MULTICAST ALGORITHMS FOR MOBILE SATELLITE COMMUNICATION NETWORKS

THESIS

Ryan W. Thomas, Second Lieutenant, USAF

AFIT/GCE/ENG/01M-04

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

20010706 143

AFIT/GCE/ENG/01M-04

Multicast Algorithms

for

Mobile Satellite Communication Networks

THESIS

Ryan William Thomas
Second Lieutenant, USAF

AFIT/GCE/ENG/01M-04

Approved for public release; distribution unlimited

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the United States Government.

AFIT/GCE/ENG/01M-04

Multicast Algorithms
for
Mobile Satellite Communication Networks

THESIS

Presented to the Faculty of the Graduate School of Engineering and Management

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Computer Engineering

Ryan William Thomas, B.S.

Second Lieutenant, USAF

March 2001

Approved for public release; distribution unlimited

AFIT/GCE/ENG/01M-04

Multicast Algorithms

for

Mobile Satellite Communication Networks

Ryan William Thomas, B.S.

Second Lieutenant, USAF

Approved:

Major Richard Raines Committee Chair	6 Mar 01 Date
Major Rusty Baldwin Committee Member	6 Mar 01 Date
Dr. Michael Temple Committee Member	6 Mar 01 Date

Preface

I would like to first thank my thesis advisor, Major Richard Raines, for his advice and input during the course of this thesis effort. I would also like to extend a special thanks to my committee members, Major Rusty Baldwin and Dr. Michael Temple for their guidance and knowledge.

Of course, I would also like to thank my wonderful and helpful wife, who always made time for my schoolwork. I also appreciate the encouragement and support of my family and friends.

Ryan William Thomas

Table of Contents

		Page
Preface		iii
List of Figures		vii
List of Tables		ix
Abstract		x
I.	Introduction	1-1
	1.1 Background	1-1
	1.2 Research Problem	1-2
	1.3 Scope	1-3
	1.4 Approach	1-3
II.	Literature Review	2-1
	2.1 Introduction	2-1
	2.2 IP Multicast	2-1
	2.2.1 Address Allocation	2-2
	2.2.2 Membership Management	2-3
	2.2.3 Routing Protocols	2-4
	2.3 Mobile IP	2-8
	2.3.1 Agent negotiation	2-9
	2.3.2 Packet tunneling	2-12
	2.4 Mobile Multicast	2-12
	2.4.1 Mobile IP (nomadic) Multicasting	2-13
	2.4.2 Mobile Ad Hoc Network (MANET) Multicasting	2-16
	2.5 Internet-in-the-sky	2-18

Page

2.5.1 Inter-Satellite Links (ISL's) 2-19

2.6 Conclusion 2-20

III. Methodology 3-1

3.1 Introduction 3-1

3.2 Background 3-1

3.3 Problem Definition 3-1

3.3.1 Hypothesis 3-2

3.4 System Boundaries 3-2

3.4.1 Mobile IP Boundaries 3-3

3.4.2 Ad Hoc Parameters 3-8

3.5 System Services 3-9

3.6 Performance Metrics 3-12

3.7 Parameters 3-13

3.7.1 System 3-13

3.7.2 Workload 3-17

3.7.3 Algorithm Timing Issues 3-19

3.8 Factors 3-25

3.9 Workload 3-27

3.9.1 Scaling 3-27

3.10 Experimental Design 3-30

3.11 Evaluation Technique 3-31

3.11.1 Implementation Details 3-32

3.11.2 Verification and Validation 3-34

3.12 Summary 3-35

Page

IV. Results 4-1

4.1 Introduction 4-1

4.2 Statistical Accuracy 4-1

4.3 DVMRP / Mobile IP Scenarios 4-4

4.3.1 DVMRP Low Membership 4-4

4.3.2 DVMRP High Membership Levels 4-11

4.4 ODMRP Scenarios 4-15

4.4.1 Data-to-Overhead Analysis 4-16

4.4.2 Received-to-Sent Analysis 4-18

4.4.3 End-to-End Analysis 4-21

4.5 Reliability Scenarios 4-21

4.6 Protocol Comparison 4-25

4.6.1 Data-to-Overhead Analysis 4-25

4.6.2 Received-to-Sent Analysis 4-27

4.6.3 End-to-End Analysis 4-27

4.6.4 Conclusions 4-28

V. Conclusions 5-1

5.1 Restatement of Research Goal 5-1

5.2 Research Contributions 5-1

5.3 Conclusions 5-1

5.4 Future Research 5-2

5.4.1 ODMRP Modifications 5-2

5.4.2 DVMRP Modification 5-3

Appendix A. Data A-1

Bibliography BIB-1

Vita VITA-1

List of Figures

Figure		Page
2.1.	Host State Transition Diagram [1]	2-4
2.2.	Foreign Agent Registration	2-11
2.3.	Collocated Registration	2-11
2.4.	Home Agent Registration	2-12
3.1.	Sample One, Two, Three and Four Satellite Dispersal	3-5
3.2.	ODMRP Flow Chart	3-15
3.3.	DVMRP Flow Chart	3-16
3.4.	ODMRP Timing Sensitivity	3-21
3.5.	Mobile IP Timing Sensitivity	3-21
3.6.	DVMRP Neighbor Probe Sensitivity	3-22
3.7.	DVMRP Flash Update Sensitivity	3-23
3.8.	DVMRP Prune Sensitivity	3-23
3.9.	DVMRP Packet Size Sensitivity	3-24
3.10.	ETE Delay Components	3-28
3.11.	OPNET Satellite Node Model	3-32
3.12.	Method of Determining Satellite Footprint	3-33
4.1.	Sample DVMRP/MobIP All-to-all Simulation Run	4-5
4.2.	Sample DVMRP One-to-all Simulation Run	4-5
4.3.	DVMRP/MobIP One-to-all	4-6
4.4.	DVMRP/MobIP All-to-all	4-6
4.5.	DVMRP All-to-all, High Membership, Urban Areas	4-11
4.6.	DVMRP All-to-all, High Membership, Evenly Distributed	4-12
4.7.	DVMRP All-to-all Full Comparison	4-13
4.8.	Sample ODMRP One-to-all Simulation Run	4-16

Figure Page

4.9. Sample ODMRP All-to-all Simulation Run 4-16

4.10. ODMRP One-to-all 4-17

4.11. ODMRP All-to-all 4-17

4.12. ODMRP received-to-sent Quantile-Quantile Plot 4-19

4.13. ODMRP One-to-All, Received-to-Sent Metric Bimodality 4-20

4.14. ODMRP Forwarding Group Membership 4-21

4.15. Effect of Start Time on DVMRP Satellite Failure 4-22

4.16. DVMRP Critical Failure 4-23

4.17. ODMRP Critical Failure 4-24

4.18. Effect of Seed on ODMRP Satellite Failure 4-24

4.19. Overhead comparison of ODMRP, DVMRP 4-26

4.20. Comparison of low membership level end-to-end delay metrics . . . 4-28

List of Tables

Table		Page
3.1.	Packet Sizes	3-18
3.2.	Significant Parameters Summary	3-19
3.3.	ODMRP Baseline Configuration	3-19
3.4.	DVMRP / mobile IP Baseline Configuration	3-20
3.5.	DVMRP / mobile IP Final Timing Configuration	3-25
3.6.	Mobile Node Home Locations	3-26
3.7.	Loading Levels	3-27
3.8.	Scaling Comparison	3-30
3.9.	Factors and Workloads	3-31
4.1.	Regression Analysis, data-to-overhead, DVMRP	4-13
4.2.	Regression Analysis, Loading, DVMRP	4-13
A.1.	DVMRP, All to All, Low Membership, Urban Areas	A-1
A.2.	DVMRP, One to All, Low Membership, Urban Areas	A-1
A.3.	DVMRP, All to All, High Membership, Urban Areas (Sparse)	A-2
A.4.	DVMRP, All to All, High Membership, Even Distribution (Dense)	A-2
A.5.	ODMRP, All to All, High Membership, Urban Areas	A-2
A.6.	ODMRP, One to All, Low Membership, Urban Areas	A-3
A.7.	DVMRP Failure Data	A-3
A.8.	ODMRP vs DVMRP ANOVA	A-4
A.9.	Low Membership DVMRP ANOVA	A-5
A.10.	ODMRP ANOVA	A-6
A.11.	High Membership DVMRP ANOVA	A-7
A.12.	DVMRP Failure ANOVA	A-8
A.13.	ODMRP Failure ANOVA	A-9

AFIT/GCE/ENG/01M-04

Abstract

With the rise of mobile computing and an increasing need for ubiquitous high speed data connections, Internet-in-the-sky solutions are becoming increasingly viable. To reduce the network overhead of one-to-many transmissions, the multicast protocol has been devised. The implementation of multicast in these Low Earth Orbit (LEO) constellations is a critical component to achieving an omnipresent network environment. This research examines the system performance associated with two terrestrial-based multicast mobility solutions, Distance Vector Multicast Routing Protocol (DVMRP) with mobile IP and On Demand Multicast Routing Protocol (ODMRP). These protocols are implemented and simulated in a six plane, 66 satellite LEO constellation. Each protocol was subjected to various workload,to include changes in the number of source nodes and the amount of traffic generated by these nodes. Results from the simulation trials show the ODMRP protocol provided greater than 99% reliability in packet deliverability, at the cost of more than 8 bits of overhead for every 1 bit of data for multicast groups with multiple sources. In contrast, DVMRP proved robust and scalable, with data-to-overhead ratios increasing logarithmically with membership levels. DVMRP also had less than 70 ms of average end-to-end delay, providing stable transmissions at high loading and membership levels. Due to the fact that system performance metric values varied as a function of protocol, system design objectives must be considered when choosing a protocol for implementation.

Multicast Algorithms

for

Mobile Satellite Communication Networks

I. Introduction

The Internet was initially conceived to provide a means of transferring data from one machine to another. Today, it is frequently being used as a mechanism for sending the same data to many users. In an effort to provide efficient communications, multicasting was developed to ease the pressure that duplicate message transfer place on corporate, educational, and military network bandwidth constraints.

As Internet usage becomes more ubiquitous and in demand, typical connection methods will not be able to meet the demand for either bandwidth or mobility. With the advent of Low Earth Orbit (LEO) satellite networks such as Globalstar, Teledesic, and Iridium, the potential for having a constant broadband connection to the Internet becomes more realistic.

These two technologies – mobile satellite data networks and the multicasting of information – are not isolated from one another. Mobile users will want to leverage the power of multicasting as they connect into the satellite network. Supporting this technology with the unique dynamic topology of a satellite constellation requires a careful choice of algorithms.

1.1 Background

The fundamental mechanism of routing data in the Internet is the Internet Protocol (IP). IP has a special set of extensions and rules exclusively designed for multicasting. There have been many different algorithms developed for multicast routing with the IP, but very few have been implemented. The Multicast Backbone of the Internet (MBone) serves as the largest implementation of the multicast technology. Uses and applications for this technology are still growing.

To create a satellite network that truly satisfies the demand for an omni-present Internet connection, it will be necessary to support the multicasting aspect of IP. The algorithm used to support this protocol is of special interest as the choice can have a large effect on the amount of overhead and delay a multicast packet must experience to be delivered. Implementing a mobile multicasting algorithm will give insight into the constellation properties with respect to traditional, terrestrial mobile networks.

1.2 Research Problem

The problem of implementing IP in a mobile environment has already received a large amount of attention. In particular, the mobile IP protocol outlines a system to provide reliable IP support to mobile nodes on a network of fixed routers and topology. As an off-shoot of this research, the problem of multicasting in a mobile network has also been examined. In particular, a mobile multicasting protocol has already been submitted in a Request-For-Comment (RFC) format by the Internet Engineering Task Force. This RFC is in addition to several application-specific mobile multicasting protocols presented by the academic community.

The mobile IP problem has several interesting analogies to the Internet-in-the-sky problem. From the view of a LEO constellation, movement on the earth is insignificant when examined relative to the high speed satellite motion. For this reason, transmitting and receiving nodes appear fixed. Having mobile routers (satellites) and fixed nodes is really the inverse of the mobile IP problem which featurd mobile nodes and fixed routers.

Another branch of mobility research has focused on the idea of "ad-hoc" networks. These networks consists of rapidly changing topologies of independent units operating as both routers and nodes. While the primary focus of this research has been Multicast over IEEE 802.11 networks, there is again a definite analogy to the dynamic nature of a satellite constellation.

The application of these mobile multicasting protocols to an Internet-in-the-sky could provide a way to implement the multicasting portion of the IP protocol over the dynamic

topology of a LEO constellation. This research will simulate various multicast protocols in a LEO constellation in order to gain insight into their effectiveness.

1.3 Scope

This thesis examines system performance resulting from the implementation of two very antipodal protocols for LEO multicast communication, On Demand Multicast Routing Protocol (ODMRP) [2] and Distance Vector Multicast Routing Protocol (DVMRP) [3].

To determine the protocols effectiveness, the amount of overhead, effective number of packets received, and the time to deliver the packets are examined. These metrics are examined under various workloads, membership levels, geographic positioning, and satellite failure configurations. This research was conducted using a computer simulation environment allowing both an economical and practical method of determining which protocol is more appropriate.

1.4 Approach

This thesis presents several existing mobile multicast protocols and examines the implementation features that distinguish them. From these protocols, two representative protocols are chosen based on features that may be of potential interest in a LEO constellation. The methodology for examining these protocols is then discussed and implemented. Finally, the results of these experiments is presented.

This work expands on research into unicast routing protocols in a LEO network by Pratt [4], Fossa, [5] and Janoso [6]. It also furthers work done by Muller [7] into the behavior of mobile IP and multicasting.

The objective of this thesis is to perform a comparative analysis of mobile multicasting algorithms in a discrete-event simulation environment. Following in the Pratt's work [4], the the algorithm's performance in a damaged constellation is examined by performing a critical satellite failure analysis.

II. Literature Review

2.1 Introduction

This chapter examines the IP multicast standard, and associated routing algorithms. The Mobile IP standard is then discussed, along with particular ways that the mobile nodes negotiate and communicate with foreign networks. These two concepts are then combined to explain requirements for implementing a mobile multicasting system. Finally, the inner workings of various Internet-in-the-sky solutions are presented, concentrating on how information is potentially routed through these networks.

2.2 IP Multicast

IP multicast is a solution for one-to-many communications over the Internet. Traditionally, the Internet has been set up to communicate in a one-to-one (unicast) fashion. A unicast system requires a node to send individual messages for every recipient it wishes to communicate with. This works well until a single message needs to be sent to n nodes rather than just one. In this case, n copies of the message must be sent. This approach wastes bandwidth and resources associated with the communications system.

Multicast routing, on the other hand, sends a single message out per link, instead of sending a copy for each node accessing the information on the link. This single copy is then reproduced and branched to individual nodes wishing to receive the message by a multicast router, instead of by the sender. Broadcasting in this manner provides an efficient, lower network load solution to the one-to-many communication problem.

While multicast is not yet widely used, the applications for which multicasting can provide benefits are numerous. Some of the most commonly mentioned multicast applications include video-conferencing, shared workspace, distributed interactive simulation, software upgrading, and resource location [8].

In 1992, Deering [1] presented what is known as the "standard model" for IP multicasting. It includes a set of requirements that dictate how potential multicast protocols must behave and is summarized as follows:

- IP-style semantics. Multicast messages are to be based on User Datagram Protocol (UDP, a best-effort policy) and require no scheduling or registration before sending. This allows a source to transmit at any time.
- Open groups. Sources are not to be limited in quantity or scope. In other words, sources can come from outside the group, and there can be any number of sources.
- Dynamic groups. Group members can join and leave at will. They do not need to "register, synchronize, or negotiate" with a central manager.

To accomplish multicast as defined above, there must first be a manner to distinguish multicast messages from unicast messages within the IP addressing scheme. Secondly, nodes wishing to be a part of a multicast group must have a mechanism to notify to their router whether or not they are members. Finally, there must be an algorithm for efficiently routing and delivering the multicast messages to the nodes [8].

2.2.1 Address Allocation. Multicast addresses are part of what is known as the class D address group. A class D address is defined as one having `1110` as its highest order bits. Since IP addresses are in the dotted decimal format, this means that multicast host groups range from `224.0.0.0` to `239.255.255.255`. The first 256 multicast addresses are reserved for exchanging of low-level protocol information such as routing data or topology discovery and maintenance. Even with reserved addresses, the addressing space allows for over 250 million potential multicast groups. The Internet Assigned Number Authority is in charge of assigning multicast addresses to organizations and corporate groups. They are also charged with maintaining and assigning the 256 reserved addresses [9].

Unlike unicast routing in which an IP address is statically bound to a single Local Area Network (LAN) interface on a single IP network [1], multicast IP addresses are dynamically bound to several interfaces on several networks. This means that the routers do not need to maintain a list of individual nodes that are group members group. Instead, they only maintain a list of local members.

One thing that is not contained within the multicast address is the scope of the multicast group. There is a field in the IP packet, however, that controls this. Known as the Time-To-Live (TTL) field, it is decremented at every node that the packet passes

through. When the field reaches zero, the packet is destroyed. By setting the TTL field, messages can limited in scope [10].

2.2.2 Membership Management. Membership management is handle by the Internet Group Management Protocol (IGMP). There are two different versions of IGMP: IGMPv1 as described by Deering in RFC 1112 or IGMPv2 as described in RFC 2236 [11, 1]. IGMPv2 adds some amount of refinement to the process but the two are inter-operable in function. IGMP works by allowing nodes to report their memberships to any immediately neighboring routers and by enabling routers to learn about group membership on their attached networks. It does so by using a query and response called ROUTER DISCOVERY and ROUTER ADVERTISEMENT.

