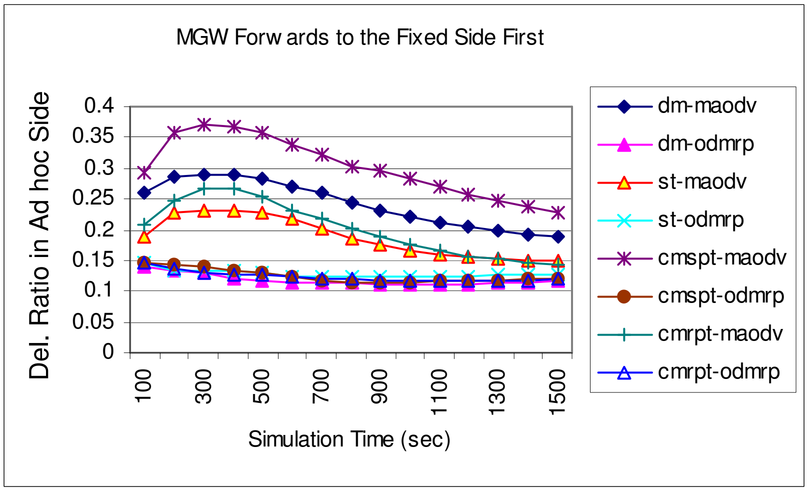

Figure 5.4: Data Packet Delivery Ratio in the Ad hoc Subnet as a Function of Simulation Time (MGW Forwards to the Fixed Side First)

Figure 5.5 demonstrates the overall data packet delivery ratio for the mixed network as a function of the simulation time. Figure 5.6 is the overall control overhead (more than 95% of the control messages were from the ad hoc side) per data packet delivered. In Figure 5.5, the highest delivery ratio appears to be 49.58% at time 300 seconds for the cmspt-maodv protocol combination, and its delivery ratio drops to 41% at time 1500 second. The other three MAODV involved combinations also show their highest delivery ratio at about 42% before 400 seconds simulation time, and decrease to 31 – 33% by the end of the simulations. During the 1500 seconds simulations, all four ODMRP combinations are close together with their delivery ratio decreasing from 27.7% to 22.5%. Source specific tree protocol combinations like dm-odmrp and cmspt-odmrp have evenly distributed traffic, and they show weakly higher delivery ratio than st-odmrp and cmrpt-odmrp combinations, whose fixed side RP tree structure has a concentrated traffic nature.

For the control overhead per data packet delivered, Figure 5.6 categorizes eight combinations into two distinct groups according to whether the ad hoc side protocol is MAODV or ODMRP. All ODMRP protocols generate above 70% control overhead, compared to MAODV involved protocols, which have less than 30% control overhead during the simulation. In ODMRP, each sender broadcasts route requests periodically, while in MAODV only one group leader sends out periodic Group Hellos through the network, therefore ODMRP results in much higher absolute routing (control) packet numbers (recall that we have 3 ad hoc senders in this simulation). Another thing we need to point out here is that although cmspt-maodv showed superior delivery ratio over the other 7 combinations, it also generates more control packets that makes it show ordinary (but less) control overhead within the MAODV group.

Compared with ODMRP, it seems that the performance of the MAODV participated multicasting did not stabilize as the simulation time increases. But by 900 seconds, the data delivery ratio and control overhead are relatively stable within the different categories. Therefore we chose 900 seconds as the simulation time for the future experiments. 900 seconds was also used in other studies [Cheng01, Qin01].

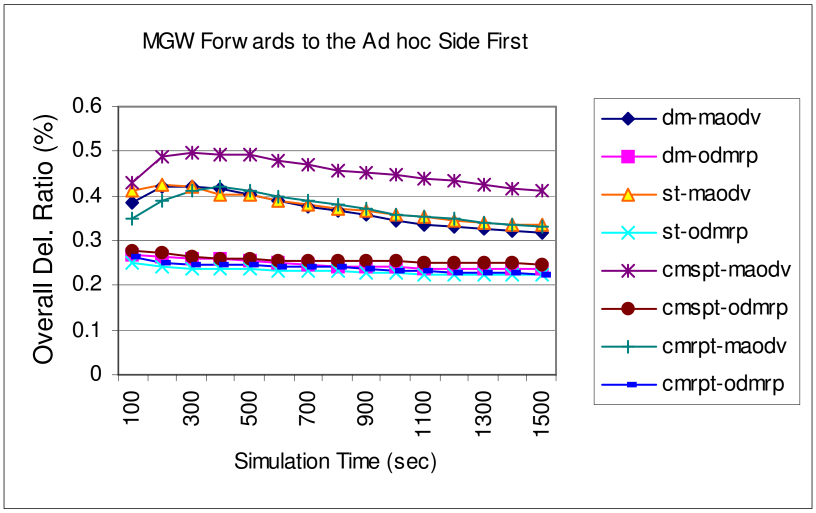

Figure 5.5: Overall Data Packet Delivery Ratio in the Mixed Network as a Function of Simulation Time (MGW Forwards to the Ad hoc Side First)

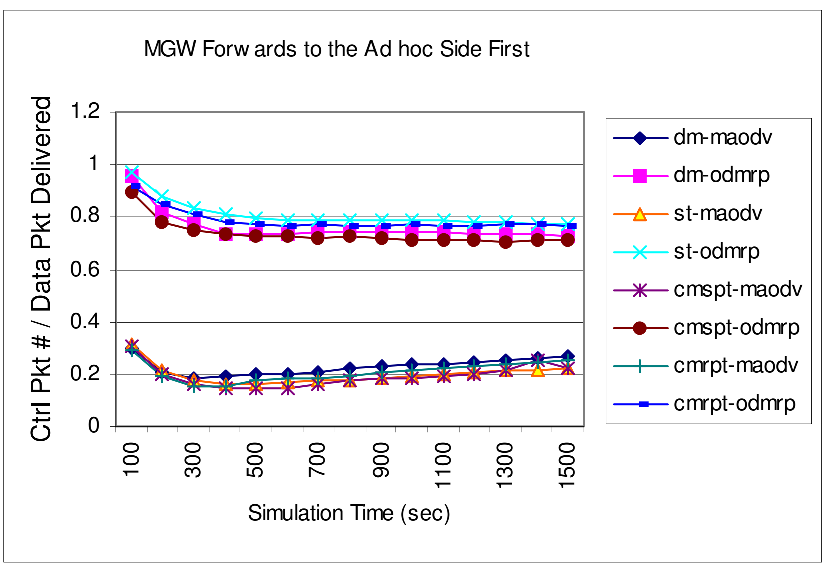

Figure 5.6: Control Overheads per Data Packet Delivered as a Function of Simulation Time (MGW Forwards to the Ad hoc Side First)

5.2 Number of Senders

To investigate the scalability of different kinds of multicast protocol combinations, we varied the number of senders in both subnets from only one sender (either in the fixed side or in the ad hoc side) up to twenty senders. In Figure 5.7, the delivery ratio with one sender is the average of two experimental results (each with seven repetitions) – one experiment with the only sender located in the fixed side, the other with the one sender in the ad hoc side. The same scenario applies to Figure 5.8 where we consider the control overhead. The original values for those two distinct scenarios are displayed in Table 5.1, where 1a represents the scenario that has the only sender in the fixed side, and 1b represent the scenario with the one sender in the ad hoc side.

From Figure 5.7, two categories emerge according to whether MAODV or ODMRP was used. With only one sender (either one or zero sender in the ad hoc side), ODMRP is 55.8% more effective than relative MAODV involved multicasting. But as the number of senders increased from 5 to 20 (the ad hoc side sender number ranged from 3 to 12), ODMRP data delivery ratios drop significantly and become at least 27% less effective than MAODV when the number of senders reaches 20. All protocol combinations do not scale well as the number of senders increases. In ODMRP, each sender periodically broadcasts route requests through the network. Together with the rebroadcasts of the request at intermediate nodes, more senders in the ad hoc side (greater than 2 mobile senders were simulated in this research) will result in much higher congestion and lower delivery ratio than MAODV, which has only one multicast group leader that sends periodic Group Hellos through the network. Also in MAODV, while the sender number increases, the multicast tree size enlarges, and with the roaming of the mobile nodes, the delivery ratio drops due to more frequent tree partitions. With only one sender, ODMRP's mesh topology works fine with not very much traffic congestion. At 5 senders, cmspt-maodv outperforms other MAODV combinations by 27%.

In Figure 5.8, all protocol combinations scale well in terms of the control overhead with the increase of the number of senders. MAODV outperforms the ODMRP combinations except for one case when the only sender is located in the ad hoc side, and when the two categories of MAODV and ODMRP show almost the same level of control overhead (1.51/1.55 vs. 1.63) (Table 5.1, 1b in overall control overhead). There exist significant control overhead degradations from 1 sender to 5 senders in both categories. The four MAODV combinations have up to 86% less routing overhead than the ODMRP combinations when the sender number reaches 20. In MAODV the routing overhead for tree maintenance is independent of the number of senders, whereas in ODMRP, the increase in the number of senders generates more control messages due to source broadcasting. Hence ODMRP involved protocols demonstrate higher control overhead than MAODV. But since the percentage increase of data packets delivered is larger than that of the control packets increase, all protocol combinations show control overhead degradation with an increasing number of senders.

Table 5.1: Original Values and Their Averages for Two One-Sender Cases

		dm - maodv	dm - odmrp	st - maodv	st - odmrp	cmspt - maodv	cmspt - odmrp	cmrpt - maodv	cmrpt - odmrp
Overall -	**1a**	0.55458	0.85925	0.5512	0.8461	0.55437	0.85904	0.5512	0.84607
Delivery-	**1b**	0.28871	0.45861	0.2851	0.4572	0.2886	0.45848	0.2851	0.45716
Ratio	**(1a+1b)/2**	0.42164	0.65893	0.4181	0.6516	0.42149	0.65876	0.4181	0.65161
Overall -	**1a**	0.39722	0.70027	0.4723	0.714	0.39584	0.69946	0.472	0.71398
Control -	**1b**	1.55404	1.63159	1.5171	1.6314	1.55292	1.63107	1.516	1.63135
Overhead	**(1a+1b)/2**	0.97563	1.16593	0.9947	1.1727	0.97438	1.16526	0.994	1.17266

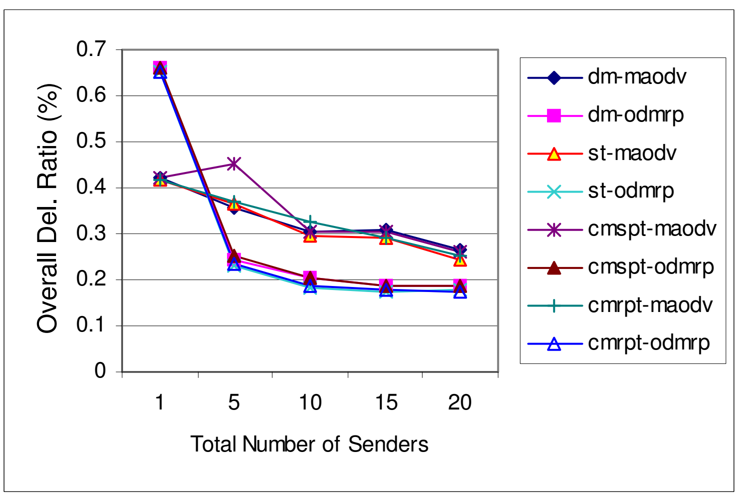

Figure 5.7: Overall Data Packet Delivery Ratio as a Function of the Total Number of Senders

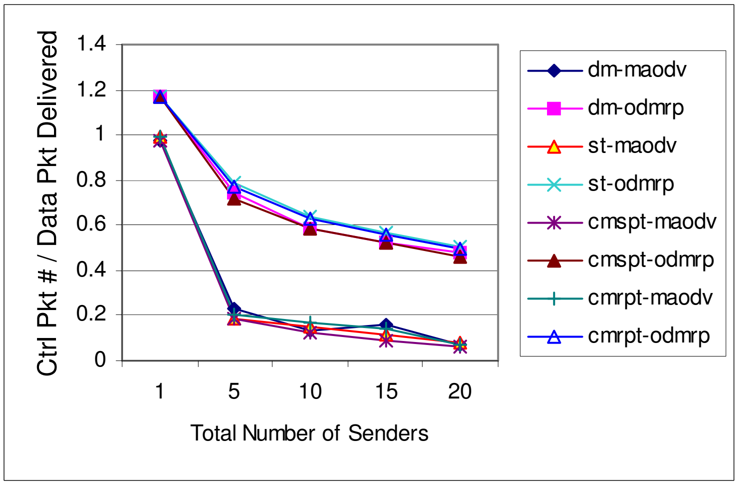

Figure 5.8: Control Overheads per Data Packet Delivered as a Function of Number of Senders

It further shows in Figure 5.9 and Figure 5.10 that the performance effectiveness for the mixed network is mostly affected by the ad hoc side protocols. MAODV outperforms ODMRP by itself, and the fixed side protocols all have more than 99% delivery ratio by themselves. The one sender case in Figure 5.9 represents the scenario with the only sender located in the ad hoc side; and the one sender point in Figure 5.10 is the scenario with the only sender in the fixed side.

Same as in Figure 5.3, Figure 5.7 and Figure 5.9 show the outlier of cmspt-maodv in five senders cases. Although the trend should be degrading with the increase of number of senders, this superior delivery ratio exists for some unknown reason. But when one looks at Figure 5.8, cmspt-maodv shows ordinary control overhead as the other three MAODV combinations when there are five senders, which indicates that by delivering more data packets, cmspt-maodv also generated higher amount of control packets.