The query message is called ROUTER DISCOVERY. There are two forms of ROUTER DISCOVERY: the general query and the group specific query. The general query is used by routers to find out which groups have members on the attached network. A general query receives a response for every group represented on the network. The specific query is used to determine if a particular group has any members on a network. A specific query receives a response only if the group requested in the query is present in the network. The response in both cases is called ROUTER ADVERTISEMENT.

There are three states that a node can be in when using the ROUTER DISCOVERY and AGENT ADVERTISEMENT protocols. These node states and transitions are summarized in Figure 2.1. The simplest state is the *non-member state.* This state is left only when a node joins the group.

Nodes with group membership that have not been contacted with a ROUTER DISCOVERY message are in the *idle state.* If a node leaves the group, it transitions to the *non-member state.* If the node is contacted by a ROUTER ADVERTISEMENT, it sets a timer and enters the *delaying member state.*

The *delaying member state* is made up of member nodes contacted by Router Discovery messages but that have not yet replied. The timer is used to wait before reponding to a ROUTER ADVERTISEMENT and ensures the responses are spread out and so not to flood the subnet. This state is only left if one of three scenarios occur: the timer expires and the

node responds to the ROUTER ADVERTISEMENT, the node leaves the group, or another node on the subnet transmits a ROUTER ADVERTISEMENT duplicating the delaying node's Router Advertisement [1]. The last case is used to conserve bandwidth by eliminating extra ROUTER ADVERTISEMENTS.

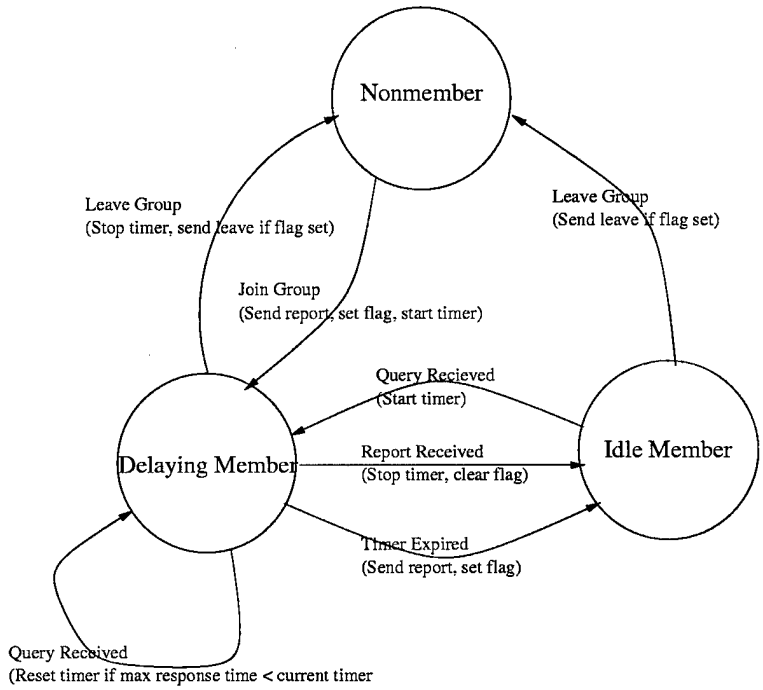

Figure 2.1. Host State Transition Diagram [1]

2.2.3 Routing Protocols. Once nodes have subscribed to a group, there must be a routing protocol that allows for the transmission of data from source to receiver. To be of any real value, multicasting must be efficient and scalable. For networks without uniform multicasting ability, the protocol must also be incrementally deployable [8].

Efficiency, in the context of multicasting, means the route must be maintainable with a minimum number of control messages. Scalability is the ability for a protocol to maintain a linear expansion of these control messages as the network grows. These requirements have led to the creation of several algorithms, each of which has particular strengths and weaknesses.

The Distance Vectoring Multicast Routing Protocol (DVMRP), Protocol Independent Multicasting (PIM-DM), Multicast Open Shortest Path First (MOSPF), and Core

Based Tree (CBT) protocols are representative of routing algorithms developed to work with multicasting. They represent the three major categories of algorithms: source-based trees with no a priori topology knowledge, source-based trees with complete topology knowledge, and core-based trees.

2.2.3.1 Distance Vector Multicast Routing Protocol (DVMRP). DVMRP has the advantage of being the first algorithm used to route multicast messages. Being the first, it was designed to be both incrementally installed into the existing Internet infrastructure and works with non-multicast compliant routers. DVMRP was derived from the Routing Information Protocol (RIP), a unicast method of routing that calculates the best next hop to use in order to transmit a packet to a destination. DVMRP uses a source-based multicast tree to pass packets to their destinations. When a source wishes to send a multicast message, the network is flooded with the first packet in such a way as to form a tree. The routers then "prune" off unnecessary tree branches, i.e. those with no receivers on them [3]. This flooding and pruning is called Reverse Path Forwarding (RPF) and is the key to designing source-based trees without any prior network state knowledge.

The flooding part of the RPF algorithm works as follows: when a node receives a packet, the DVMRP routing table is used to determine if the link that the packet came in on is the best way to get to the source node. If it is, then the node is forwarded on all out-going links except the one it arrived on. If it isn't, then the packet is destroyed. This ensures the tree only contains the shortest paths from source to destination [3].

Next, IGMP is used to prune branches by querying to see if there are any subscribers on each of the branches. If there are no subscribers, a prune command is sent to the upstream router, instructing it not to forward multicast packets to this branch. If all branches leaving from a particular node have been pruned, that node forwards a prune message down the tree from itself, completely cutting itself off from the group.

If a node on a previously pruned branch decides to subscribe to a multicast group, it is "grafted" on by sending a cancel-prune message to the upstream nodes. This allows for a tree to change without completely starting the process over.

Periodically, however, the tree may need to be refreshed because of changes in the topology and membership. The refresh mechanism involves a repetition of the above process. This flood and prune technique negates some of the bandwidth benefits of multicasting. For groups with many sources, this is a significant burden because of the high network and memory utilization needed to maintain many source tree routes [8].

2.2.3.2 Protocol Independent Multicasting - Dense Mode (PIM-DM). Similar to DVMRP, PIM-DM is an enhanced RPF algorithm. Therefore, it incorporates a flood and prune mechanism to generate trees dynamically. However, unlike DVMRP, it does not have a built-in mechanism of determining the unicast routing table used to choose the best incoming interface for a given source. Instead, it utilizes any underlying unicast routing mechanism that is available.

The advantage is that the unicast routing tables are not constrained to the DVMRP or other specific mechanism of finding the best source-based routes. The protocol can simply use what already exists. Unfortunately, not all methods provide as much information as the RIP-like method does in DVMRP. As a result, some prunes and grafts that might have been avoided with additional information may occur under PIM. The protocol creators determined that this was a fair trade-off for having the additional flexibility that no underlying unicast routing mechanism provided [12]

2.2.3.3 Multicast Open Shortest Path First (MOSPF). While DVMRP incorporated a RIP-like route discovery mechanism, MOSPF uses the Open Shortest Path First (OSPF) protocol. OSPF is a unicast routing mechanism that keeps track of the state of every other router in a specified region. Using OSPF data, nodes are capable of performing flood and prune messages internally. The savings in bandwidth and protocol complexity, however, is added as a computational burden.

OSPF routers communicate the state of the network through Link-State Advertisements (LSA). These messages propagate to all MOSPF routers within the OSPF routing area, and then are stored in a link state database at each router. When a multicast data packet arrives at a MOSPF router, this data is accessed and the router computes where the

packet should be sent using Dijkstra's algorithm. This route is cached and only updated when a new LSAs arrive [13].

Unfortunately, in highly active and dynamic groups, the number of calculations required to maintain the group becomes tremendous. Not only do the routers need to recalculate the routing table whenever there is a change in the set of receivers, but routers are required to create a new source-based tree for every new source that begins broadcasting to the group. Additionally, any dynamics in the set of receivers produces a large amount of LSA traffic to track these membership changes [8].

2.2.3.4 Core Based Trees (CBT). CBT was developed to address specific weaknesses in MOSPF and DVMRP routing. It eliminates the need for DVMRP to periodically flood and prune and for MOSPF to maintain the routing state for every source of every group in every node. However, CBT does not contain some of the multicasting advantages of these protocols.

CBT depends on a "core" router that is the focus of all group traffic. This core router has no responsibility other than to join the tree together. Beyond that use, it is identical to every other node in the routing tree. When a new member wishes to join a group, it sends a join message to the core router along the shortest path from itself to the core. It uses any unicast routing table to do this and is thus protocol independent (unlike MOSPF and DVMRP which are each dependent on a specific routing protocol).

When the join request finds its way to a node already on the tree, a join-acknowledge is sent in reverse order back to the new member, and the intermediary router makes a state entry recording the member's incoming and outgoing interfaces. It does not matter whether the join message makes it all the way to the core router, only that it find its way to a group member [14].

Since CBT uses a shared instead of a source-based tree, traffic is focused over fewer links. This results in increased bandwidth utilization over fewer links and thus, higher potential for service drop out. Furthermore, determining which node to use as the optimal center of the tree is an NP-complete problem that is dependent on the complete knowledge of the network topology and group membership. Thus, even though CBT does not require

complete topological knowledge at each router like MOSPF, it does require this knowledge to be held somewhere so that an optimal core can be chosen. For these reasons, CBT only makes sense for relatively static groups [8].

2.3 Mobile IP

When making the transition from wired networks to wireless networks, there is a basic change in operating assumptions. The most obvious is the simple fact that nodes are no longer physically constrained to stay connected to one another at all times. The topology of the network becomes dynamic instead of static. Other differences include a rapidly changing physical layer, characteristically high bit error rates, and router firmware that may or may not support certain routing schemes. These problems have been addressed at the network layer by the Mobile IP standard.

While it is possible to solve some of the above stated problems in the link or physical layers, only at the network layer can a solution be devised that is medium independent and widely scalable. Two of these more narrowly scoped solutions exist in the IEEE 802.11 protocol (a wireless LAN system) and CDPD (Cellular Digital Packet Data, a cell phone system) [9].

Mobile IP is primarily designed to deal with mobile networks operating on the fringe of a fixed network. It assumes that this fixed network has the capacity to route or tunnel IP packets across it. Furthermore, being stacked somewhere between the transport layer and the IP network layer in the OSI model, it is not concerned with how these routing mechanisms work. Mobile IP is only designed to maintain a nomadic node's connection with its fixed home Internet location.

Mobile IP is described in RFC 2002 through 2006 [15, 16, 17, 18, 19], and collectively these documents present a system that allows nodes to maintain a permanent IP address wherever they go. To accomplish this, there are three primary players in Mobile IP [15]:

- The mobile node. This is a node that has the ability to change the link used to connect into the network and still maintain communication over a permanent IP address.

- The home agent. This is a router that can access the mobile node's home link as determined by the mobile node's permanent IP address. The home agent is responsible for:

  1. Knowing the mobile node's current location,
  2. Advertising its reachability in the mobile node's absence, and
  3. Intercepting and forwarding packets destined to the mobile node to the mobile node's care-of-address.

- Foreign agent. This is a router that can access the mobile nodes current link. It is responsible for:

  1. Helping the mobile node alert its home agent of it's care-of-address,
  2. If necessary, providing a care-of-address, receive packets from the home agent, and send them to the mobile node, and
  3. Routing packets generated by the mobile node to their appropriate destinations.

The care-of-address mentioned repeatedly in the agent roles is just what it sounds like. Like a vacationing tourist might have his or her mail forwarded by a servant to a temporary address he or she is staying at, mobile nodes have their traffic forwarded to them via a care-of-address. The three players' responsibilities are centered around negotiating for that care-of-addresses and then using it.

2.3.1 Agent negotiation. The first step in agent negotiation is coined "agent discovery" and is the process the mobile node uses to determine whether or not it is connected to the its home network or a foreign network [9]. At the same time, the mobile node determines whether or not it has moved since the last time it performed agent discovery. During this state that the mobile node also negotiates for a care-of-address when is connected to a new foreign link.

There are two messages that nodes and agents use to discover each other. These messages are based on the ROUTER ADVERTISEMENT and ROUTER DISCOVERY messages used by IGMP. AGENT ADVERTISEMENTS are broadcast or multicast periodically by home

and foreign agents to announce their capabilities and identity much like ROUTER ADVERTISEMENTS are broadcast periodically through IGMP to announce group membership. Impatient mobile nodes can use AGENT SOLICITATIONS which request all local agents to transmit AGENT ADVERTISEMENT messages [9].

Mobile nodes determine if they have moved since they last received an AGENT ADVERTISEMENT by two methods. The first method uses the lifetime field found in the AGENT ADVERTISEMENT packet. The lifetime field indicates how long the mobile node should wait to hear another AGENT ADVERTISEMENT message before it assumes that it out of the agent's range. Since AGENT ADVERTISEMENTS can (and do) get lost, RFC 2002 recommends the advertisements be broadcast approximately three times as fast as the lifetime field. If a mobile node receives no advertisement inside of the lifetime field, it assumes that itself has moved.

The second method is interpreted from the network prefixes. In an AGENT ADVERTISEMENT there is an optional field called the prefix-length extension. The prefix-length field indicates the number of leftmost bits of the router address field that need to be taken at a time to determine the network prefix of the foreign agent. The mobile node can then extract the prefix-length extension from the AGENT ADVERTISEMENT and then back-calculate the network prefix. If the prefix is different from the previous AGENT ADVERTISEMENT, the node can conclude itself has moved, otherwise, it assumes both advertisements were received on the same link. A node cannot simply use the source address field since it is possible to have multiple agents per router, and thus there can be different source addresses on the same link.

Once the node determines that itself has moved or the previous registration's lifetime is about to expire, the node registers. However, this only occurs if the node has heard an AGENT ADVERTISEMENT and knows what agent to register with. Like the two messages used when performing AGENT DISCOVERY, there are two messages used with registration: the REGISTRATION REQUEST and REGISTRATION REPLY messages.

A mobile node transmits the REGISTRATION REQUEST based on which of the three states it is in. The first state is when the mobile node is on a foreign network and will be

using a foreign agent's address as its care-of-address. The second state is when the node is on a foreign network and will be using a collocated address (i.e. an available address that the network saves for visitors) as its care-of-address. The last state is that the node is on the home network and will not be using a care-of-address.

The first state works as follows: when a foreign agent receives the REGISTRATION REQUEST, the agent examines it and, if the agent is able to perform as requested, the agent relays the message to the home agent. If everything is correct when the packet is received by the home agent, the home agent updates the mobile node's bindings and then sends the REGISTRATION REPLY to the foreign agent. If the foreign agent finds nothing wrong with the reply, it relays the reply to the mobile node which then runs its own set of checks on it. If there is a problem at any point in this chain of events, the mobile node receives notification and can attempt to request registration again after fixing the problem. If, for some reason, the mobile node receives no reply at all, it repeats the process, increasing the interval between requests [9]. This is illustrated in Figure 2.2.

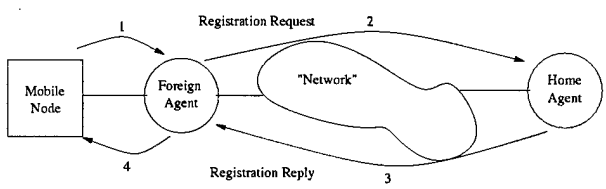

Figure 2.2. Foreign Agent Registration

The second (collocated address) state works similarly to the first. Instead of replying to the foreign agent, the home agent replies directly to the mobile node. The foreign agent is completely ignored, as the mobile node receives its collocated address through means other than the agent. This is illustrated in Figure 2.3.

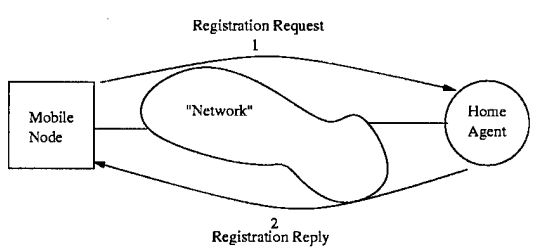

Figure 2.3. Collocated Registration

The third possible state is a specialized case of the second, i.e. the mobile node acts like it has a collocated address when on its home network. This becomes more important once messages start getting transferred or tunnelled between the home agent and the mobile node. This process is illustrated in Figure 2.4.

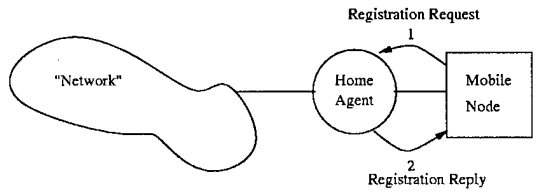

Figure 2.4. Home Agent Registration

2.3.2 Packet tunneling. Tunnelled packets are IP packets that have been placed inside of another IP packet. This is known as IP-in-IP encapsulation. While encapsulation provides an effective manner of transmitting packets from a home agent to mobile node, it adds a considerable amount of complexity and overhead to the transmission [16].

When a packet is sent to a mobile node, it is intercepted by the home agent and then tunnelled to the mobile node. Whether it is tunnelled directly to the node or to the foreign agent depends on whether or not the node is collocated. If the node is collocated, the tunnelled messages are sent directly to the mobile node. If not, they are sent to the foreign agent who de-tunnels the packets and sends them to the mobile node [9].