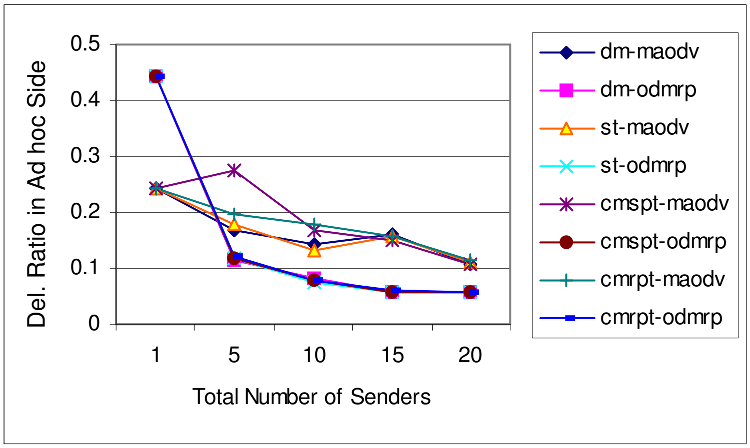

Figure 5.9: Data Delivery Ratio in the Ad hoc Side as a Function of the Total Number of Senders

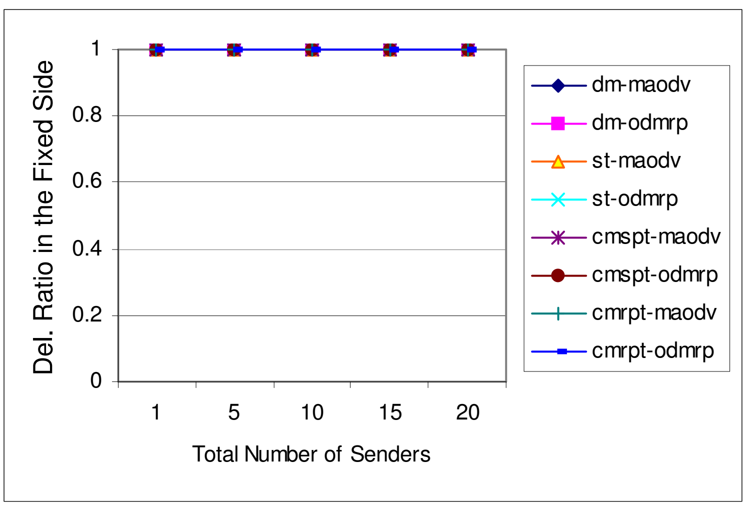

Figure 5.10: Data Delivery Ratio in the Fixed Side as a Function of the Total Number of Senders

5.3 Number of Multicast Group Members

For the third set of simulations, we tested the mixed network performance when increasing the multicast group size. In Figure 5.11, the overall delivery ratio for MAODV and ODMRP categories degrade as the total number of multicast group members increase from ten to fifty (when all 50 nodes in both subnets are multicast receivers). MAODV still clearly outperforms ODMRP combinations. With the receiver number increases in ODMRP, more Join Replies (to the Join Query from the source) messages will be generated by the receivers and propagated by the forward group members through the network. This definitely incurs higher traffic congestion, and drags the delivery ratio down. For MAODV combinations, while the ad hoc side subnet has more receivers, the multicast tree size increases, which introduces frequent tree partitions and more data packet drops. As a result, it lowers the delivery ratio.

In terms of the control overhead, one can see that both categories scale quite well (Figure 5.12). ODMRP combinations show a routing overhead decrease from 1.29 to 0.44 with increasing group size, and MAODV combinations stay at the same level in this process. ODMRP's lower delivery ratio and relatively higher control packet numbers result in its much higher control overhead than MAODV involved combinations.

Figure 5.13 and Figure 5.14 demonstrate that the ad hoc side protocols play major roles in the mixed network performance. Also as we mentioned before, more than 95% of control packets are from the ad hoc side.

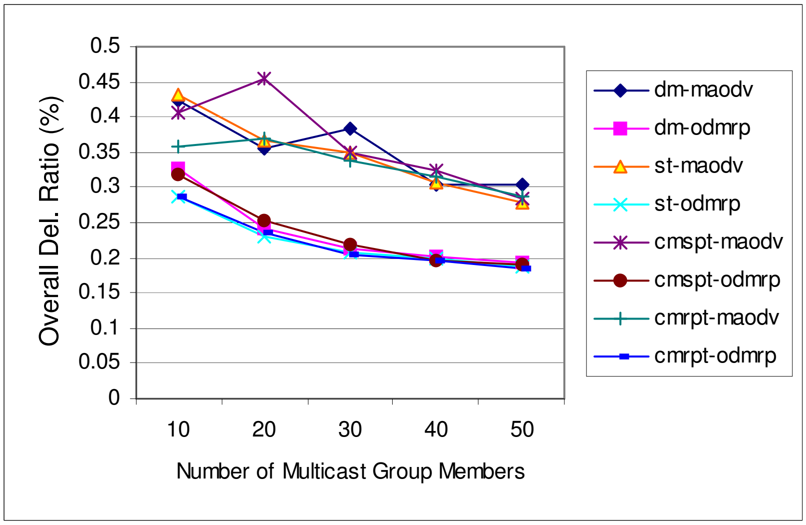

Figure 5.11: Overall Data Packet Delivery Ratio as a Function of the Number of Multicast Group Members

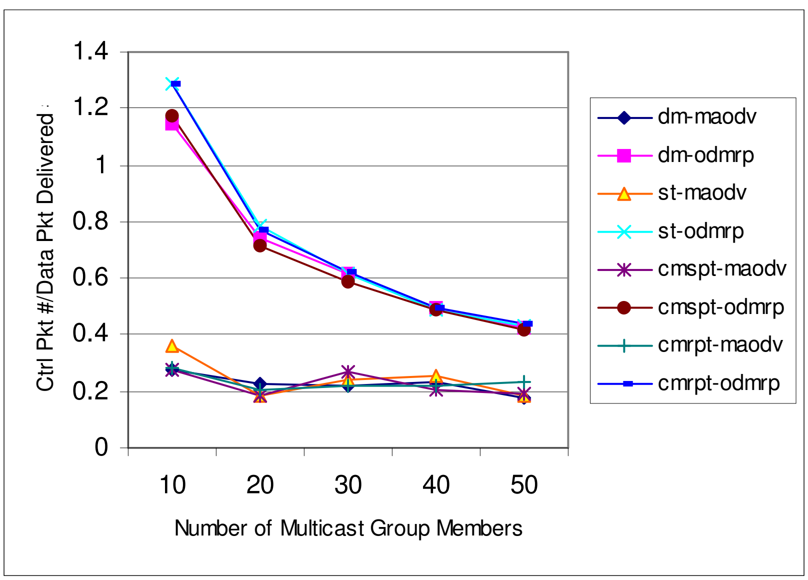

Figure 5.12: Control Overhead per Data Packet Delivered as a Function of the Number of Multicast Group Members