In particular, if the Maximum Transfer Unit (MTU) (the largest size of IP packet permitted on a network) is exceeded by the overhead that encapsulation requires, it becomes necessary to fragment the packets into multiple parts for reassembly upon reception. This causes additional overhead. Furthermore, since the MTU value is dependent on the networks over which the fragments are sent, fragmentation may occur repeatedly over the path of a transmission.

2.4 Mobile Multicast

Multicast over mobile networks presents special problems that are not found in fixed networks. The severity of these problems depends on the model used to describe the network.

Up to this point, the mobility approach has been consistent with the view the Internet community has taken towards the mobile node problem, i.e. a view maintaining that mobile nodes are primarily nomadic hosts that are roaming away from their fixed-address domain space [20]. This view also assumes the core network functions (e.g. routing) are performed using pre-existing routing protocols inside a fixed network topology.

Another model used for analyzing mobile networks assumes the networks consist of autonomous nodes that are independent of any fixed infrastructure. These networks are called Mobile Ad hoc Networks (MANETs) for the method that they create and destroy infrastructure amongst themselves through wireless connections. The primary difference between a MANET and a nomadic network is the MANET's lack of a fixed network topology to utilize. Another difference is that MANET's are usually assumed to be completely wireless networks, utilizing an IEEE 802.11 or some other wireless data transfer layer.

In either model, assumptions have changed from the fixed network case. Both models assume that the mobile node has a dynamic network address and has changing points of attachment to the network. Both models assume that a node could act as both a host and router, while the MANET depends on the fact that it will.

2.4.1 Mobile IP (nomadic) Multicasting. The most common multicast protocol following the nomadic network model builds off the framework set by the Mobile IP standard. Since these protocols assume they are being built on top of Mobile IP, they share all the assumptions of Mobile IP. Most significantly this means that no attention is paid to actual hop-to-hop routing, instead, effort is expended on address management and protocol interoperability.

2.4.1.1 Collocated. The collocated method of multicast group subscription, sometimes also called remote subscription, requires a mobile node to unsubscribe from a multicast group when moving and then resubscribe to that multicast group upon arrival at a new foreign network [21]. The foreign router must able to act as a multicast router for this to work. Since multicast addresses do not represent a physical address, there is no need to use a home agent with a permanent address to forward messages. The multicast

address can be found no matter where the mobile node roams. However, to use the foreign router for multicast reception and transmission it is necessary that the mobile node to use a collocated address so that the multicast routing algorithms can transmit and receive appropriately.

With highly dynamic nodes, however, it is possible that there can be a significant amount of multicast loss caused by the setup and transition time. Additionally, this protocol assumes that there is a collocated address available at every foreign network; something that is not always true in practice. Finally, this method is built on top of a fixed network multicast protocol. Many of these protocols, such as DVMRP, do not scale well to dynamic groups [22].

Despite these shortcomings, Muller determined that the collocated solution provides the best path efficiency of any nomadic network solution [7]. Collocated effectively implements the fixed-network multicasting protocols in a nomadic mobile network since, unlike other implementations, it requires no additional intermediary nodes or agents.

2.4.1.2 Tunneling. If there is no collocated address available or interruption of multicast service is a major concern, it is possible to use tunneling to provide mobile multicast [21]. For this to occur, the home agent must first be configured as a multicast router that it can manage the mobile nodes' multicast subscriptions.

Having the home agent act as a multicast router means that the transmission and reception of multicast messages occurs through that agent. The home agent tunnels multicast receptions to the foreign agent which in turn de-tunnels the packets and hands them to the mobile node. Transmission happens in the reverse: the node generates packets and sends them to the foreign agent which tunnels the message to the home agent. The home agent de-tunnels the packets and sends them to the multicast group. Tunneling now requires extra overhead as they must now be doubly encapsulated to make up for the ambiguity created from multicast IP addressing [10].

When a home agent receives a multicast message intended for the mobile node, it encapsulates the message so it is addressed specifically to the mobile node, since the multicast subscription address (which has no physical location) is unknown to the foreign

agent. The packet is encapsulated a second time for tunneling to the foreign agent. This double encapsulation increases the amount of overhead and can cause fragmentation.

There are other drawbacks to tunneling. The greatest is commonly referred to as the tunnel convergence problem. Tunnel convergence is caused by multiple home agents tunneling multicast messages to mobile nodes connected to the same foreign agent. This produces packet duplication which negates one of the major reasons for implementing a multicast scheme in the first place[22].

Additionally, there is the question of defining of 'local' when IGMP performs a message query. For example, if an IGMP message queries a mobile node connected to a home agent in LAN A, the mobile node will hear it and reply, even if it is currently tunneling through a foreign agent in LAN B. This can cause scoping confusion, since in the tunnelled multicast case, the TTL field will not prevent messages from being delivered outside of their intended range.

Muller also examined tunneling and a variant, called minimal encapsulation tunneling, and found neither provided adequate efficiency for a multicast system. Particularly, the problem of tunnel convergence can increase agent loading by a factor of 20 over the remote subscription protocol [7].

2.4.1.3 Mobile Multicast (MoM) Protocol . Mobile Multicast (MoM) attempts to address these problems of locality and tunnel convergence by making some modifications to the tunneling multicast protocol [10, 22]. The first change MoM introduces is the Designated Multicast Service Provider (DMSP), which is designed to eliminate the tunnel convergence problem. When multiple home agents begin forwarding duplicate multicast messages to a foreign agent, the foreign agent nominates one (or two for redundancy) to act as the DMSP for the particular group.

When the mobile node leaves a foreign agent or unsubscribes from the group, another home agent is passed the responsibility of being the DMSP. Harrison et al., simulated the protocol with different metrics for choosing the DMSP. They used the criterion of handoff count, route optimality, and fairness were used to determine the optimal DMSP selection rule [10]. When handoff count (number of changes in DMSP over a period of time) is

measured, temporal strategies such as oldest mobile node and oldest home agent proved to have the lowest counts. On the other hand, when route optimality was examined, the spatial strategies of choosing by closest home agent proved to have the lowest hops per route. Finally, in the class of fairness (defined to be how evenly is the DMSP forwarding task distributed among the home agents in the network), all algorithms were found to be acceptably fair. The exception was for the spatial methods which placed an unfair burden on those home agents closest to the majority of the foreign agents.

The MoM approach showed marked improvement in scalability over regular tunneling. If N is the number of networks in the system, G is the number of forwarded multicast groups, c is the average number of mobile hosts at a network and k is the number of redundant DMSPs forwarding per group, then the big 'O' of $NumberOfMessages = O(c \cdot N^2 \cdot G)$ for tunneling and $NumberOfMessages = O(k \cdot N \cdot G)$ for MoM [10]. The number of redundant DMSPS cannot be larger than the number of visiting mobile nodes, i.e. $k \leq c$. Thus, the number of MoM messages is at least an order-of-magnitude less than in a similar sized network that utilizes the regular tunnelled approach.

2.4.2 Mobile Ad Hoc Network (MANET) Multicasting. There has been a significant amount of research into unicast MANET protocols [23], many providing specialized optimizations in areas such as power consumption. Some of these unicast protocols have provide rudimentary multicast capabilities but overall, multicast protocols have lagged behind. These specialized protocols are primarily grouped into two categories: a mesh-based architecture and a tree architecture.

2.4.2.1 On Demand Multicast Routing Protocol (ODMRP). ODMRP is a mesh-based approach to routing [2, 24]. Meshes are advantageous to trees in ad hoc networks because they are able to deal with many of the problems of mobility. The capacity to handle intermittent connectivity and uneven traffic density are two of the more significant advantages. It also is an on-demand algorithm, meaning it builds routes between nodes only as required by source nodes.

When a multicast source has a message to send (and there is no route determined from this source to the group yet), it sends out a request in the form of JOIN QUERY. The JOIN QUERY is made up of a table with fields. The first field is the source's IP address followed by the node address of the query's last hop. In the case of the originator, the last hop is the source IP address again. The next field is the sequence number which is designed to differentiate packets from one another. Finally, a TTL field is configured to prevent the message from being propagated outside the multicast domain.

The JOIN QUERY is sent to all neighbors of the source. Each neighbor records the source and sequence of the packet, as well as the last hop IP address. The node then decrements the Time-To-Live field of the packet and checks whether it is greater than 0. If it is, it forwards the query to its neighbors; otherwise, the packet is destroyed. The node finally examines whether it wants to subscribe to the group. If it does, it creates and transmits a JOIN REPLY [2].

The JOIN REPLY packet is made up of the packet's current IP address and the next-hop IP address (gleaned from the last-hop record made upon receipt of the JOIN QUERY). If there are multiple queries the node is responding to, they are all grouped into one JOIN REPLY packet, which is sent upstream.

When a node receives a JOIN REPLY packet, the packet is examined to see if the address is located under the next-hop field. If it is, the node goes back into its records and finds the next-hop address for that particular source. The node itself then creates a JOIN REPLY and sends it upstream. At this point, the node becomes a forwarding node for the multicast group. A flag is set by the node to indicate it should forward this group and records the next-hop address. Like the node that requested to receive the group, this routing node groups together JOIN REPLIES before broadcasting upstream to save bandwidth.

When the JOIN REPLY message reaches the source, the source continues to use the designated route until a predefined route expiration timer elapses. The process then repeats itself [24]. Note that each source will generate new routes, potentially adding to existing routes to form a mesh.

Lee et al. determined ODMRP has significantly lower channel and storage overhead than DVMRP as well as the capability to exploit redundant paths – something none of the other protocols allow. Additionally, future enhancements incorporating positional information may allow for more efficient route determination by decreasing the frequency of route refresh [25].

2.4.2.2 Ad Hoc Multicast Routing (AMRoute). Whereas ODMRP was a mesh-based protocol, Ad Hoc Multicast Routing (AMRoute) is a tree-based protocol [26]. The protocol works by connecting disjoint group members with bi-directional tunnels, creating a tree through frequent updates.

At the beginning of the tree construction, each node declares itself a "core" of one member (itself) for the subscribed multicast groups. Periodically, the core floods the the links around it with JOIN REQUEST messages. If another core belonging to the same group receives the JOIN REQUEST from a core with the same group membership, it marks the core as a neighbor and replies with a JOIN ACKNOWLEDGE. The acknowledging core also marks the other core it acknowledges as a neighbor. This creates a mesh of same group cores.

To form a tree, each core transmits a TREE CREATE packet to its neighbor cores. When a core receives a non-duplicate TREE CREATE message, it marks the sender as a member of the tree and forwards the TREE CREATE message on all outgoing mesh links. If a duplicate TREE CREATE is received, a TREE CREATE NAK (Not ACKnowledge) is returned to the sender.

Lee et al. found AMRoute had a high packet delivery ratio for low rates of mobility. However, at higher rates of mobility, the overhead of recreating the tree structure did not scale well. AMRoute performed well under moderate load, but failed at higher traffic loads [25].

2.5 Internet-in-the-sky

Internet-in-the-sky is a phrase used to describe the concept of creating a TCP/IP compatible network in a satellite constellation. This can either be used as part of the

larger Internet when routing between nodes or exclusively, depending on the destination and source. Most Internet-in-the-sky solutions focus on Low Earth Orbit (LEO) satellite technology for propagation delay and availability reasons.

Geostationary (GEO) satellites are named such because they do not change in position relative to the earth. However, GEOs cannot provide coverage above $\pm$ 70 degrees due to their positioning over the equator. Additionally, since GEO satellites are placed at higher altitudes (36,000 km), the time required for a transmission to propagate from source to destination can easily exceed 600ms [27]. Power is also a consideration as the distance between earth surface and satellite can cause considerable power loss. Higher power requirements dictate the satellites be larger and heavier, increasing cost and decreasing the expected lifespan.

LEO satellites, on the other hand, orbit close to the earth in a band between 500 to 2000 km [28]. This minimizes the power and propagation problems associated with the GEO satellites. LEO satellites are not stationary however, and so it takes many LEO satellites to have constant coverage of an earth location.

The large number of satellites is not as much of a drawback as it may appear. Because of the closer proximity to earth, LEO satellites can be smaller and thus launched in multiples, a strength for commercialization of the technology. Additionally, the large satellite count permits higher cell densities than provided by sparser GEO constellations.

Since LEO satellites are not fixed, the topology of a LEO constellation is constantly changing. The links between satellites (termed Inter-Satellite Links or ISL's) are constantly changing. This makes routing for non-Ad Hoc protocols in a LEO constellation particularly difficult.

2.5.1 Inter-Satellite Links (ISL's). Janoso [6] and Pratt [4] looked at dynamic routing techniques in satellite constellations utilizing ISL's and specifically specifically examined the Extended Bellman Ford and Darting algorithms.

The Extended Bellman Ford algorithm works by taking the traditional Bellman Ford algorithm and removing the counting-to-infinity and bouncing behavior. The counting-to-infinity behavior was determined by Cheng et al.[29] to be caused by routers advertising

routes to neighbors that go through the advertised neighbor. An added field in the routing table allows for the detection of these "non-simple paths."

The Darting algorithm [30] uses data packets to transmit topology information thereby cutting down on overhead. Each data packet contains all recent topological updates and are thus passed on to successive nodes as the packet is routed. Darting compares the topological viewpoints of a node and its predecessor (where the packet came from). If there is a difference, it updates the the predecessor through a special packet containing the correct, updated topological data. Since each node now has the most updated view of the topology, it can use these tables for routing in conjunction with a modified Dijkstra algorithm.

The Dijkstra algorithm alone is used as a baseline by many studies for its simplicity and ease of implementation. However, it is not as robust as Darting or Extended Bellman Ford since it it does not account for how topological information is shared. For this reason it is not considered a practical algorithm for implementation [4].

Janoso determined the Extended Bellman Ford algorithm provided better performance with smaller overhead. When Pratt modelled Darting with complete topological knowledge, it was able to converge to an optimal solution that outperformed Extended Bellman Ford for a limited number of earth stations.

2.6 Conclusion

In this chapter, the current research and developments in multicasting and mobility support were examined. Focusing primarily on the various implementations of mobile multicasting with Mobile IP and in a MANET, these protocols were examined and critiqued.

To implement a true Internet-in-the-sky solution, it is necessary to support multicasting. The relative performance of the aforementioned protocols with respect to a satellite constellation compared to traditional multicasting and mobile multicasting is currently unknown. Research is needed to determine both their appropriateness and performance for satellite-based systems.

III. Methodology

3.1 Introduction

This chapter describes the methodology used to define and answer the thesis problem. From the problem statement to the actual design of the experiments, this chapter considers the full range of the thesis development process.

3.2 Background

The literature search examined previous research in mobility solutions and multicasting algorithms. However, these methods are designed for disparate views of mobile networks. Mobile IP focuses on a mobility solution in which mobility is the exception, whereas the ad hoc approach looks at mobility from the viewpoint that rigidity is the exception and mobility the rule.

Research into unicast routing schemes in Low Earth Orbit (LEO) satellites has been examined by several groups [4, 5, 30, 29, 6]. There is, though, a noticeable absence of research that investigates the behavior of multicast algorithms in LEO satellites. Because of the unique characteristics of these constellations, it is not immediately obvious what method of multicasting should be implemented.

As satellite constellation research shows, LEO satellite Inter-Satellite Links (ISLs) are constantly changing as they fly over the earth's poles. However, outside of these areas of rapid change, the ISLs remain relatively fixed. Additionally, the ISL changes are deterministic and can be can calculated for any point in time. The contrast of these areas of changing topology against the presence of the relatively fixed sections brings the system to a classic engineering design trade-off. Whether the system behaves more like a fixed network with mobile nodes or an ad hoc network with relatively fixed inter-node links depends on what paradigm the constellation is viewed under.

3.3 Problem Definition

The focus of this research is to compare the performance of two multicasting protocols, one representing the ad hoc school of thought and the other the nomadic point of

view. These protocols are examined under various group membership, density and loading levels as well as in the presence of satellite failures. In particular, the mobile IP multicasting protocol is compared against the On Demand Multicast Routing Protocol (ODMRP) ad hoc multicasting protocol.

3.3.1 Hypothesis. Each of these protocols comes with a trade off. The mobile IP protocol has the advantage that it is built on top of the IP multicasting protocol and is therefore independent of underlying multicasting mechanisms. This indicates the protocol only performs as well under nodal failures as the underlying protocol. It also implies that as the membership and density levels increase, this protocol will only scale as well as the underlying routing algorithm.

On the other hand, mobile IP multicast has a very high overhead associated with moving since individual nodes are given the responsibility of maintaining group membership as transitions occur. Mobile IP multicast should therefore have worse performance as loading levels and mobility increase, causing the overhead of subscribing and unsubscribing from groups to become more of a bottleneck to receiving group messages.

ODMRP is different from mobile IP in that it has little additional overhead associated with mobility since the system assumes that a node is in motion. This indicates as mobility levels increase, ODMRP will perform better than mobile IP. However, increased loading and membership may cause scaling problems under ODMRP because of the messaging that must occur as each additional member joins and as traffic increases. The mesh-based paradigm the protocol uses should create a resilient fabric against satellite failure.

It is anticipated that at low traffic levels, ODMRP will outperform mobile IP. However, as group traffic increases, mobile IP's fixed architecture will contain less overhead than ODMRP's mobile architecture, allowing it to perform better.

3.4 System Boundaries

The system under test (SUT) in this research is the entire LEO satellite network and the accompanying mobile nodes. These nodes are assumed to be fully mobile, i.e. capable

of changing geographical location at any time. However, this does not require the nodes be modelled as mobile. Section 3.7.2 discusses this in more detail.

The rest of this section discusses the boundaries of the SUT as they apply to the two algorithms under test. These boundaries are different than parameters and factors because they define the domain where the parameters and factors exist.

3.4.1 Mobile IP Boundaries. Mobile IP has two major multicast components that typically are determined by services available in real world systems. The first component defines how the system handles multicast subscriptions. This determination is based on whether the foreign router is multicast compliant and if there is a co-located address available. The second component is the underlying multicast routing mechanism. This is usually determined by the multicast group routing algorithm physically implemented in the LANs and WANs

3.4.1.1 Subscription Management. Mobile IP has two documented mechanisms for performing multicast subscription management of multicasting nodes – co-located and tunnelled (Mobile Multicast (MoM) [10] is a special case of the tunnelled management mechanism). When using tunneling, the question facing the system designer is where to place home and foreign agents. The LEO satellites examined in this study are designed to route on-board data sent to them from the ground stations. Therefore the foreign agent would naturally be the satellite in the ground node's field of view. As the LEO constellation changed overhead or the ground node moved, the foreign agent would change.

The home agent location presents a different problem. There are basically two locations where the home agent could be placed: on a stationary ground node or on a satellite. Ideally, Mobile IP's habit of "dog-legging" (the routing effect that occurs when a message is tunnelled away from the most efficient route because of its need to go through an agent) would be minimized by whatever choice made.

By generalizing the situation that occurs when a home agents is presumed to be on-board a satellites or when the home agent is presumed to be on an earth stations, it is possible to examine the problem mathematically. This mathematical model makes

the following assumptions. First, that the locations of the home agents are randomly distributed. Whether this distribution is contained within the satellite constellation or around the earth depends on the placement choice for the home agents. Second, mobile nodes are not restricted and can potentially use any satellite as their foreign agent. Third, order of placement has no effect on the combinations, ie. placing the agent "A" on a satellite and then "B" is the same as placing "B" and then "A." Thus, this is a combinatoric problem.

Basing the home agents on satellites leads to 66^4 possible arrangements of foreign agents and home agents. Four of these arrangements are illustrated in Figure 3.1. Box 3.1a represents a two mobile nodes sharing one satellite. Box 3.1b represents the two satellite dispersal with an IP tunnel between. Boxes 3.1c and 3.1d are sample configurations involving three and four satellites, respectively.

Placing the home agent on earth creates a slightly different picture, since a foreign agent and home agent can never share the same node. This is because the home agent is not on a satellite and the foreign agent is. However, there are still four possible arrangements.

To analyze how often each of these will occur in a fully mobile network, one must determine how many of these arrangements occur in a constellation of size n satellites. For the home agent located in a satellite scenario, the number of combinations, c_i, of locations of the four agents on i different satellites is expressed in Equations (3.1) through (3.4) and illustrated in Figure 3.1. Their are $\binom{n}{i}$ ways to choose i satellites from a constellation of n. After these i satellites have been chosen the number of ways to place the four agents on the satellites with all satellites having at least one agent is calculated. The four satellite case, Figure 3.1d, has 24 different ways to places the four agents on four satellites.

The three satellite case, Figure 3.1c places two agents on the same satellite. There are 36 different combinations of where to place the agents for this configuration on the satellites. The two satellite case must co-locate three agents on one satellite or two agents on two satellites. The first case is presented in Figure 3.1b. There are 14 ways to layout the satellites in these configurations. The one satellite case has only 1 combinations, since all four agents must share one satellite.

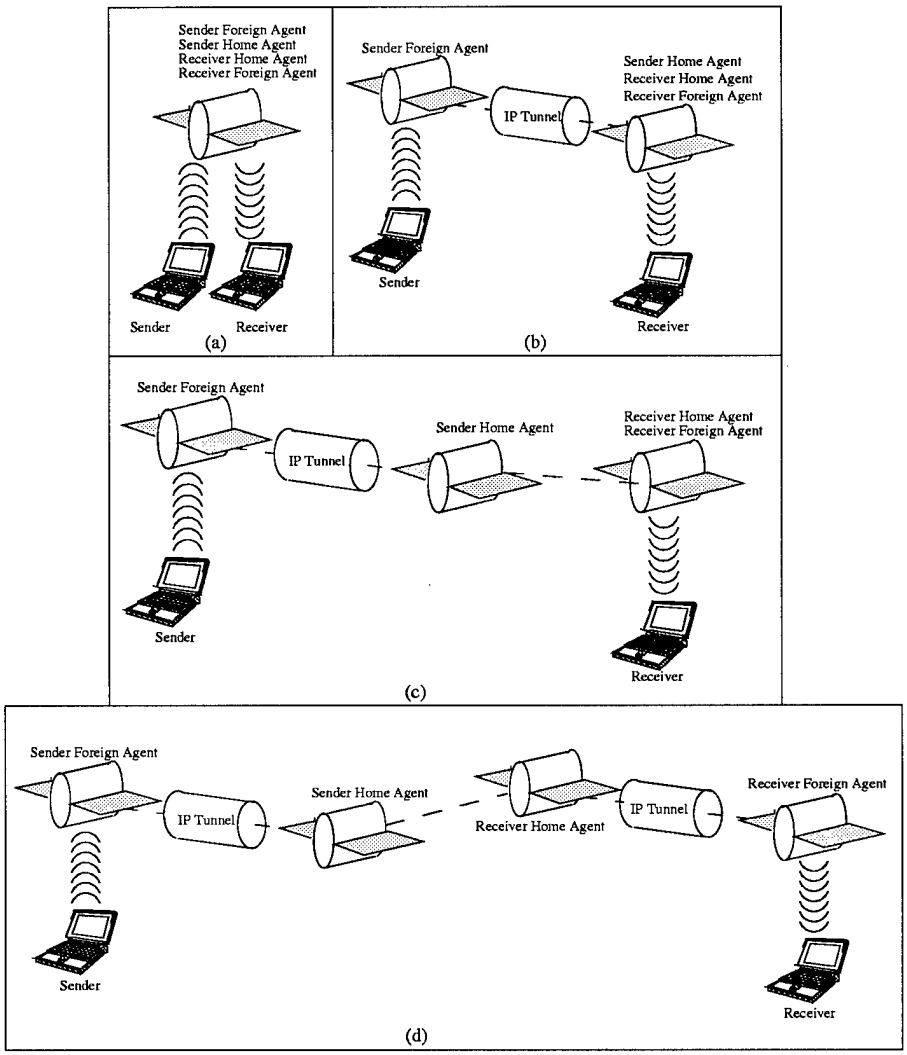

Figure 3.1. Sample One, Two, Three and Four Satellite Dispersal

$$c_4 = 24\binom{n}{4} \tag{3.1}$$

$$c_3 = 36\binom{n}{3} \tag{3.2}$$

$$c_2 = 14\binom{n}{2} \tag{3.3}$$

$$c_1 = 1\binom{n}{1} \tag{3.4}$$

Thus, the total number of combinations is

$$\sum_{i=1}^{4} c_i = 66^n \tag{3.5}$$

For a satellite constellation of size 66, this means 91% of all satellite distributions are those that have all four agents on separate satellites – the group of combinations that has the most amount of dog-legging.

Chuah et al. [31] found file transfer over mobile IP took up to 90% more time to complete than a non-mobile IP transfer. Of that, 90% of the dog-leg routing overhead accounted for 76% of the delay. The Chuah study did not incorporate propagation delay and the authors felt dog-leg routing overhead would increase even more dramatically in networks where propagation was a major issue.

Muller [7] concurs with this, finding the hop-ratio (ratio of number of hops taken to number required) of bi-directional tunneling was slightly greater than three. Thus using tunneling has several undesirable attributes. Many, if not all of these can be eliminated by choosing mobile IP's other alternative for multicasting, the co-located approach. This is also supported by Muller's research, which found little overhead (hop ratio of one) generated by the local subscription model.

There are, however, two major caveats to being able to implement the co-located approach. The first is the router on the mobile node's foreign link must support multicasting. In a closed system specifically designed for Internet such as an Internet-in-the-sky, it only makes sense the on-board routers be multicast enabled. The second is there must be enough co-located addresses on the server. Given that a satellite has a finite number of channels that can support a limited number of users at a time, it would be realistic to assume each satellite would have enough co-located addresses to handle the maximum number of simultaneous users.

3.4.1.2 Routing Protocol. The second design choice in mobile IP is to determine the underlying routing protocol used by the constellation. Ideally, this choice would be representative of other routing algorithms. The routing protocol will impact the

overall performance of the mobile IP solution since the routing choice has a tremendous effect on the performance of non-mobile networks.

Multicast Extensions to Open Shortest Path First (MOSPF) is dependent on OSPF to function. The implementation of OSPF in a satellite network is a difficult proposition. Maintaining the state information required of each node by MOSPF is computationally heavy, especially given the changes in the satellite topology and the number of nodes in a LEO system. Finally, MOSPF is not widely deployed and there is little data for model validation, hindering any simulation efforts.

Core Based Tree (CBT) multicasting is not adept at handling dynamic networks because of its reliance on a single core node to be reachable at all times. Additionally, to even be somewhat useful, CBT must place its core node nearest to the main source. This could only occur if the network connection between the main source and the core node remained static.

Distance Vectored Multicast Routing Protocol (source-based tree without prior knowledge) makes sense for the satellite constellation because it dynamically creates and updates the multicast tree. DVMRP has already been field tested in the MBone, the multicast backbone of the Internet. However, the route finding algorithm is composed of a Routing Information Protocol-like (RIP) mechanism. Furthermore, RIP has its basis in the Bellman-Ford routing algorithm. Pratt found Bellman-Ford was not an optimal way to construct a unicast routing algorithm in a satellite constellation [4]. His simulation showed the Darting algorithm outperformed the Bellman-Ford algorithm under moderate to high loads.

Protocol Independent Multicast-Dense Mode (PIM-DM), which is very similar to DVMRP, appears to be an ideal choice. Similar in most aspects of operation to DVMRP, PIM-DM does not require DVMRP's use of a Bellman-Ford unicast route discovery mechanism. Instead it can use any unicast mechanism. An implementation of PIM-DM might be able to combine PIM-DM with the Darting algorithm [4, 6] to create an ideal multicasting protocol for mobile IP. Additionally, since unicast traffic would already be traversing the network, utilizing the routing information associated with it would be at no additional

cost to PIM-DM. However, the system boundaries do not include a unicast system, and thus implementing one would be outside the scope of the simulation. Furthermore, this unicast-multicast overlap would also make it difficult to compare PIM-DM against other protocols.

For these reasons, DVMRP is the best choice for the underlying multicast protocol. It presents a complete multicast solution and is widely implemented. It is also close enough in operation to PIM-DM to allow insight into this class of algorithm.

3.4.2 Ad Hoc Parameters. Since ad hoc networks are typically smaller in scope than mobile IP, there are fewer parameters to be defined. However, there are multiple ad hoc multicasting protocols. Determining which one to use and what part of the SUT to incorporate it into are parameters that need to be defined.

3.4.2.1 Ad Hoc Multicasting Protocol. The ad hoc multicasting protocol chosen should ideally be the best performing of the alternatives. Lee et al. examined the performance of On Demand Multicast Routing Protocol (ODMRP), AMRoute, Ad hoc Multicast Routing Protocol with Increasing id-numberS (AMRIS) and Core Assisted Mesh Protocol (CAMP). ODMRP consistently had among the best metrics in the workloads of mobility speed, number of senders, group size, and traffic load [2].

Additionally, ODMRP presents a mesh-based approach which is of particular interest for its ability to handle multiple routes. The ability for this approach to overcome satellite failure is of interest to the research. For these reasons, ODMRP is an excellent candidate to represent the ad hoc protocol.

3.4.2.2 Ad Hoc Placement. Once ODMRP has been chosen as the ad hoc multicasting protocol, it is necessary to determine where the ad hoc network begins and ends. There are two places where the boundary can be drawn: either including just the satellite constellation or both the constellation and the ground stations.

Defining the edge of the ad hoc network at the satellite/ground station boundary means each satellite is responsible for determining the multicast groups for each connected mobile node, and then utilizing ODMRP to send and receive messages on their behalf.

On the other hand, defining the boundary at the ground station/rest of Internet requires propagating all ODMRP messages to the ground stations. These stations are treated as being in the same class as the nodes in the rest of the network. The satellites are considered as a large forwarding group, since the stations on the ground do all the actual subscriptions and thus multicast receiving and transmitting.

The first option places the ODMRP solution into the same class as the co-located mobile IP solution. Not having the ground stations as part of the ad hoc network forces the ground stations to use some sort of Dynamic Host Configuration Protocol-like (DHCP) mechanism to register and deregister with the satellites as they changed coverage. ODMRP then becomes just another routing protocol, much like the underlying protocol of mobile IP. While the ISL mesh does have some of the properties of an ad hoc network, the most significantly ad hoc component comes from the fringe of the network where ground nodes are joining and leaving the constellation. Utilizing ODMRP as simply a routing protocol instead of an ad hoc manager is a waste of the overhead put into ODMRP to handle the ad hoc aspect. Additionally, the mechanism for having ground nodes register and deregister with the satellites is a repetition of services already potentially provided by ODMRP. Therefore the ad hoc boundary will include both the satellite and the ground stations.

3.5 System Services

Broadly speaking, the SUT provides only one service to the user. This service is the transmission of packets from a given sender. In particular, the constellation should be able to transmit unicast, multicast and broadcast packets. This research focuses on multicast packet transmission.

Despite the fact there is only one service of interest provided by the constellation, there are many possible outcomes for the packet that is transmitted. The packet could be:

- delivered (contents: correct, receiver: correct)
- delivered in error (contents: don't care, receiver: wrong)
- delivered with error (contents: wrong, receiver: correct)

- not delivered (contents: don't care, receiver: none)

To provide the multicast transmission service, several other services are required. In particular, multicast transmission depends on routing and mobility management services. While there may be other ways to define these sub-services, this particular division reflects the services provided by mobile IP and ODMRP.

The outcomes of the lower level service are no different than those at the higher level. When the previous outcomes are applied to the lower level services, they take on several sub-outcomes. However, not every outcome at the higher level is represented by these lower level services.

Packets given to the multicasting routing algorithm can be:

- delivered
  - with time delay. The routing algorithm may choose a multicast route that minimizes the amount of time it takes for the data to transfer from sender to receiver.
  - with hop efficiency. Much like the time delay outcome, there exists many routes a routing protocol can choose. One or more routes may contain the minimum number of hops from source to receiver. The number of hops a route takes in comparison with the minimum defines the route efficiency outcome of a delivered packet.
  - with overhead. The amount of information required to send a given packet of data determines this sub-outcome of the SUT. A delivered packet may be set with little control data, or may be only a small fraction of the total stream of data.
- delivered in error or not delivered due to
  - an inaccurate view of topology. A routing scheme must have a current and updated view of the network topology to make routing choices. If the information is old, it may deliver the packet in error or not at all.

- a locality problem. When a packet escapes its Time To Live (TTL) constraint, it will be destroyed, losing the packet.
- a traffic blockage. Another outcome like hop efficiency and time delay. An algorithm can choose to route over low traffic or high traffic links and routers. A router with high traffic may drop packets. While the router services are not part of either the routing or mobile management service, their usage is part of the routing service.
- with low-level errors. When the link or physical layers introduce errors into packets, a packet can be mis-delivered or dropped. As lower level errors are outside the scope of the SUT outcomes and will not be considered further.

- delivered with error due to
  - tunneling or fragmentation error. If a route forces the transfer of packets across a link that require packet fragmentation or tunneling, fragmented or tunnelled packets can be lost from earlier outcomes and result in errors in the original packet payload.
  - low-level errors. Other than the two previous outcomes, a routing mechanism should be independent of the data it is routing. Therefore, any further errors are lower-level and again will not be considered.

Compared to the routing service, the mobility management service has a much smaller subset of possible outcomes. This follows from the scope of the mobility management service. The mobility management does not depend on the routing service. However, the routing service assumes the mobility management is handling mobile nodes and updating the topology accordingly.

Packets sent by mobile nodes through the mobility management service are:

- delivered, delivered with errors
  - These outcomes are not effected by mobility management.
- delivered in error or not delivered due to

- unrecognized arriving nodes. If newly arrived nodes are not recognized, packets will not be delivered to them.
- unremoved departed nodes. If departed mobile nodes are not removed from a connection, then packets will either continue to be sent incorrectly to the old location or they will be delivered correctly or in error.

3.6 Performance Metrics

The aforementioned services and their outcomes provide insight into what performance metrics to use during evaluation. Ideally, each of the outcomes will be represented in the metrics to allow insight into the effectiveness of the services provided.

Loss ratio or received to sent ratio are two complimentary ways of reporting the ratio of packets delivered to those which are transmitted. These two metrics allow an investigation into the packets not delivered and delivered in error outcomes. Since packets delivered to either the wrong node or not at all are counted as lost packets, this metric blurs the distinction between the two outcomes. This is useful in that it acts as a Quality of Service (QoS) metric by separating the desirable outcomes from the undesirable ones. In this manner, the loss ratio acts as a representation of the overall effectiveness of the protocol. Loss ratio is defined as the total number of packets sent by all multicast sources multiplied by the number of multicast receivers. This number is divided into the total number of packets received uniquely by all receivers.