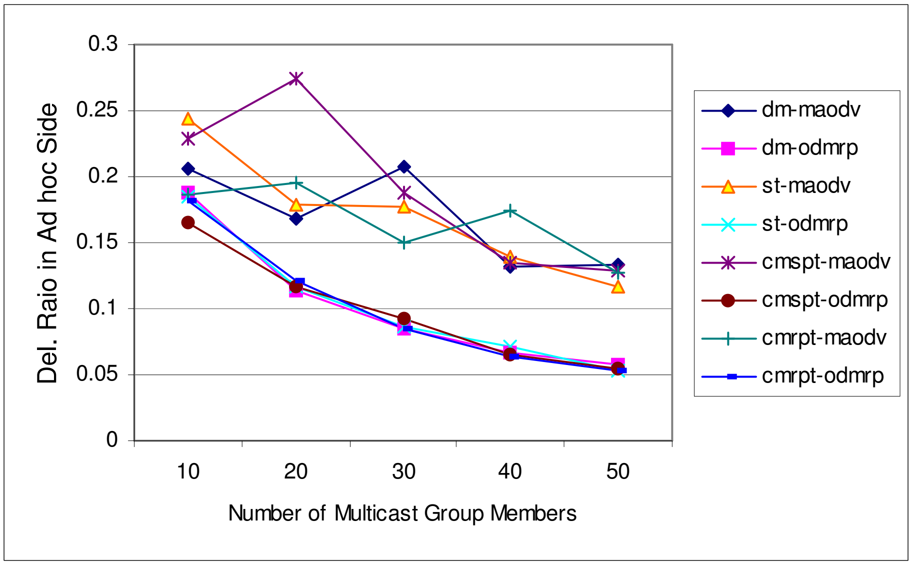

Figure 5.13: Data Packet Delivery Ratio in the Ad hoc Side as a Function of the Number of Multicast Group Members

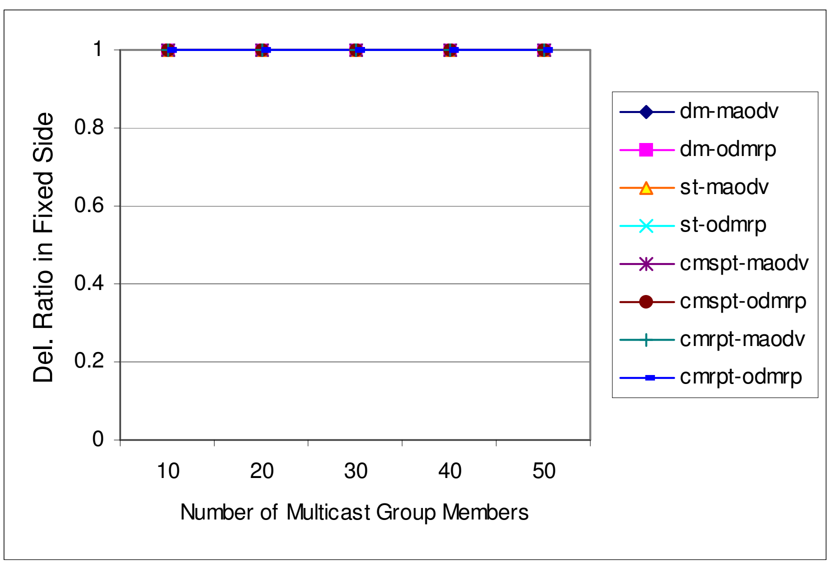

Figure 5.14: Data Delivery Ratio in the Fixed Side as a Function of the Number of Multicast Group Members

5.4 Subnet Size

In the last set of simulations, we studied the network performance by changing the relative subnet size on each side. This has special interest for the mixed network environment. In Figure 5.15, with the number of the fixed side nodes increasing from 10 to 40 (mobile ad hoc node numbers decreasing from 40 to 10), all protocol combinations scale well for the data delivery ratios. MAODV combinations increase from 16% to 86%, while ODMRPs increase from 10% to 90%. MAODV shows almost linear progress except for one point at 20 fixed nodes for cmspt-maodv. In 10, 20 and 30 fixed nodes situations (4, 3 and 2 senders in the ad hoc side respectively) ODMRP combinations have lower delivery ratio than MAODV combinations. But an overlap happens when the number of ad hoc nodes decreases to 10 and there is only one sender in the ad hoc side, where ODMRP shows a slightly higher ratio than MAODV in the combinations with st, cmspt and cmrpt respectively. This is due to the lower number of control packets generated by this single ad hoc sender in ODMRP.

The control overheads in Figure 5.16 trend down as the number of fixed nodes increases (recall that less than 5% control packets are generated from the fixed side when we have 20 fixed nodes and 30 ad hoc nodes). Both MAODV and ODMRP show a significant decrease as the number of fixed nodes increases from 10 to 20. At the point of 40 fixed nodes, the control overheads are less than 0.06.

Figure 5.17 and Figure 5.18 represent data delivery ratio for each subnet side. In Figure 5.17, similar to what is shown in Figure 5.15, with 10 mobile ad hoc nodes (one ad hoc sender only), ODMRP shows some 3.6% higher delivery ratio than the MAODV cases. For 40, 30 and 20 ad hoc nodes scenarios, ODMRP combinations have lower delivery ratios than MAODV combinations.