Mean delay is a metric that exposes the packet delivery outcome. While a metric such as hop count or hop ratio (ratio of hops taken to minimal number of hops) might be useful here, the constellation is not composed of equidistant ISLs and hop count could be a misleading metric. Since the distance between satellites at the sixty degree latitude mark is half the distance at the equator, this means the number of hops is not necessarily proportional to the amount of delay. Additionally, since satellites are so far apart, the mean delay is a better indicator of the effectiveness of the routing service. Mean delay is simply the average delay of all packets received by all receivers.

The ratio of effective data bits sent on the network to overhead bits sent on the network is another effective metric of measuring the packet delivered outcome. Unlike mean delay, this control ratio is representative of the overhead each protocol requires. As such, this metric provides insight into the other metrics by showing how much work is required to achieve the outcome. Data bits are defined to be bits that are successfully transmitted from source to receiver. Overhead bits are all other information transmitted over the system, including data bits that do not reach their destinations.

The ratio of data bits to overhead bits is examined system-wide. It is calculated by first counting the total of all bits that are transmitted in the system from any source. This is the net network traffic. Next, when packets are received at a destination, the packet id, size, and all hop IP addresses are recorded in a master list. When other multicast group destinations receive the same packet, they update only the new hops received at that location into the master list. At the end of the simulation, the size of the packet and total number of hops are combined to determine the total data bit traffic. This is subtracted from the net network traffic to determine the total overhead traffic, and a ratio is taken of the total data traffic and the total overhead traffic.

3.7 Parameters

There are many parameters that affect the performance of the SUT. They can be organized into two categories: the system parameters, those that are static from one instance to another, and the workload parameters, or those that can be varied by the user at any point.

3.7.1 System. The number of satellites (or network routers, since each satellite contains router capacity for the constellation) and the number of ISLs between these satellites determines the overall network topology. The distance between satellites is dependent on the number of satellites in the constellation, so this parameter can be lumped in with the topology parameter. For this SUT, the topology is fixed to a 66 satellite, 6 plane constellation. Each plane will consist of 11 satellites and be in near-circular orbit. Co-rotating planes are spaced 31.6^o apart, and counter-rotating planes are spaced 22° apart. These

values were chosen because they are similar to values used in the Iridium system, allowing for some amount of model validation.

Related to the network topology parameter is the number of users per satellite. A typical configuration of 48 spot beams with 80 users per cell is used [32]. The data rate is 2.5 gigabits per second on up and downlinks as well as the ISL's. This is consistent with potential rates in satellite data systems.

Processing time per packet is assumed to be negligible. Thus queueing occurs in the transmitter modules, which is distributed with some general density dependent on the channel data rate. Since packet creation time is assumed to be exponentially distributed, this indicates a M/G/1 system.

Also related to the number of processors on a node is the length of the queue. Pratt set the length of the queue at 4000 packets to eliminate those that would have to wait longer than 400ms [4], a voice-data requirement. Since the voice requirement is not an issue for this implementation, the queue length will be set to infinity. This parameter allows for the scaling to be implimented as discussed in Section 3.9.1. Additionally, by creating an infinite queue, a priority is being placed on the ratio of received to sent over the end-to-end delay.

Finally, the multicast algorithm in use (ODMRP or mobile IP utilizing DVMRP) is the last system parameter. Implementation issues make it necessary to examine each protocol in further detail. Figure 3.2 represents the process followed by ODMRP as the algorithm was implemented in the simulation.

There are two "events" in the ODMRP process. The first event is the expiration of the route at the source. When the route expires, the source assumes it is no longer valid and transmits JOIN QUERY packets to the network until it receives a JOIN REPLY.

The second event is the expiration of the forwarding group members that make up the route from source to destination. According to the ODMRP protocol, the forwarding group members should expire at a slower rate than the route expires at the source. When a forwarding group member expires, it does not forward data packets again until it receives a JOIN REPLY with its address in the Source-Next field.

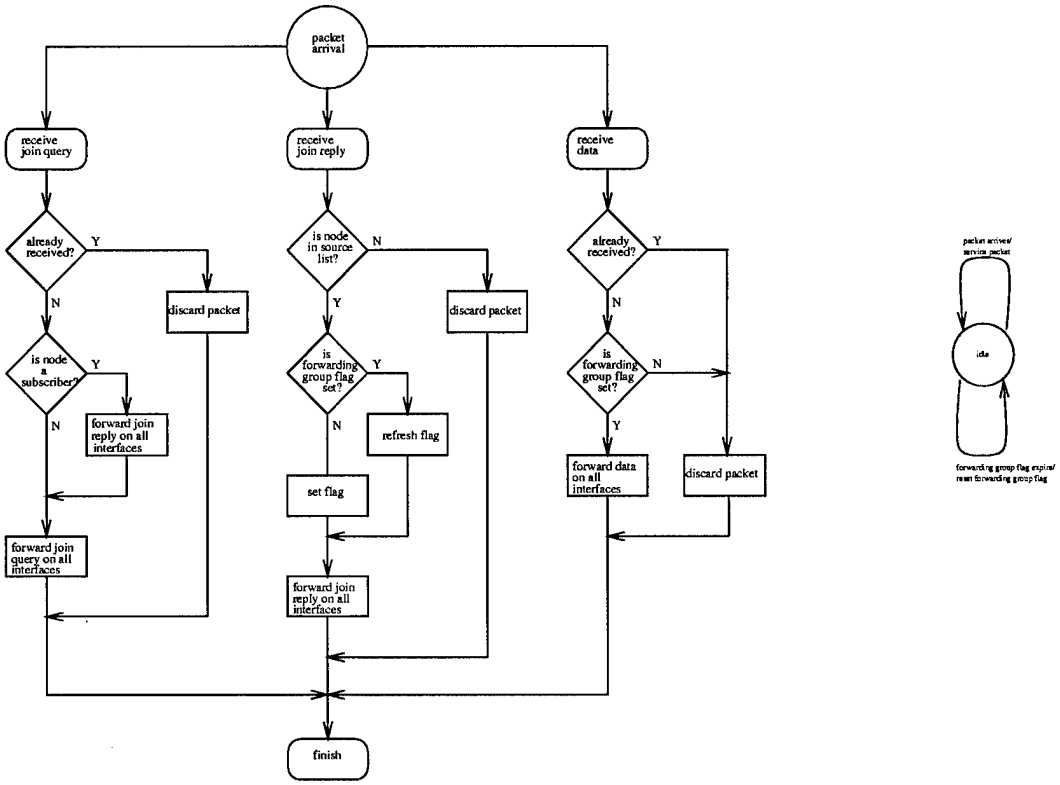

Figure 3.2. ODMRP Flow Chart

Figure 3.3 shows the logic of the DVMRP algorithm as implemented in the simulation. It is interesting to note the algorithm does not use the IGMP protocol. Instead, the simulation uses packets transmitted as an IGMP-like message. If the node receiving the packet does not have any children and does not subscribe to the group the packet is addressed to, then it prunes itself from the tree. This makes the DVMRP algorithm more "on-demand" like ODMRP. It also removes a number of factors from the DVMRP algorithm, since the parameters of IGMP did not have to be determined or optimized for the satellite network. Additionally, removing IGMP has little effect on the results since the metrics of interest are primarily those having to do with the performance of the routing protocol. It can be assumed a well-tuned IGMP protocol would be no more reliable than the data packet based protocol used here.

There are many events in the DVMRP-mobile IP process. They can be placed into four groups by functionality. The first group is the mobile IP events. These events consist of the rate of AGENT SOLICITATIONS, length of wait to hear from REGISTRATION REQUESTS,

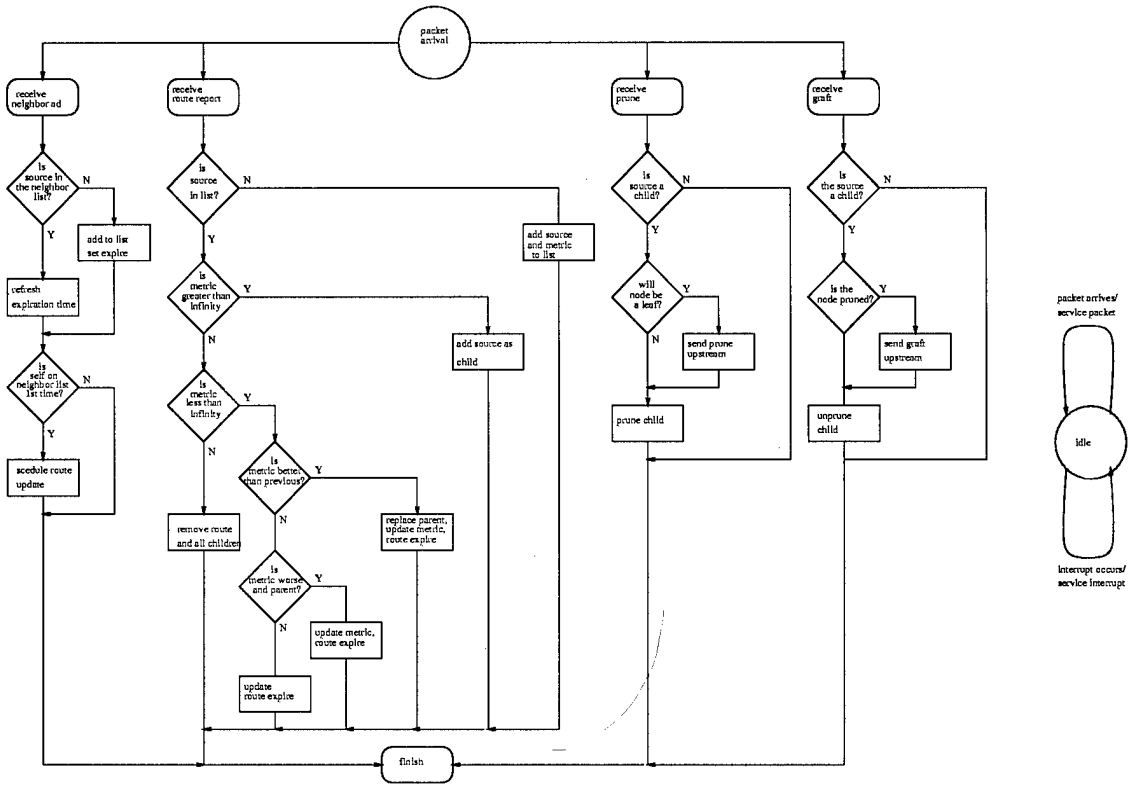

Figure 3.3. DVMRP Flow Chart

co-located expiration time, and registration expiration time. The solicitation rate controls how often a mobile node polls for available satellites. When a mobile node sends a request to an AGENT ADVERTISEMENT satellite, there is a maximum length of time the node will wait to hear the REGISTRATION REPLY. The satellite agent has a set length of time the ground node can be co-located without hearing a AGENT SOLICITATION from the ground node without dropping the node. Finally, the last event is the amount of time a mobile node stays registered with the satellite agent without hearing a AGENT ADVERTISEMENT.

The second group is the DVMRP neighbor discovery events. There are two events that make up this group: the rate at which neighbors are probed and the length of time a node keeps a neighbor without hearing a probe from it. The length of time a neighbor keeps a neighbor without hearing from it must be larger than the inter-transmission time of the NEIGHBOR PROBE packets.

The last two groups are events that have no logical partners. The first is the waiting time to transmit new routing information to the neighbors. This transmission is accomplished through FLASH UPDATE packets. The second is the waiting time for a node to prune itself from a tree after determining it should prune. These factors and those mentioned above will be examined in further detail in Section 3.7.3

3.7.2 Workload. The average number of packets sent per second and the interarrival distribution of these packets is a workload parameter. Packets are generated exponentially distributed interarrival times, for both ease of simulation and mathematical tractability. The packet arrival rate is based on the maximum packet per second rate calculated in Section 3.9.

The next parameter considered is packets length. Control packets have the fixed size and structure as shown in Table 3.1. Since the intended function of the network is to transmit data packets, the size of the data packets should be representative of typical IP network traffic. Studies by [33] and others have shown packet lengths are not typically normally or uniformally distributed. There is typically a primary mode of packets that are small, with a large "tail" of packets extending up to the Maximum Transfer Unit (MTU) of the network.

Packet traces at [34] indicate the mean packet size of traffic over the NASA Ames Internet eXchange (AIX) is approximately 400 bytes with a standard deviation of approximately 500 bytes. Research by [35] and others argue the packet length distribution is heavy tailed where heavy tailed is defined to be

$$P[X > x] \sim x^{-\alpha} \text{ as } x \to \infty \tag{3.6}$$

and $0 < \alpha < 2$. The Weibull and Pareto distributions are, by this definition, heavy tailed distributions. However, fitting the AIX data to one of these distributions proved problematic since raw data (size versus time data) was unavailable.

Since packet size is a discrete value, the geometric distribution (the discrete equivalent of the exponential distribution) is used to determine the number bits in a data packet.

While it is possible for packet sizes to contain no data and be of the header size, the minimum size of significant occurrence reported by [34] is 44 bytes (of which 20 bytes are IP header). Thus, the mean data size was chosen to be 376 bytes, and the packet size minimum 44 bytes. This gives a standard deviation of 375 bytes.

Table 3.1. Packet Sizes

		Base Size (bits)	**Extended Size**
DVMRP	Data	160	$24 + geometric(0.00266)$
	Graft	288	0
	Graft Ack	288	0
	Prune	352	0
	Probe	224	$32(NumNeighbors)$
	Route Update	224	$72(NumRoutes)$
Mobile IP	Agent Solicit	160	0
	Agent Ad	384	0
	Registration Req	420	0
	Registration Rep	384	0
ODMRP	Data	160	$24 + geometric(0.00266)$
	Join Query	224	0
	Data	192	$64(NumSourceNext)$

The number of multicast receivers, transmitters and groups is all related under the network traffic parameter. The number of receivers and transmitters is assumed to be static during the simulation. The number of messages sent per second is also assumed to be the same for all the senders and is defined by the workload in use.

For sake of simulation time, only one multicast group is simulated at a time [7]. By eliminating the interaction between multicast groups it is possible to focus the research on only the routing performance. Although this interaction between multiple groups may affect performance, it would not be a result from the routing protocol. Any performance change would be a result of limitations in the SUT congestion handling.

Included in network traffic parameters are the rate and distance at which these users move. These levels specifically affect the mobility aspect of the multicasting algorithms. However, unless the participants move out of the satellite footprints faster than the satellites change footprints, the mobility will not be noticed. The average connection time between a user and a satellite for this design is ten minutes [36], meaning a user will have

to move extremely fast to make a contribution to the system mobility more than what is caused by the orbital mechanics.

Table 3.2 presents a summary of all parameters defined up to this point, and reveals the parameters analyzed in the next section as experimental factors.

Table 3.2. Significant Parameters Summary

System	Number of Users per Satellite	3840
	Queue Length	∞
	Number of Satellites	66
	Algorithm	Factor
	Number of Available Satellites	Factor
Workload	Packet size	$24 + geometric(0.00266)$
	Number of Groups	1
	Mobility	0 m/sec
	Number of Members	Factor
	Group Density	Factor
	Transmitted Packets Per Second	Factor

3.7.3 Algorithm Timing Issues. There are many timing parameters in the protocols themselves. To minimize the number of factors in the system, pilot studies were conducted to analyze the sensitivity of the protocol's performance to each group of timing factors. For each protocol, a baseline configuration was chosen based on the work of [2] for ODMRP or [15] and [3] for DVMRP/mobile IP. The configurations are found in Tables 3.3 and 3.4.

Table 3.3. ODMRP Baseline Configuration

Forwarding Group Timeout	300 sec
Route Timeout	100 sec
Packet Size	fixed, 460 bytes
Inter-arrival Rate	exponential, mean 1 pps

These baseline settings were changed one grouping at a time. To change a grouping, the initial ratio was kept, but components were increased and decreased from the initial settings. For instance, the ODMRP group of route timeout and forwarding group timeout were initially at a 100:300 or 1:3 ratio. By decreasing the forwarding group timeout to 30, but maintaining the ratio at 1:3, the effect of the factor's length was investigated. By

Table 3.4. DVMRP / mobile IP Baseline Configuration

Time Between AGENT SOLICITATIONS	5 sec
REGISTRATION REQUEST Timeout	10 sec
Collocated Address Timeout	20 sec
Registration Timeout	10 sec
Neighbor Probe	10 sec
Neighbor Timeout	35 sec
Flash Update	5 sec
Prune Update	10 sec
Graft Retransmission	5 sec
Route Expire	140 sec
Prune Expire	600 sec
Packet Size	fixed, 460 bytes
Inter-arrival Rate	exponential, mean 1 pps

changing the ratio to 2:3, the effect of each factor on the other was investigated. Although no conclusive findings can be made from these studies, an idea of how sensitive the metrics are to each factor was determined.

Each pilot simulation was executed for one hour of simulation time, with the same random seed. While many random seeds would allow for true confidence intervals and variance, using the same random seed allowed for an "apples to apples" comparison. With a fixed seed, each scenario ran identically except for those changes implemented in the particular run.

The only ODMRP grouping is the ratio of route timeout to forwarding group timeout. The results of this study are presented in Figure 3.4. Two ratios were examined: 1:3 and 2:3. For the large part, each performed identically, tracking or overlaying the results of the other. However, in graph (b) of Figure 3.4 the 2:3 ratio begins to fall off significantly from the 1:3 ratio. The cause of this is most likely attributable to the fact that the 2:3 ratio is not an even divisor. In other words, only once every three expirations do both the forwarding group and the route time-out at the same time. This leads to "gaps" in which there are a sub-optimal number of forwarding group members in the network. This, in turn, leads to packet loss. At faster timeout rates, the rate of packet generation is relatively close, and few packets are lost during the gap. As the timeout rates decrease, the gaps increase in length and more packets are lost. Regardless, at a forwarding group

timeout of 150 seconds, the 2:3 ratio performs equally or better than all metrics at the 1:3 ratio. Thus a 100 second route timeout and 150 second forwarding group timeout will be used in the final simulations.

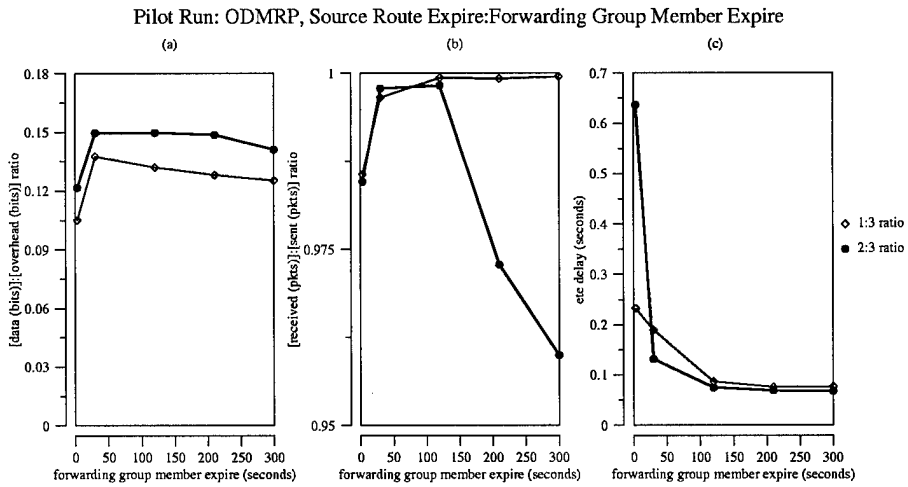

Figure 3.4. ODMRP Timing Sensitivity

The mobile IP group of timing parameters is the first of the DVMRP/Mobile IP protocol. Figure 3.5 shows for the most part, the two ratios of co-located expire/register expire/solicit time/wait expire track very closely. Smaller times seem to yield better metrics. A choice of 4:2:1:2 with a 10 second co-located timeout, 5 second wait timeout, 2.5 second solicit delay, and 5 second registration timeout yields acceptable results.

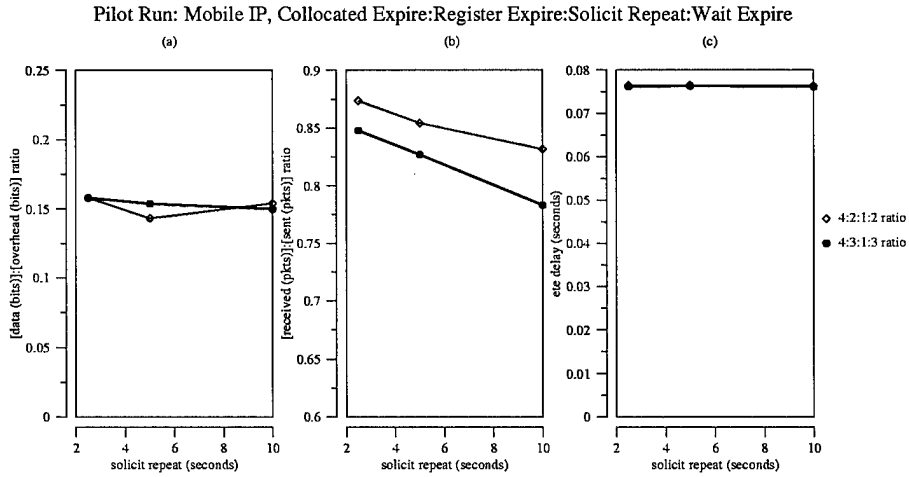

Figure 3.5. Mobile IP Timing Sensitivity

In Figure 3.6, the ratio of neighbor probe to neighbor expire is examined. Again, the two different ratio values, 2:7 and 4:7, track each other fairly closely. While the system does seem to respond to this timing group as the expiration times increase, its results are expected. Shorter time between neighbor probes and expirations allow for the protocol to converge on new topologies more rapidly. This in turn leads to fewer lost packets, but retains a higher overhead from the the more frequent probing. As the probes become longer in-between, the number of lost packets increase, causing more overhead, until this overhead becomes greater than the overhead savings of sending out fewer probes. The 2:7 ratio with neighbor expiration at 91 seconds and neighbor probe at 26 seconds seems to be the most ideal settings, as it maximizes the data to overhead ratio without compromising the neighbor expire or end-to-end delay greatly.

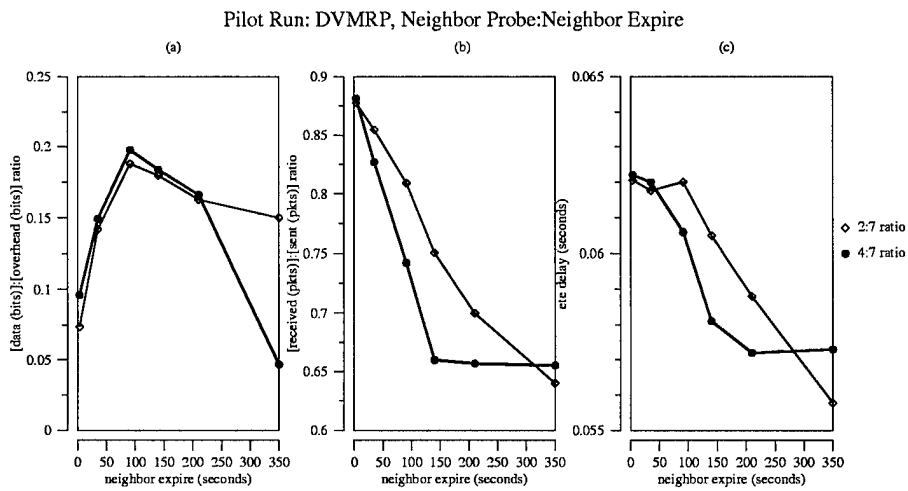

Figure 3.6. DVMRP Neighbor Probe Sensitivity

The next two timing parameters are singular, as they have no logical pairings. The flash update study shown in Figure 3.7 shows the system is relatively insensitive to changes in the flash update timing. However, the flash update does appear to cause contradictory behavior. In 3.7a, increasing the amount of time between flash updates creates increasing data to overhead ratios(0.15 to 0.18), yet in 3.7b it yields decreasing received to sent ratios (0.87 to 0.83). This is caused for reasons similar to those in Figure 3.6. The increased packet generation causes more overhead, but allows the system to adjust more quickly to

changes in topology and lose fewer packets. It appears that a value of 10 seconds provides performance benefits in each metric.

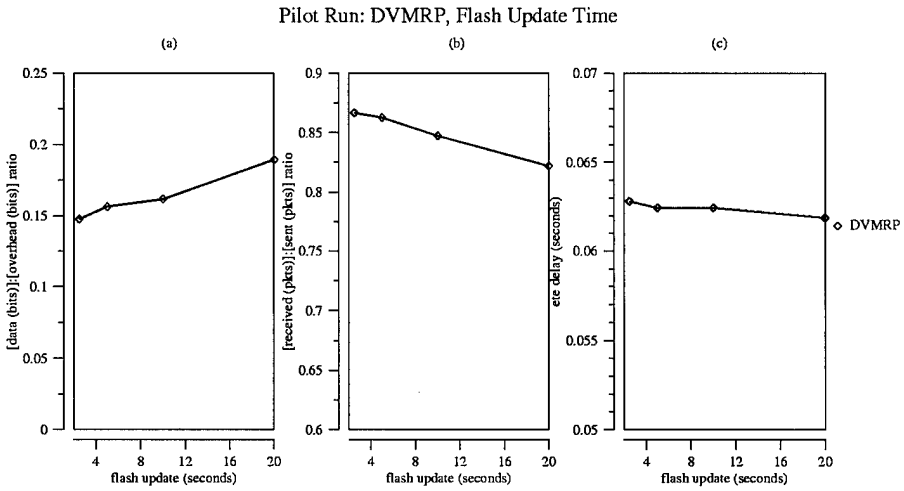

Figure 3.7. DVMRP Flash Update Sensitivity

Finally, the prune delay presents a case against this kind of analysis. While the system response shown in Figure 3.8 appears to have little sensitivity to the prune delay, pilot studies into higher packet sizes show otherwise. Packet size, which appears to be a completely unrelated parameter, reaches a horizontal data to overhead asymptote in Figure 3.9.

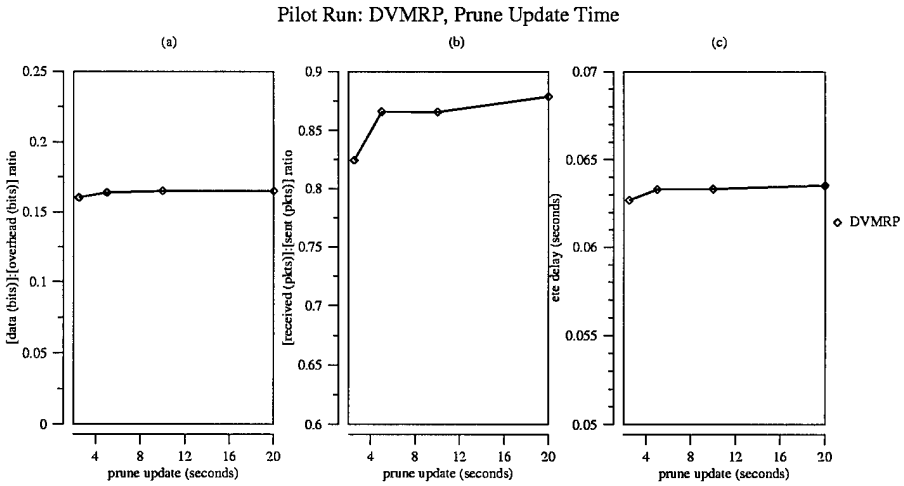

Figure 3.8. DVMRP Prune Sensitivity

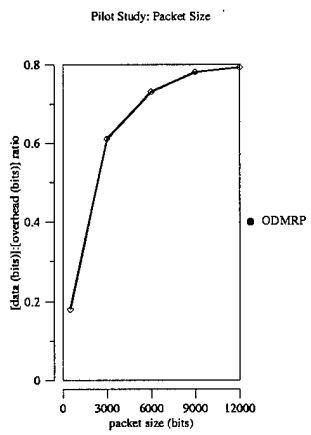

Figure 3.9. DVMRP Packet Size Sensitivity

In an ideal DVMRP network, the tree is pruned often enough and accurately enough that packets only go down branches leading to destinations. In this case, the data to overhead ratio should increase without bound as packet sizes increase. This is because the true overhead (flash updates, prunes, grafts, etc.) is completely independent of the data flowing over the trees it creates. Thus the data to overhead ratio is

$$D : O = \frac{d(s)}{O} \tag{3.7}$$

where $d(s)$ is data as a function of the packet size and the overhead, O, is constant. However, this ideal case doesn't really occur. Instead, some branches exist that lead to nodes that have not yet pruned themselves (this is also an effect of using data packets instead of the IGMP protocol, as discussed in Section 3.7.1). In this case, the data to overhead ratio in the ideal case is

$$D : O = \frac{d_{rc}(s)}{O + d_{ri}(s)} \tag{3.8}$$

where d_{rc} is data routed correctly and d_{ri} is data routed incorrectly. As packet size increases, the constant term becomes negligible and the data to overhead ratio becomes a ratio of data routed correctly to data routed incorrectly.

The amount of data routed incorrectly is a function of how accurately and quickly the system adapts itself to changes. In particular, the effectiveness of pruning the network

directly controls how many packet are misrouted. This is an example of what seemed to be two unrelated parameters, packet size and prune time, exhibiting some correlation.

Further pilot studies show decreasing the prune time from the baseline of 10 seconds to 2.5 seconds increases the data to overhead asymptote in the packet size study by about 10% to .88. This makes sense, as faster pruning yields fewer dead branches and less misrouted data. Thus, while in Figure 3.8 the system seems relatively insensitivity to prune time out, faster prune times may yield better results at higher packet sizes. To avoid the packet loss that occurs from pruning too quickly, a 5 second prune wait time is implemented.

Table 3.5 presents a summary of the timing choices for each protocol.

Table 3.5. DVMRP / mobile IP Final Timing Configuration

ODMRP	Forwarding Group Timeout	150 sec
	Route Timeout	100 sec
Mobile IP	Time Between AGENT SOLICITATIONS	2.5 sec
	REGISTRATION REQUEST Timeout	5 sec
	Collocated Address Timeout	10 sec
	Registration Timeout	5 sec
DVMRP	Neighbor Probe	26 sec
	Neighbor Timeout	91 sec
	Flash Update	10 sec
	Prune Update	5 sec
	Graft Retransmission	5 sec
	Route Expire	140 sec
	Prune Expire	600 sec

3.8 Factors

The parameters varied for the SUT are the group membership, group density, and the satellite failure rate. Of course, the multicasting protocol is also a factor. These factors are separated by definition from the workloads, which are network traffic patterns.

The group density levels are at two states: sparse and dense. The definitions for sparse and dense is taken from the definitions Deering et al. [37] used in the PIM protocol. Deering defines sparse as "the number of networks or domains with group members is significantly less than the number of network/domains in the Internet." For the SUT,

this consists of group members only in urban areas, with the membership level randomly distributed amongst seven urban areas. For comparison to past research, these urban locations are the same as those chosen by Pratt [4]. This is shown in Table 3.6. Dense mode consists of membership randomly distributed over the earth's surface.

Table 3.6. Mobile Node Home Locations

City	Longitude	Latitude	Altitude
Rio de Janero	-43.22	-22.90	0.01
Melbourne	144.97	-37.80	0.00
Kansas City	-94.59	39.13	0.23
Dharan	50.00	27.00	0.76
Beijing	116.47	39.90	0.18
Berlin	13.42	52.53	0.03
Capetown	18.37	-33.93	0.00

Initially, group membership was to follow Muller's loading levels. Because of computing resource limitations, it is necessary to reduce the membership levels based on the protocol. For ODMRP, which requires a large number of packet transmissions (to be discussed in more detail in Chapter 4), light is defined as 5 mobile nodes per group, one each in Rio de Janero, Melbourne, Kansas City, Dharan, and Beijing. Medium consists of 10 nodes, distributed among all seven urban locations. Heavy is 15 nodes also distributed among all urban locations.

DVMRP, being a lower overhead solution, allows for higher group membership levels. It still is not possible to adapt Muller's levels, but levels of 40, 60 and 80 members were achievable. These levels were accomplished using both the sparse and dense group density levels. To allow comparison with ODMRP, the 5-10-15 member levels were also conducted. For the lower membership levels, the simulation was conducted with a one-to-many (n members, one sender, n receivers) and many-to-many (n members, n senders, n receivers) transmission scheme scenarios. At higher membership levels, only the many-to-many scenario was used.

The satellite failure rate takes on two values: single critical satellite failure or no failure. Critical failure are determined in a manner modified from the algorithm Pratt defined [4]. The steps are as follows:

1. Generate packets from all senders to all receivers.
2. Count packets that traverse each node.
3. Remove the satellite that has the most number of packets traversing it.
4. In case of a tie, remove the satellite with the most packets destined for Dharan.

3.9 Workload

The workload submitted to the SUT consists of network traffic of various intensity. While a case can be argued validly that this workload is actually another factor, it is convenient to think of the traffic as the workload because it is "representative of system usage in real life" [38]. Based on the maximum throughput of the network links, 2.5 Mbps, and the mean packet size of 400 bytes, three workloads of low, medium and high intensity are defined [4]. Table 3.7 presents the values for each loading level.

Table 3.7. Loading Levels

Loading Level	**Total Traffic Generation Rate**
High (100%)	780 pps
Medium (80%)	624 pps
Low (50%)	390 pps

The total traffic generation is evenly distributed amongst all mobile nodes. This guarantees the arriving bits per second never sustain rates greater than the bits per second a link can carry. In terms of queueing theory, by distributing the total traffic evenly amongst all senders, the ρ of the system will never be, on average, greater than 1. This means the system will be stable.

3.9.1 Scaling. Generating high loading levels requires a large amount of computing power, as each individual packet must be tracked (requiring memory) and simulated (requiring CPU time). If a simulation could be scaled in such a way that less packets were generated but equivalent results were obtained, this would result in a dramatic reduction in simulation requirements.

The system under test consists of many satellites, each acting as a unique processor and operating with a unique queue. The end-to-end (ETE) delay of the packet is the sum

of the time spent at each satellite and the propagation between satellites. Thus the ETE delay for a packet is the sum of the average time spent in each system:

$$ETE = \sum_{i=1}^{n} T_{av_i} + \sum_{i=1}^{n-1} d_i \tag{3.9}$$

where n is the number of hops in the route, T_{av} is the time spent spent at each hop, and d is the propagation time between nodes.

The time spent in each system (hop) is composed of four parts:

- transmission time,
- propagation time,
- processing time, and
- queueing time.