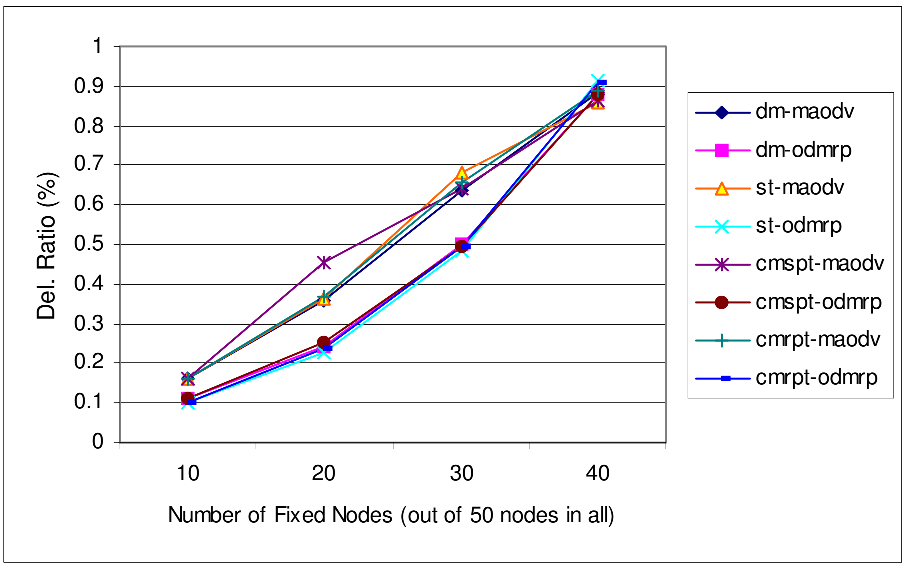

Figure 5.15: Overall Data Packet Delivery Ratio as a Function of the Subnet Size

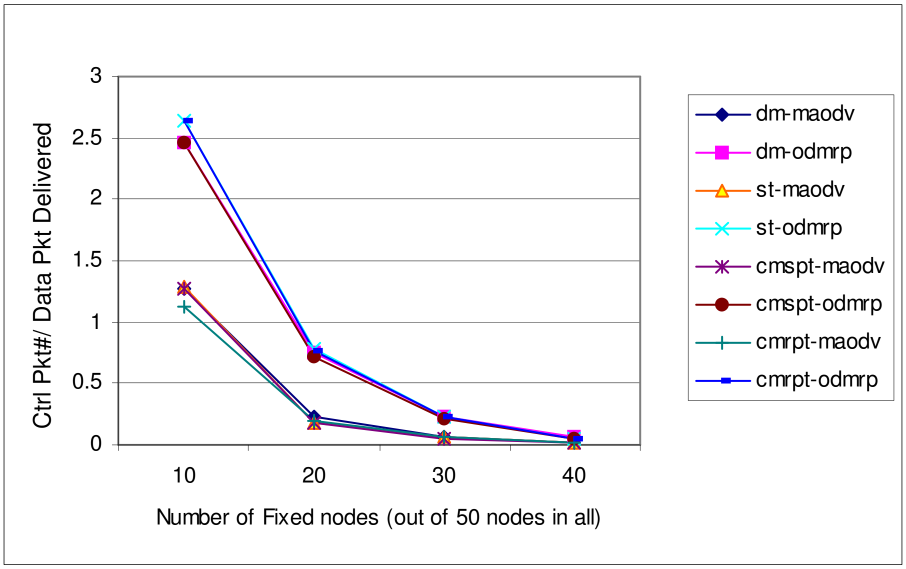

Figure 5.16: Control Overhead per Data Packet Delivered as a Function of the Subnet Size

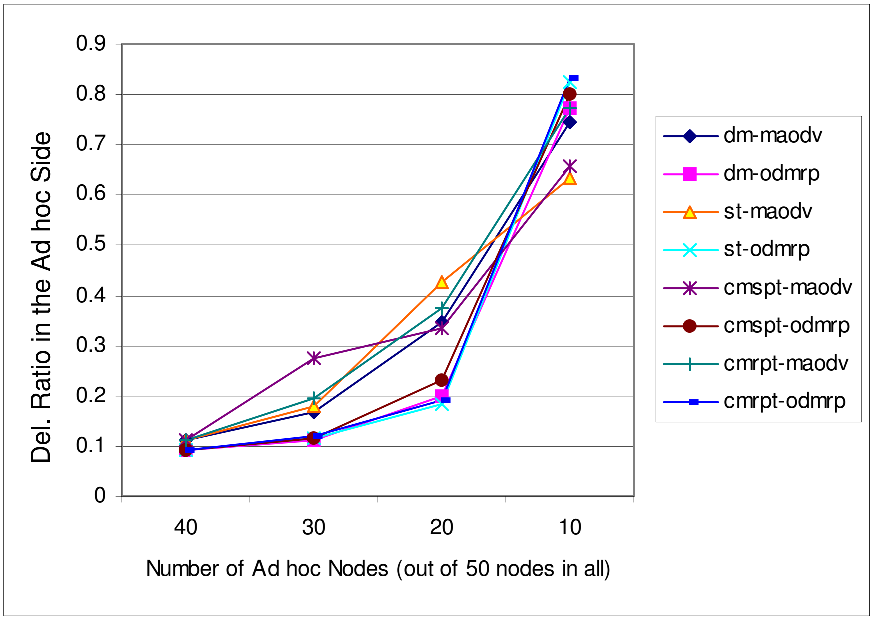

Figure 5.17: Data Packet Delivery Ratio in the Ad hoc Side as a Function of the Subnet Size

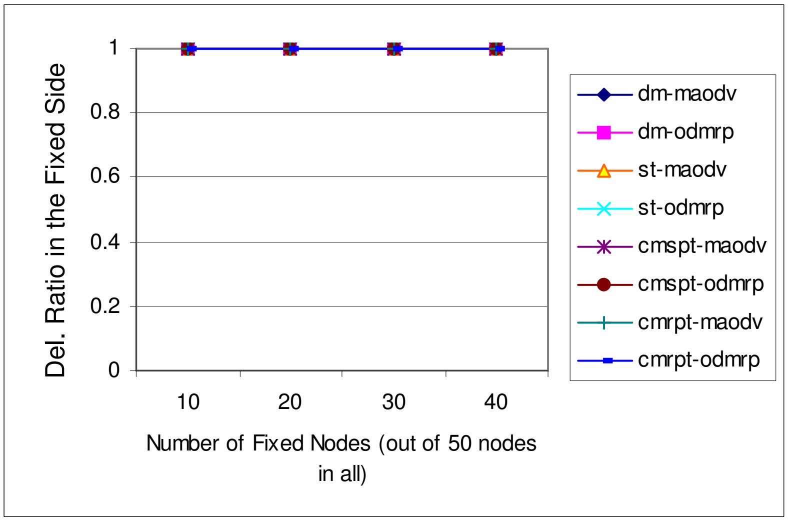

Figure 5.18: Data Delivery Ratio in the Fixed Side as a Function of the Subnet Size

5.5 Protocol Combination Analysis

We ran simulations across the mixed network with eight protocol combinations for the fixed and wireless mobile ad hoc multicast routing. Overall data delivery ratio and control overhead per data packet delivered were analyzed as a function of the number of senders, number of multicast group members and the size of the subnets. Data delivery ratio was also analyzed for each subnet separately to compare the protocol performance for each side. Control overhead was not considered separately for each side since the fixed side multicast routing generates only tens to hundreds of control packets, which account for less than ten percent of the total control messages in the mixed network. The four fixed side protocols worked quite stable and perfect with always more than 99%

delivery ratios. For the ad hoc side protocols, MAODV outperformed ODMRP with higher data delivery ratio except for two cases: one was when there was only one sender (in either subnet) for the whole network, ODMRP showed 56% higher delivery ratio than MAODV; the other case happened when there were just ten nodes in the ad hoc side (forty nodes for the fixed side) with only one ad hoc sender, ODMRP weakly beat MAODV for the data delivery ratio. When considering the entire network, it was the ad hoc side protocols that played the determinant roles for the overall performance. Except for the above two scenarios where ODMRP displayed higher delivery ratio, MAODV involved protocol combinations outperformed ODMRP involved ones with higher data delivery ratio and relatively lower control overhead.

With the increase of data sources or the number of receivers, all protocol combinations demonstrated degraded delivery ratio, on the other hand, they all showed non-increased relative control overhead. The more ad hoc mobile nodes existed, the worse the relative data delivery ratio and the higher the control overhead.

Although the four fixed side protocols work perfectly by themselves, after they get combined and cooperate through the MGW with the ad hoc side protocols, trivial differences appeared among them. And it seems that the differences are not consistent between MAODV and ODMRP groups.

After MGW receives a data packet from the ad hoc side, the data forwarding order of MGW to the fixed side and the ad hoc side has some effects on the ad hoc side delivery ratio as well as the overall network delivery ratio.