Two of these components, the transmission time and the processing time, create the service time (T_s) for a packet. The queueing time, or waiting time (T_w), is a function of how fast the packets arrive and how fast they are serviced. The propagation (T_p) time is a factor of the distance between the sending and receiving satellites. These delays are illustrated in Figure 3.10.

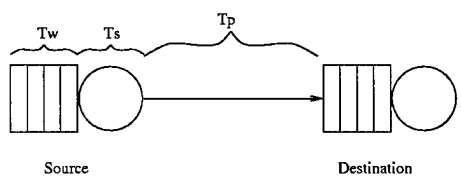

Figure 3.10. ETE Delay Components

The processing time at each satellite can be assumed to be extremely fast when compared to the transmission time and eliminated. The transmission time is a function of the size of the packet and the data rate of the transmission channel. From this, the expected service time (T_s) for a packet is

$$E[s] = \frac{b}{R} \tag{3.10}$$

where b is the size of the packet in bits and R is the date rate of the channel in bits/second.

This would be a deterministic service rate if all channels had equal data rates or all packets were equally sized. However, this is not the case. The service rate should be categorized as having a "general" density function. Combined with an exponentially distributed inter-arrival rate of mean λ (packets/sec), the satellites can be modelled as a constellation of M/G/1 processors.

The average time a packet spends in a system for a M/G/1 system $(T_s + T_w)$ is [38]

$$T_{av} = E[s] + \frac{\rho E[s](1 + C_s^2)}{2(1 - \rho)} \tag{3.11}$$

where ρ is $\lambda E[s]$ and C_s is the coefficient of variation of the service time.

In a non-scaled simulation, increasing the inter-arrival rate by a factor of F makes ρ become ρF. This change in (3.11) causes the time in the system to become

$$T_{av} = E[s] + \frac{\rho F E[s](1 + C_s^2)}{2(1 - \rho F)} \tag{3.12}$$

To achieve the same results in a scaled system, the service time $E[s]$ can be increased by a factor of F. Thus $E[s]'$ becomes $E[s]F$ and ρ' becomes ρF. These "scaled" changes cause the time in the system shown in (3.11) to become

$$T'_{av} = E[s]F + \frac{\rho F^2 E[s](1 + C_s^2)}{2(1 - \rho F)} \tag{3.13}$$

or

$$\frac{T'_{av}}{F} = E[s] + \frac{\rho F E[s](1 + C_s^2)}{2(1 - \rho F)} \tag{3.14}$$

which a scaled version of (3.12). Thus by increasing $E[s]$ by a factor of F instead of the inter-arrival rate, it is possible to get statistically equivalent systems, with the exception that

$$T'_{av} = T_{av} F \tag{3.15}$$

Since a packet has been scaled to represent F packets, the propagation delay is also scaled by F, giving

$$T_p' = T_p F \tag{3.16}$$

which combined with (3.15) means that

$$ETE' = \sum_{i=1}^{n} T_{av_i} F + \sum_{i=1}^{n-1} d_i F \tag{3.17}$$

or, comparing to (3.9)

$$ETE' = F(ETE) \tag{3.18}$$

It is interesting to note no knowledge of C_s is needed derive this result.

To increase the expected service time by F, the components of $E[s]$ must be increased by a factor of F. From (3.10) it can be seen that

$$E[s]F = \frac{bF}{R} \tag{3.19}$$

Thus, increasing packet size and propagation delay by factor F and reducing the ETE by factor F is equivalent to increasing the packet inter-arrival rate by factor F.

To confirm this result, the simulation was run for one hour with a 2 packet per second load, both with scaling and without. The results are shown in Table 3.8.

Table 3.8. Scaling Comparison

	2 packet/sec	**Scaled**	**Unscaled**
DVMRP	Data:Overhead	0.261860	0.262580
	Sent:Received	0.865269	0.864792
	ETE	0.634093	0.063464
ODMRP	Data:Overhead	0.119033	0.117510
	Sent:Received	0.996448	0.997870
	ETE	0.064544	0.063263

3.10 Experimental Design

With three workloads and four factors, this experiment would require 72 runs to evaluate every combination. Table 3.9 presents in summary the factors that need to be

changed to measure every interaction. Utilizing ANaysis Of VAriance (ANOVA), it is necessary to have repetition of these results to get the required confidence intervals.

Table 3.9. Factors and Workloads

Workload	Traffic Density	High
		Medium
		Low
Factors	Mechanism	One-to-all
		All-to-all
	Group Density	Sparse
		Dense
	Satellite Failure	Failure
		No failure
	Group Membership	Low
		Medium
		High
	Algorithm	ODMRP
		Mobile IP

3.11 Evaluation Technique

When analyzing a system, there are three primary methods of investigation. They are analytical modelling, simulation, and measurement. Each is appropriate for a given circumstance, with relative weaknesses and strengths. For this study, simulation is the only evaluative method that makes sense. The SUT in question does not exist in any shape or form, making it impossible to take measurements. Analytical modelling, while capable of providing valuable insight, is not able to fully explore the nuances of the problem.

In particular, OPNET Modeler 7.0b is used as the simulation environment allowing for both development and execution of the simulation model. While OPNET does have the capacity to model the orbital trajectories of satellites, it is not particularly adept at determining these trajectories. For constellation design, Satellite Toolkit (STK) 4.0 by Analytical graphics, is used. The ephemeris data created in STK can be imported into OPNET. This gives OPNET the necessary data to model the orbits.

Figure 3.11 depicts the structure of the nodes. In both models, the processor layer is where the ODMRP or DVMRP and mobile IP algorithms reside. The "MAC" layer

is responsible for knowing what interfaces to send broadcast and unicast messages. The source layer both creates

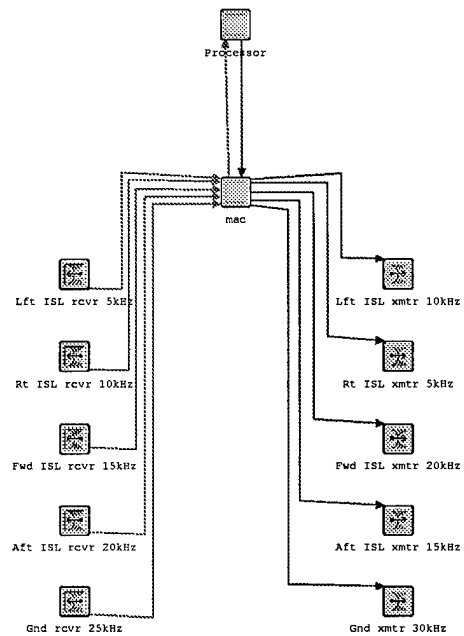

Figure 3.11. OPNET Satellite Node Model

3.11.1 Implementation Details. To implement the protocols presented in Section 3.7.1 into OPNET, there are several techniques that needed to be used, both at the system and the process levels.

For both protocols, the process model consisted of an initiation state, which immediately transferred into an idle state. The idle state was only exited when either a packet arrived or an internal interrupt occurred. Packet arrival caused the system to enter into a case statement where appropriate processing was conducted on the packet. Interrupts were either set or cleared based on the outcome of the case statement.

In the DVMRP/Mobile IP protocol, these interrupts were stored in an interrupt list. This list contains an interrupt's unique id, type, and relevant data for that type. When OPNET activates an interrupt, that particular ID is removed from the interrupt list and executed. The ODMRP model works similarly, but since there are only two primary interrupts (forwarding group member expire and route expire) which are independent of the packets received, they are not tracked in a list format.

At the system level, there are several interesting implementation features. The first implementation of note involves involves the footprint of the satellites antennas. This is obtained creating a radius from which any ground station inside of will be considered in view. By using the satellite elevation and the footprint radius from [5], Pythagoreans theorem can be used to find this radius, since the maximum reach of the satellite is the hypotenuse of these two vectors. Calculating this determines the maximum communication radius of 2436 km. This concept is illustrated in Figure 3.12.

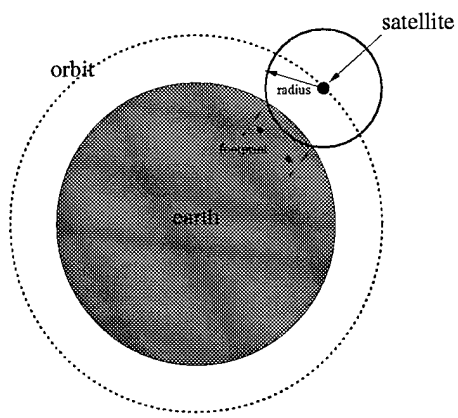

Figure 3.12. Method of Determining Satellite Footprint

The procedure was used in reverse for the ground stations to determine what satellites are in-view. In the ODMRP case, each ground station finds all satellites in the maximum communication radius and then transmits only to the closest one. In the DVMRP case, communication occurs between the the ground and all in view satellites when in broadcast mode and between the ground station and the satellite it is co-located with in transmit mode.

Also at the system level, the manner of communicating between the ground and satellites is implemented in a non-standard manner. Instead of simulating a multiple-access scheme, which requires some manner of collision detection, an alternative scheme was implemented. By increasing the bandwidth of the up and down links to 2.5 Gbps, multiple ground stations are capable of communication with a single satellite. This allows typical packets to spend less than 1.5 μs in the channel, making it appear as if multiple stations can access the satellites simultaneously. Collisions are still possible if packets are

transmitted from two sources inside of this window, but they are much less likely. While this mechanism removes the bottleneck from the up and down links, the study is examining the performance of the routing protocols, not the multiple access techniques of the ground link. Additionally, there is no consistent data on what parameters are common for this component of the constellation.

The ODMRP system relies on unique packet identifiers to discard duplicate packets. To reduce overhead, packets received twice are discarded. This is done by maintaining a list of the last 2500 packet sources and identifiers that were received at each ODMRP node. The number was chosen by manual observation. It is the point at which under high loading levels and high membership levels repeat packet reception was not observed.

3.11.2 Verification and Validation. Verification of the system occurred on several levels. The first level of verification occurs at the compiler level. The code is written to the specifications of the protocol and then checked against it. After the code compiled, the debugging phase of verification begins.

To debug the code, a variety of pilot runs were introduced. At each point, the results of the pilot runs were compared against the expected results. If differences existed, the algorithms were examined until the source of the discrepancy was found. Other metrics, such as hop count and end-to-end delay were examined. Traces of the geographic positions of hops, as they progressed from satellite node to satellite node were compared against the expected trace. Once the algorithms performed as expected, the validation began.

Validating the model presented a unique challenge as there are no real-world systems implemented in the manner being simulated. Additionally, there is very little research into this application. However, there is a significant amount of research on the DVMRP, mobile IP and ODMRP protocols in their "typical" environments. By comparing the performance trends against expectations described in [23, 25, 24, 31, 8], it is possible to get an indication whether the system is performing as the other studies.

Using the validation list from [38], validation consists of:

- Assumptions,

- Input parameter values and distribution, and
- Output values and conclusions.

It was possible to successfully complete the first two items.

By presenting the assumptions to experts, a determination of their reasonableness was obtained. Previous work from [5, 6, 4] also provided an expert source from which to validate the assumptions found throughout the research.

The input parameters were obtained from real world requirements and data. Packet formats, sizes and size distribution were designed to mimic or closely resemble real world implementations. Ground station locations were chosen to represent potential worldwide communication scenarios. Real network system measurements from [34] validated inter-arrival times.

While it is difficult to validate the output values since there are no experts or real systems of this type to take measurements from. However, by using the system in a single sender - single receiver multicast format, it was possible to create a simple unicast system. This simple case was validated against research by [5] as well as the expert intuition of the author and advisor.

3.12 Summary

This chapter presented the SUT, and determined what parts of the system are contained in it. The parameters defining the SUT were described and either simplified or assigned a state. Factors of interest were extracted and representative levels were chosen. Finally, a full general factorial experimental design with replication was proposed.

IV. Results

4.1 Introduction

This chapter presents an analysis of the system performance results obtained from the simulation trials. The statistical accuracy of the simulation results is examined first. This is followed by an investigation into the Distance Vector Multicast Routing Protocol (DVMRP) performance metrics. Section 4.4 discusses the observed performance of the On Demand Multicast Routing Protocol (ODMRP). Finally, the two protocols are compared in Section 4.6.

4.2 Statistical Accuracy

To determine results that are an accurate representation of an entire simulation run, only data from the simulation steady-state period was sampled. At startup, both protocols have a period of transience as routes are constructed and ground stations link up with satellites.

To eliminate this transient period from the results, pilot simulations were executed for 33 to 50 minutes of simulation time. These times were chosen based on the amount of time required to reach steady-state and a coefficient of variance (C.O.V.) of less than 0.1. Each run was inspected and the transient period was identified. Defining transient periods is, at best, a heuristic process. The period of transience was defined to be the point at which the majority of the metrics had stabilized to a relatively constant value. The smoothest section of the pilot simulation with stable metric values was used to mark the beginning of the steady-state period. Smooth was defined by inspection. Once this was determined, the simulations were then performed again, with data only being collected 100 to 200 seconds after the steady-state period began.

By running the simulation sets multiple times with multiple random seeds, a confidence interval can be obtained. For this research, a confidence interval of 90% was chosen. This interval allows for a reasonable level of accuracy without requiring an inordinate number of trials. The confidence interval was calculated using the method discussed in [38],

which states for a given mean, there is a 90% confidence the real mean lies inside

$$\left(\bar{x} - z_{[1-\frac{\alpha}{2}]}\frac{s}{\sqrt{n}}, \bar{x} + z_{[1-\frac{\alpha}{2}]}\frac{s}{\sqrt{n}}\right) \tag{4.1}$$

where $\bar{x}$ is the mean, z is the unit normal distribution area, α is the confidence interval, s is the standard deviation and n is the number of samples. For all experiments, the simulation was considered complete when the standard deviation did not exceed $\pm 10\%$ of the mean (this is the Coefficient of Variation). According to [38], the coefficient of variation is insignificant if less than 20% regardless of the mean. This was achieved in 4 to 5 replications for all experiments.

These intervals were used to determine both the confidence of the measurements made as well as the significance of values. As described in [38] and other statistical sources, if a confidence interval includes the mean value of another measurement, those two measurements are statistically identical. If the two confidence intervals of the measurements do not overlap at all, the values are statistically significant (at the $n\%$ confidence level). If the two values overlap only in the confidence interval, the students t-test is used to determine significance.

The student-t test is calculated by first computing the mean and standard deviations of the difference of the two measurements (a and b),

$$\bar{x} = \bar{x_a} - \bar{x_b} \tag{4.2}$$

and

$$s = \sqrt{\frac{s_a^2}{n_a} + \frac{s_b^2}{n_b}}, \tag{4.3}$$

respectively, where (again) s is the standard deviation, $\bar{x}$ is the mean and n is the number of samples. The number of degrees of freedom are calculated according to

$$\nu = \frac{\left(\frac{s_a^2}{n_a} + \frac{s_b^2}{n_b}\right)^2}{\frac{1}{n_a+1}\left(\frac{s_a^2}{n_a}\right)^2 + \frac{1}{n_a+1}\left(\frac{s_a^2}{n_a}\right)} \tag{4.4}$$

The confidence interval is now found using

$$\left(\bar{x} - t_{[1-\frac{\alpha}{2};\nu]}, \bar{x} + t_{[1-\frac{\alpha}{2};\nu]}\right) \tag{4.5}$$

where $t_{[1-\frac{\alpha}{2};\nu]}$ is the $(1-\alpha/2)$-quantile of a t variate with ν degrees of freedom. If this confidence interval includes zero the two measurements are considered statistically identical. If the confidence interval does not include zero, the two measurements are considered unique.

To determine the contribution each factor played in a metric's variation, ANalysis Of VAriance (ANOVA) was used as described in [39]. ANOVA is a means of calculating the total contribution of each factor and the the contribution of the interactions between factors. Before this analysis was performed, the following three assumptions about the data were verified:

- Independent errors: checked with expected response versus residual plot.
- Normally distributed errors: checked with normal quantile-quantile plot.
- Constant standard deviation of errors: checked with experiment versus residual plot.

(4.6) represents the total sum of squares for the metric where there are four factors, and a, b, c, and d represent the number of levels for these factors. Equations (4.7) and (4.8) show how the sum of squares for the main effects A and B are found. (4.9) shows how the sum of squares for the A-B interaction is computed.

$$SS_T = \sum_{i=1}^{a}\sum_{j=1}^{b}\sum_{k=1}^{c}\sum_{l=1}^{d} y_{ijkl}^2 - \frac{y_{....}^2}{abcd} \tag{4.6}$$

$$SS_A = \frac{1}{bcd}\sum_{i=1}^{a} y_{i...}^2 - \frac{y_{....}^2}{abcd} \tag{4.7}$$

$$SS_B = \frac{1}{acd}\sum_{j=1}^{b} y_{.j..}^2 - \frac{y_{....}^2}{abcd} \tag{4.8}$$

$$SS_{AB} = \frac{1}{cd}\sum_{i=1}^{a}\sum_{j=1}^{b} y_{ij..}^2 - \frac{y_{....}^2}{abcd} - SS_A - SS_B \tag{4.9}$$

To determine the percentage of the variance caused by an effect or an interaction, the sum of squares for that effect is divided into the total sum of squares. Finally, an F-test was computed for each factor, verifying the significance of each factor's contribution to the variance.