Chapter 6

6 Conclusions and Future Work

Our multicast gateway (MGW) transmits data packets between the fixed wire-line network and the mobile wireless ad hoc network. Eight protocol combinations were applied to the simulations. It is found that the overall network performance is primarily determined by the ad hoc side protocols (MAODV or ODMRP) that were used. In general, MAODV involved protocol combinations outperform ODMRP involved ones in terms of delivery ratio and control overhead. An exception happens when there is only one sender in the whole network, or when there are only ten mobile nodes (with only one sender in the ad hoc side), in which case we see higher delivery ratios for the ODMRP involved protocols. The periodic broadcasting of Join-Query by each sender in the ODMRP introduces a large amount of control messages and traffic congestions in this subnet. As a result, when there is more than one sender in the ad hoc side, ODMRP shows lower data delivery with higher control overhead compared with MAODV, which has only one group leader to send periodic Group Hellos. But in the case that the number of sender in the ad hoc side is null or one, the traffic congestion is not bad, resulting in ODMRP having somewhat higher delivery ratio than MAODV due to its mesh structure.

The MGW has both a wireline multicast interface and a wireless communication interface. It joins the multicast groups on both sides explicitly in the beginning (not because it is interested in the data itself, but because it takes the data forwarding responsibility). Due to the high delivery ratio by the fixed side multicast protocols, MGW receives greater than 99% data packets originated in the fixed side and forwards almost all the data packets that arrive at it from the ad hoc side to the fixed domain. But the problem is that by following the current ad hoc multicast protocols (either MAODV or ODMRP), the chances for MGW to get all data packets originated from the ad hoc side is low, which is unfavourable to the fixed side receivers receiving from the ad hoc side. In order to raise the delivery ratio for the mixed network, future work could modify the current ad hoc side protocols to treat the MGW not simply as an ordinary ad hoc mobile node, but make it be recognized by all the ad hoc side senders, so that they could do their best to deliver their data to the MGW.

In this study, we designed and implemented a MGW model to work for one multicast group between one fixed subnet and one ad hoc subnet. Future investigation could expend this to study more than one MGW, and more than one ad hoc subnet (what happens if mobile node moves from one subnet to another subnet), as well as more than one multicast group. As well, our MGW is designed to replace the data packet source ID with its own ID when it receives a datagram from one domain and forwards it to the other domain. In cases that the multicast group members need to know the real data source even though the data is from the other domain, modifications (e.g. by including an extension field containing the address of the original source in the packet headers) have to be conducted in order to keep this information.

The implementation of fixed side multicast protocols in *ns-2* work perfectly in our static and relatively small fixed subnet. In real large network (e.g., the Internet) with thousands of nodes, more control messages could be expected due to frequent group joining and leaving by receivers, and the delivery level needs to be evaluated. With varied group member distribution pattern and group density, protocols in PIM-dense mode or PIM-sparse mode should be able to show preference in terms of performance.

The impact of MGW forwarding order on the ad hoc side and overall data delivery ratios needs further understanding in *ns-2*. It gives evidence and leaves space for future mixed network multicast routing as well.

Bibliography

[Bal97] A. Ballardie. Core Based Tree (CBT) Multicast Routing Architecture. RFC 2201, September 1997. Available at http://www.faqs.org/rfcs/rfc2201.html

[Ball97] A. Ballardie. Core Based Tree (CBT version 2) Multicast Routing:Protocol Specification. RFC 2189, September 1997.
Available at http://www.faqs.org/rfcs/rfc2189.html

[BHW97] Richard B. Bunt, Tim G. Harrison, Carrey L. Williamson, Wayne L. Mackrell. Mobile Multicast (MoM) Protocol: Multicast for Mobile Hosts. In Proceeding of ACM Mobicom, Budapest, Hungary, Pages 151-160, 1997.

[BFC93] Tony Ballardie, Paul Francis, and Jon Crowcroft. Core Based Trees (CBT): An Architecture for Scalable Inter-Domain Multicast Routing. In Proceedings of ACM Special Interest Group on Computer Communication (SIGCOMM'93), San Francisco, CA, Pages 85-95, October 1993.

[BMJ98] Josh Broch, David A. Maltz, David B. Johnson, Yih-chun Hu and Jorjeta Jetcheva. A Performance Comparison of Multihop Wireless Ad hoc network Routing Protocols. In Proceedings of the 4th International Conference of Mobile Computing and networking (ACM MOBICOM'98), Pages 85-97, October 1998.

[CG97] Ching-Chuan Chiang and Mario Gerla. Routing and Multicast in Multihop, Mobile Wireless Networks. In IEEE International Conference on Universal Personal Communications (ICUPC'97), Pages 28-33, 1997.

[CGZ97] Ching-Chuan Chiang, Mario Gerla and Lixia Zhang. Shared Tree Wireless Network Multicast. In Proceedings of IEEE 6th International Conference on Computer Communincations and Networks (ICCCN'97), Las Vegas, Nevada. Pages 28-33, September 1997.

[CGZ98] Ching-Chuan Chiang, Mario Gerla and Lixia Zhang. Adaptive Shared Tree Multicast in Mobile Wireless Networks. In Proceedigns of IEEE Global Communications Conference, Sydney, Australia, Pages 1817-1822, November 1998.

[CGZ99] Ching-Chuan Chiang, Mario Gerla and Lixia Zhang. Tree Multicast Stategies in Mobile, Multihop Wireless Networks. ACM/Baltzer mobile Networks and Applications, Vol. 4, No. 3, Pages 193-207, 1999.

[Cheng01] E. Cheng. On-demand Multicast Routing in Mobile Ad-hoc Networks. M.Eng. Thesis, Carleton University, Department of Systems and Computer Engineering, 2001.

[CM99] S. Corson and J. Macker. "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issue and Evaluation Considerations", RFC2501, IETF, January 1999. Available at http://www.ietf.org/rfc/rfc2501.txt

[DEF96] S.E.Deering, D.Estrin, D.Farinacci, V.Jacobson, C-G Liu and L.Wei. The PIM Architecture for Wide-Area Multicast Routing. IEEE/ACM Transactions on Networking, Vol. 4, No. 2, Pages 153-162, April 1996.

[DEF98] S.E.Deering, D.Estrin, D.Farinacci, M.handley, A. Helmy, V.Jacobson, C.Liu, P.Sharmand, D.Thalerand L.Wei. Protocol Independent Multicast-sparse Mode (PIM-SM): Motivation and Architecture. Draft-ietf-pim-arch-05.txt, IETF, August 1998.

[DEF99] S.E.Deering, D.Estrin, D.Farinacci, V.Jacobson, A.Helmy, David Meyer and L.Wei. Protocol Independent Multicast Version 2 Dense Mode Specification. IETF, draft-ietf-pim-v2-dm-03.txt, June 1999.

[Fen97] B. Fenner. Internet Group Management Protocol, Version 2. RFC 2236, Internet Engineering Task Force, November 1997. Available at http://www.faqs/rfcs/rfc2236.html

[FV99] Kevin Fall and Kannan Varadhan (Eds). The *ns* Manual. 1999. Available at http://www.isi.edu/nsnam/ns/ns-documentation.html

[GLS00] M. Gerla, S.J. Lee, and W. Su. On-demand Multicast Routing Protocol (ODMRP) for Ad hoc Networks. IETF, draft-ietf-manet-odmrp-02.txt, 2000.

[BL00] S.H. Bae, S.J. Lee, W. Su and M. Gerla. The Design, Implementation and Performance Evaluation of the On-demand Multicast Routing Protocol in Multihop Wireless Networks. IETF Network, Vol. 14 (1), Pages 70-77, January 2000.

[Hed88] C. Hedrick. Routing Information Protocol. RFC 1058, Internet Engineering Task Force, June 1998. Available at http://www.faqs/rfcs/rfc1058.html

[JJ01] Jorjeta G. Jetcheva and David B. Johnson. Adaptive Demand-Driven Multicast Routing in Multi-Hop Wireless Ad hoc networks. The ACM Symposium on Mobile Ad Hoc Networking & Computing (MOBIHOC'01), Long Beach, CA. Pages 33-44, October 2001.

[JT87] J.Jubin and J.Tornow. The DARPA Packet Radio Network Protocols. Proceedings of the IEEE, 75(1):21-32, 1987.

[Kunz02] Thomas Kunz. Multicasting: From Fixed Networks to Ad-hoc Networks. In Handbook of Wireless Networks and Mobile Computing. Ivan Stojmenovic (ed). Pages 495-507, John Wiley & Sons, ISBN 0-471-41902-8, 2002.

[Moy98] J. Moy. OSPF Version 2. RFC 2328, April 1998. Available at http://www.faqs/rfcs/rfc2328.html

[Paul98] S. Paul. Multicasting on the Internet and its Applications. Kluwer Academic Publishers, ISBN 0-792-38200-5, 1998.

[Per96] C. Perkins. IP Mobility Support. RFC 2002, Mobile IP Networking Group, October 1996. Available at http://www.ietf.org/rfc/rfc2002.txt

[Per97] C. Perkins. Mobile IP Design Principles and Practices. Addison Wesley, ISBN 0-201-63469-4, 1997.

[RP99] E. Royer, and C. Perkins. Multicast Operation of the Ad-hoc On-Demand Distance Vector Routing Protocol. Proceedings of the 5th Annual ACM/IEEE Annual Conference on Mobile Computing and Networking, Seattle, USA, August 1999, Pages 207-218.

[QA01] B. Quinn and K. Almeroth. IP Multicast Applications: Challenges and Solutions, RFC3170, IETF, September 2001. Available at
http://www.ietf.org/rfc/rfc2501.txt

[Qin01] L. Qin. Pro-Active Route Maintenance in DSR. M.Sc. Thesis, Carleton University, School of Computer Science, 2001.

[SSK00] Hee-Soon Shin, Young-Joo Suh, and Dong-Hee Kwon. Multicast Routing Protocol by Multicast Agent in Mobile Networks. In Proceeding of International Conference on Parallel Processing, Pages 271-278, August 2000.

[WPD88] D. Waitzman, C.Partridge and S. Deering (eds). Distance Vector Multicast Routing Protocol. RFC 1075, BBN STC and Stanford University, November. 1998. Available at http://www.ietf.org/rfc/rfc1075.txt

[WZ01] Ralph Wittmann and Martina Zitterbart. Multicast Communication: Protocols, Programming and Applications. Morgan Kaufman Publishers Inc. ISBN 1-558-60645-9, May 2000.

APPENDIX A Configuration

The following configuration applies to all the simulations described in this thesis.

Computer: Pentium II, 128 MB RAM

Operating System: Linux Red Hat 6.2

NS-2 version: ns-allinone-2.1b6

URL: http://www.isi.edu/nsnam/ns

To install *ns-2* simply do a "./install" in the root directory of the extracted folder, i.e. ns-allinone-2.1b6, and follow the further instructions.

This thesis was done based on the previous implementations of MAODV and ODMRP on ns-2.1b6 [Cheng01]. Refer to [Cheng01] for detailed software package installation and configuration for these two protocols.

APPENDIX B Modifications in OTcl for MGW Configuration

Location: ns-allinone-2.1b6/ns-2.1b6/tcl/lib/

Files: *ns-lib.tcl, ns-node.tcl, ns-mobilenode.tcl, ns-route.tcl, and*

ns-allinone-2.1b6/ns-2.1b6/tcl/mcast/ns-mcast.tcl

The following is a brief summary of the implementation work:

- In *ns-lib.tcl*, we define a new node type as 'MGW' in method *get-nodetype()*. This node type is assigned to a new node when it has a 'routingAgent' (e.g. MAODV or ODMRP) configured inside and it has 'wiredRouting' with status as 'ON'. In our simulations, the multicast gateway (MGW) is the only node of that type.

- In *ns-mcast.tcl*, in method *join-group()*, when the MGW (marked by 'MGW' node type) wants to join the fixed side multicast group with a receiving agent, that agent is enabled in each replicator (configured in MGW) for this group as any other fixed group member does. Meanwhile, the routing agent (MAODV or ODMRP) in MGW is also enabled in the replicators in order to receive packets from the fixed side and forward them to the ad hoc side. Modifications for group leaving of MGW is not implemented since in all the simulations, MGW joins both groups in the beginning and does not intended to leave as long as the multicast group exists.

- In *ns-node.tcl*, some new methods and modifications have been done as follows:

- a new method *set-node-addressMGW()* is added to initialize the ARP table and the number of network interfaces.
- In the new method *mk-default-classifierMGW()*, several types of classifiers are created, initialized and connected with some other components inside the MGW. These classifiers are *classifier_mgw_*, *classifier_*, *multiclassifier_*, *switch_* and *dmux_mgw_*.
- Method *entry-NewMGW()* is added to return the *switch_* as the node entry for MGW.
- In method *attach(agent port)*, if this node is the MGW and the *port* number is not 255, then call method *add-target-NewMGW( )* as a usual fixed node to target the agent to the *dmux_* and connect it to the node entry (the *switch_*).
- Method *install-defaulttarget-NewMGW()* directs the *classifier_mgw_* to its default target (routing agent or the receiving target before the routing agent).

- In *ns-mobilenode.tcl*, the following modifications have been done:
  - In method *add-target (agent port)*, the routing agent is targeted to the *demux_mgw_* at port 255, and is targeted to the *classifier_mgw_* as the default one.
  - In method *add-interface()*, replace the normal node entry object with the *classifier_mgw_* object for a MGW node, since the latter is the node entrance for MGW's wireless mobile ad hoc communication.

- In *ns-route.tcl*, in our mixed network multicast simulations, we define a bunch of fixed node first, then the MGW node before we initialize the ad hoc nodes. Therefore, in method *compute-flat-routes()*, we need to retrieve the highest node id number for the fixed side (instead of the total node number) to be used for the fixed side route computing.

APPENDIX C Wireline Multicast Protocol Modifications

Files: *ns-allinone-2.1b6/ns-2.1b6/tcl/mcast/DM.tcl, ns-allinone-2.1b6/ns-2.1b6/tcl/mcast/ST.tcl, ns-allinone-2.1b6/ns-2.1b6/tcl/ctr-mcast/CtrMcast.tcl*

The following is a brief description of the implementation changes:

In method *init(simulation node)* of class DM (in the file *DM.tcl*), when the *node* is the MGW, the MGW's routing agent (MAODV or ODMRP) is targeted to its node entry (*switch_*). At the same time, through the OTcl linkage, an instance variable *target3_* in file *ns-allinone-2.1b6/ns-2.1b6/aodv/aodv.cc* or *ns-allinone-2.1b6/ns-2.1b6/odmrp/odmrp.cc* is assigned to be the MGW node entry in method command() of class *AODV* or *ODMRP*.

In method *init(simulation, node)* of class *ST* (in the file *ST.tcl*), usually, if it is a fixed node with a sending agent, an encapsulator object will be created and added to the encapsulator list for this group, also the encapsulator will be targeted by the sending agent and connected to the decapsulator object in the PR, so that each data packet will be first tunneled to the decapsulator object inside the rendezvous point node (RP) assigned for this group. As an implicit data source for the fixed side with data packet from the ad hoc side, the MGW is also configured to have an encapsulator object that is targeted by the MGW's routing agent (MAODV or ODMRP) and connected to the decapsulator in the RP. At the same time, through the OTcl linkage, an instance variable *target2_* in the file *ns-allinone-2.1b6/ns-2.1b6/aodv/aodv.cc* or the file *ns-allinone-2.1b6/ns-2.1b6/odmrp/odmrp.cc* is assigned to be this encapsulator in the MGW node in method command() of class *AODV* or *ODMRP*.

In the file *CtrMcast.tcl*, there is a central multicast computing agent that controls two kinds of multicast tree manipulation, i.e., rendezvous point tree (RPT) or shortest path tree (SPT). Inside the method of *join-group()*, for the MGW node, in case of RPT and the RP is not the MGW, an encapsulator is created as the target of the ad hoc routing agent and is connected to the RP's decapsulator. On the other hand, in case of SPT, the ad hoc routing agent is targeted to the MGW's node entry. Modifications for group leaving of MGW is not implemented since in our simulations, MGW joins both groups in the beginning and does not intend to leave as long as the multicast groups exist.

APPENDIX D Wireless Mobile Ad-hoc Multicast Protocol Modifications

Files: *ns-allinone-2.1b6/ns-2.1b6/aodv/aodv.h*, *ns-allinone-2.1b6/ns-2.1b6/aodv/aodv.cc*, *ns-allinone-2.1b6/ns-2.1b6/odmrp/odmrp.h* , *ns-allinone-2.1b6/ns-2.1b6/odmrp/odmrp.cc*

In both *aodv.h* and *odmrp.h*, the total number of the ad hoc node mobile nodes (except MGW) is defined as a constant *NODE_NUM*. Two instance variables for the target of routing agent in MGW are defined as *target2_* and *target3_*, which will be assigned to be the encapsulator object (RP tree multicasting) and the node entry (non-RP tree type multicasting) respectively. An instance array variable *cur_event* is used to eliminate any duplicate data packet that arrives at the MGW in the ad hoc side from being sent to the fixed side. Also two methods *sendToTarget2(Packet* p, Handler* h)* and *sendToTarget3(Packet p, Handler h)* are used to let the encapsulator object or the node entry object in the MGW receive the data.

In *aodv.cc* and *odmrp.cc*, the constructor initializes the instance variables. The *command(int argc, const char*const* argv)* method assigns the object of the encapsulator or the node entry to the variable *target2_* or *target3_* respectively. In the instance method *recv(Packet *p, Handler *h)*, modifications concerning the MGW functionalities follow the following algorithm:

- If the ad hoc routing agent in MGW receives a data packet from the fixed domain (with group address different from the one on the ad hoc side, though these addresses are treated as the same group in our simulations), and this is not a looped back data packet originated in the ad hoc side, then we modify the packet headers by changing the multicast destination address to the ad hoc side group address, making the MGW be the data source, and initializing the TTL to be the constant value of *NETWORK_DIAMETER* (We used 20 here. By increasing the TTL value from the default 4 to 20 now, one could reduce the packet drop number and help to improve the data packet delivery). After this, the newly packed datagram is ready to be forwarded to the ad hoc domain by calling the routing procedure in class *AODV* or *ODMRP*.

- When the MGW routing agent receives a non-duplicate data packet from other mobile ad hoc nodes with a non-zero TTL value, it makes a copy of this data packet, changes the data copy's source ID to itself and let the data copy's destination be the fixed side group address. Then the routing agent forwards the original data packet to the ad hoc side before it forwards the packet copy to its targeted encapsulator or the node entry, depending on whether the RP tree or non-RP tree multicast protocol is used in the fixed side.

APPENDIX E Creating Mobile Node Movement Scenario File

Node movement scenario files are generated by using the "*setdest*" scenario generation program which is located in the directory *ns-allinone-2.1b6/ns-2.1b6/indep-utils/cmu-scen-gen/setdest*. The command line to generate a scenario and pipe to a scenario file is as follows:

./setdest –n<num of nodes> -p <pause time> -s <max speed> -t <simulation time> -x <max x axis> -y <max y axis> > <outdir>/<scenario-file>

APPENDIX F Defining Mobile Node Traffic Pattern File

The connection/traffic files are defined to specify the different traffic sources as well as when a particular node joins or leaves a multicast group. For example:

```
set udp_(1) [new Agent/UDP]
$udp_(1) set dst_addr_ 0xE000000
$ns_ attach-agent $node_(0) $udp_(1)
set cbr_(1) [new Application/Traffic/CBR]
$cbr_(1) set packetSize_ 512
$cbr_(1) set random_ 1
$cbr_(1) set maxpkts_ 100000
$cbr_(1) set dst_ 0xE000000
$cbr_(1) attach-agent $udp_(1)
$ns_ at 30.000 "$cbr_(1) start"
```

APPENDIX G Defining Fixed Nodes and Traffic Pattern in the Test Script

The fixed nodes and their traffic pattern are defined with a certain topology in the beginning of the test script for the mixed network before we define the MGW and other mobile nodes afterwards. MGW is configured as a stationary mobile ad hoc node with parameters set as described in [Cheng01] and the *wiredRouting* to be "ON". One fixed node is linked with the MGW. A fixed multicast protocol is appointed for the fixed domain as well as the MGW. All the other mobile nodes are configured with the *wiredRouting* to be "OFF" and other configurations as in [Cheng01]. A mobile node movement file and traffic pattern file need to be loaded before the simulation start. Each simulation generates two sets of trace files, one for the fixed side, and the other for the ad hoc domain.