4.3 DVMRP / Mobile IP Scenarios

Two different sets of scenarios were examined for the DVMRP / Mobile IP protocol. The first scenarios involved high (when compared to the second scenario) membership levels of 40, 60 and 80 multicast members. The second scenarios involved low (5, 10 and 15) membership levels. At the high membership levels, all-to-all communication was implemented. At the lower membership levels, both all-to-all and one-to-all communication was used. For high membership, the stations were located in strictly urban (sparse) and evenly distributed (dense) configurations. In contrast, at the low membership levels, ground stations were located only in urban areas. They were placed in a round-robin fashion following the ordering in Table 3.6. For all experiments, low, medium, and high workloads were used.

4.3.1 DVMRP Low Membership. At the low membership levels, DVMRP exhibited a consistent time to converge to steady-state levels regardless of loading level and membership. For all scenarios, the simulation had converged to steady state by 1200 seconds. As a representation of this convergence, Figures 4.1 and 4.2 present the values of the metrics over simulation time for the 5 member, low loading level cases. These values were sampled at 15 second intervals and represent the cumulative value of the metric at the time sampled.

The presence of multiple sources in Figure 4.1 accounts for its smoother appearance than Figure 4.2. Since the data is dependent on only one source in the latter case, the results are not an average of multiple sources as they were in the first case. This causes the results to exhibit more variation.

Figures 4.3 and 4.4 present the metrics as a function of loading level and number of members. Plot (a) presents the data to overhead ratio, (b) the received-to-sent ratio, and (c) the end-to-end delay. In these figures, a clear symbol indicates a statistically significant

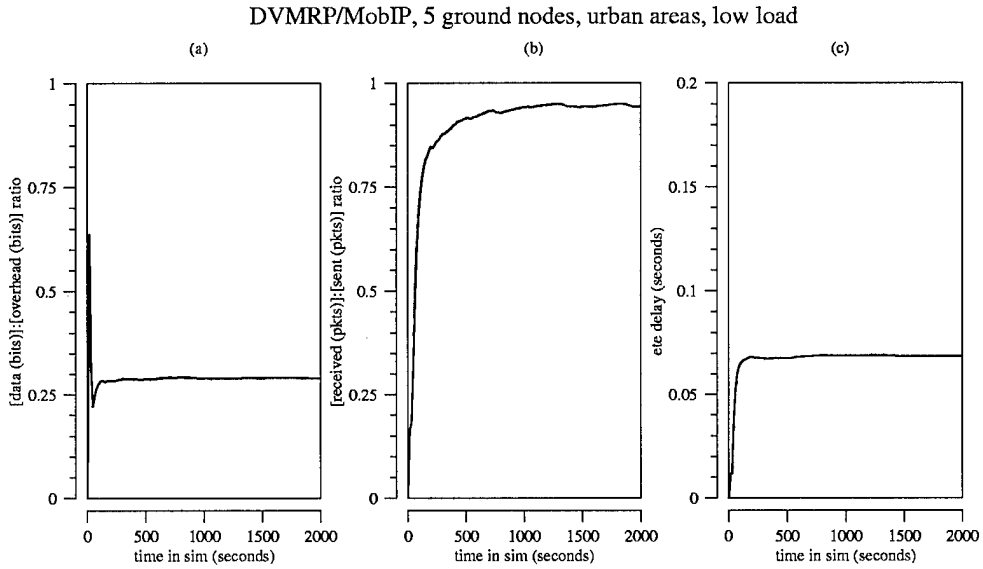

Figure 4.1. Sample DVMRP/MobIP All-to-all Simulation Run

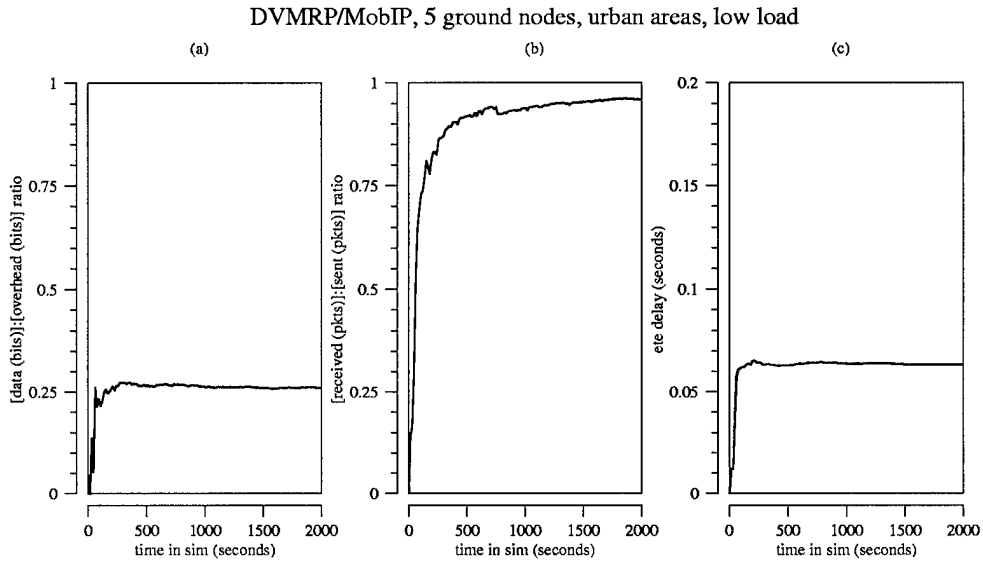

Figure 4.2. Sample DVMRP One-to-all Simulation Run

value. Two or more filled symbols represent values (for a particular loading level at the 90% confidence interval) which are statistically equivalent. The raw data can be found in Tables A.1 through A.6 in Appendix A .

4.3.1.1 Data-to-Overhead Analysis. The general trend exhibited by the data-to-overhead ratio was increasing membership led to higher data-to-overhead ratios.

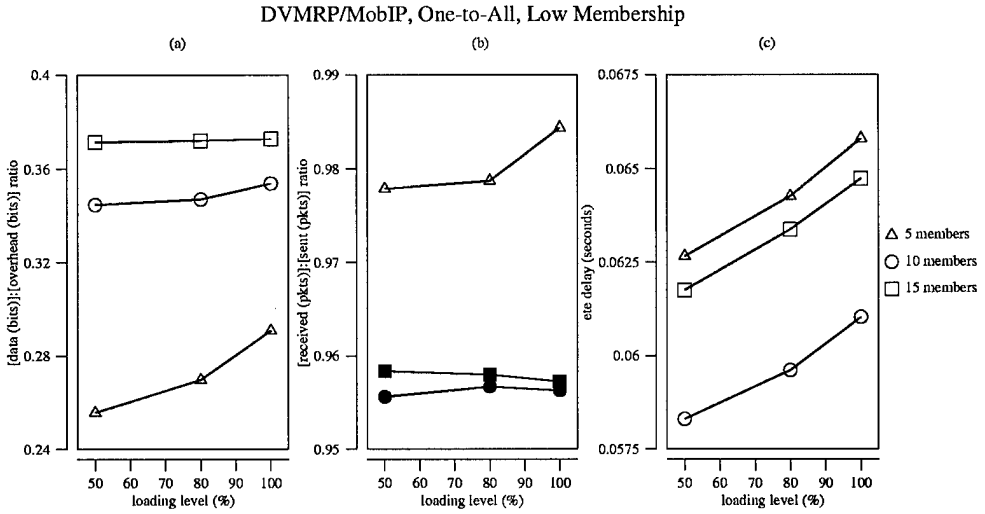

Figure 4.3. DVMRP/MobIP One-to-all

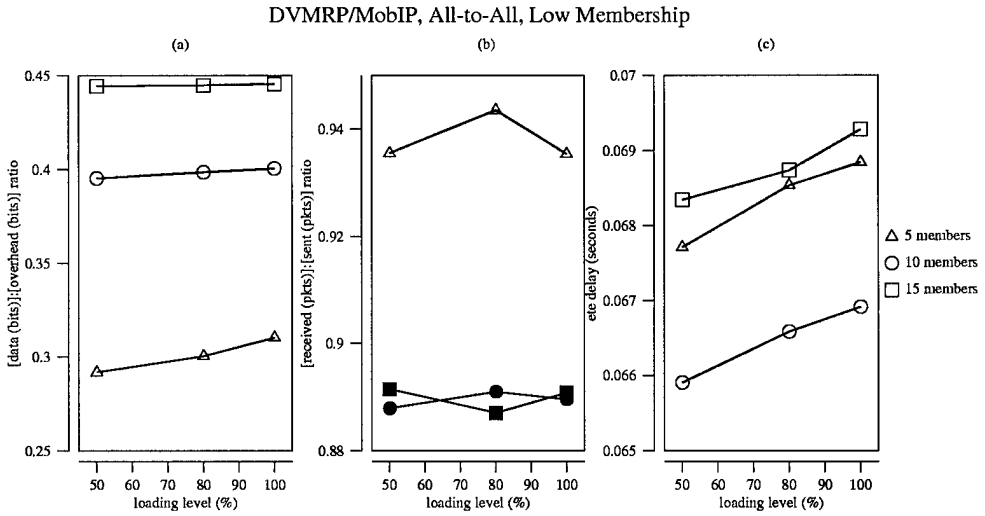

Figure 4.4. DVMRP/MobIP All-to-all

Each membership level at any chosen loading level gave statistically unique ratios. The worst (i.e., lowest) data-to-overhead ratio was the 5 member, 50% loading level, one-to-all scenario, with a mean of 0.256 bits of data sent for every one bit of overhead. The best (i.e., highest) ratio was the 15 member, 100% loading level, all-to-all scenario, which transmitted a mean of 0.446 bits of data for every one bit of overhead.

While the values are significant between memberships at a given loading level, the values are statistically identical within a given membership level. For instance, in both the

one-to-all and all-to-all transmission scenarios, the 15 member group provided statistically similar ratios (at the 90% confidence interval) across loading levels. Only the 5 member group showed an increase across all three loading levels. Specifically, in the all-to-all scenario, the 5 member case increased the ratio by 6.31% and in the one-to-one scenario had the largest increase, that of 13.78%.

Although the 5 member case had the largest increase in data-to-overhead as a function of loading level, it also had the largest variation. The one-sided 90% confidence interval consistently made up the largest percentage of the mean of any of the three membership levels. It also exhibited the largest growth in variation versus change in workload for both scenarios, increasing 99% in the one to all scenario and 234% in the all-to-all scenario.

As DVMRP is a protocol based on updating distance vectors as they change, it is expected as loading levels increase, the data-to-overhead ratio will increase. As described in Section 3.7.3, the overhead from this updating is independent from the amount of traffic that is sent. As traffic increases, the denominator of the ratio remains "fixed," causing the ratio to grow. Furthermore, the presence of misrouted data traffic is what keeps this ratio from increasing without bound. On the other hand, simply increasing the workload within a group membership level does not necessarily increase the data-to-overhead ratio. With the sparse arrangement of nodes in urban areas, the multicast tree structure is not changed when more members are added. The same fraction of packets are misrouted as before, which explains why the 10 and 15 member groups do not exhibit the same trends found in the 5 member group.

ANOVA lends additional weight to this, as 77% of the variance found in data-to-overhead metric is caused by the membership factor, while 19% comes from the mechanism of transmitting (one-to-all or all-to-all). Even more telling, less than 1% of the variance comes from the workload. A complete ANOVA table for DVMRP can be found in Table A.9.

Higher membership levels in similar urban locations create trees that are more efficient. If an urban area has only one subscriber, each packet received by the satellite it is co-located on is only transferred once over the satellite-ground link to the receiver.

Only one packetful of data is contributed to the data-to-overhead ratio in this scenario. If an urban area has n members, n times as much data is contributed per packet over the last link, which increases the ratio of data-to-overhead. This effect also brings the system closer to the natural limit of the data to overhead ratio at lower loading levels (see (3.7) and (3.8)) by maximizing the contribution of the last link.

The one-to-all case scenario represents traffic sent from one source to all the group members. As such, it represents much higher single point traffic than the all-to-all scenario. Therefore, it is expected the behaviors of the two scenarios follow the same trends but not take on the same values. Having more sources in the all-to-all case increases the data-to-overhead ratio by utilizing more of the distance vector updates. Instead of utilizing the vector information of one source to n destinations, the protocol utilizes the information for n sources to n destinations, thus amortizing the cost of the DVMRP process across more routes.

4.3.1.2 Received-to-Sent Analysis. The trends present in the received-to-sent to metric were different than those in the data-to-overhead metric. The 90% confidence intervals overlapped several of the data points. This caused the values at different loading levels to be statistically identical across the membership level. Only at the 5 member level did the values differ. At the 10 and 15 member levels, the values were statistically the same, both for the all-to-all and one-to-all scenarios.

While the values may have been statistically similar at the 90% confidence level, these intervals still were less than 10% of the overall mean. Additionally, they were a smaller percentage of the mean than the data-to-overhead ratio. All intervals were less than 1.26% of the mean. This largest percentage from the 5 member, 100% loading, all-to-all scenario.

In contrast to the data-to-overhead metric, the ANOVA reveals 71% of the received-to-sent metric variance is contributed by the transmission mechanism. The membership level contributes only 24% of the variance. This latter percentage is a result of the separation of the 5 membership level from the 10 and 15 membership levels. This is attributable to two factors: the implementation of the satellite multiple access scheme and the geographic positioning of the ground nodes.

The multiple access scheme implemented is extremely simple. Since the system does not utilize carrier-sensing, spread spectrum, or any other method of preventing channel contention, the potential for packet collisions exists. For the 5 member case (in which no more than one ground station is co-located with a satellite at any time), collisions are zero. This was confirmed by a trials in which packet collisions were recorded and tabulated. No collisions were present in packets that were received. However, as only packets received by a multicast member were counted, this tally did not cover the packets that were involved in a collision and determined to be unreadable and thus undeliverable. These trials were repeated for higher membership levels. For the 10 and 15 member cases, in which two to three ground stations are simultaneously co-located with a single satellite, 0 and 45 collided packets were recorded, respectively. This does not account for the entire difference in received-to-sent, but it does support the trend of increasing packet collisions as the number of ground stations per satellite increases.

While packet collisions do support the discrepancy in the received-to-sent ratio for the all-to-all scenario, they do not explain the one-to-all scenario. In the one-to-all case, regardless of the number of members, there can be no collisions. Since only one ground station is transmitting, there are no alternate sources to produce conflicting packets. It is notable the discrepancy in the all-to-all case is approximately a 0.04 to 0.05, while the one-to-all is 0.02 to 0.03. This supports the assertion that there is a second factor at work.

This second factor is attributable to node placement. The 5 member group only has sources at five of the seven locations. In this manner it is unique. All other membership levels have group members at all seven locations. As the urban locations were chosen to spread membership out widely over the earth, having two fewer cities in use covers 29% less surface area, and will result in shorter trees. Similarly, this will potentially result in marginally more reliable routes, resulting in the 5 member group having a slightly higher received-to-sent ratio.

4.3.1.3 End-to-End Analysis. As expected, the mean end-to-end delay increases as the loading level increases. This is directly attributable to the queue length. The transmission speed of the Inter-Satellite Links (ISLs) are of a constant value. At the

100% link loading level, ρ (ratio of bit arrival rate to bit transmit rate) approaches one for certain links, creating longer delays. The difference between the three membership levels in this regard is minimal, all increase by approximately two milliseconds from the low loading level to the high loading level.

Much like the received-to-sent ratio, the end-to-end metric was mostly affected by the mechanism of transmitting (68%) and the membership level (20%). The variance from the mechanism of transmission is most visible in the consistently higher end-to-end delay of the all-to-all delay than the one-to-all.

This difference is an expected by-product of the multidirectional properties of the all-to-all transmissions. In contrast to the one-to-all scenario, the all-to-all scenario has multiple trees with branches that converge. The one-to-all scenario only has branches that split. As such, arrival rates on the one-to-all configuration ISL's maintain or decrease with every hop from the source. Arrival rates for the all-to-all can take on additional traffic, causing additional queuing delays as the packets wait for transmission.

The other notable trend in the end-to-end delay is the 10 member scenario has a lower end-to-end delay than the 5 and 15 member scenarios. This is unusual. While there is no reason to expect any membership level to have higher or lower delays than any others, there is definitely a trend in the 10 member scenario. Most likely, this is attributable to the fact the 10 member group did not have an even distribution of ground stations. For instance, the 5 member group had one member at 5 five sites – 0 weighting. Weighting (w) is defined to be

$$w = \frac{n_{unique}}{n_{total}} \tag{4.10}$$

where n_{unique} is number of locations with a "unique" numbers of nodes and n_{total} is the the total number of locations. Unique is defined as follows:

- Cities that share common numbers of ground stations are grouped together.
- The largest group of cities is considered "standard."
- Cities not in the standard group are considered unique.

This means that w is 1 in a sparse network with no standard number of ground nodes per locations, and 0 in one with an entirely evenly distribution of ground node. Because the nodes were placed in round-robin fashion, w cannot exceed 0.5 in this study. By this definition, the 15 member scenario had two members at six sites and three members at one site, or a 0.143 weighting. However, the 10 member group has two members at three sites and one member at four sites, a weighting of 0.429. This uneven weighting skewed the results of the end-to-end delay, lowering the mean of the 10 member scenario by approximately 5 milliseconds. This is further confirmed in the next section.

4.3.2 DVMRP High Membership Levels. When increasing the membership from low (5 to 15 members) to high (40 to 80 members), the transmission mechanism was only all-to-all. Instead of placing the nodes only in urban areas, the membership was divided into sparse and dense modes. Sparse mode followed the pattern of the low membership levels, placing ground nodes in the seven urban areas defined in Table 3.6 in a round-robin fashion. The dense mode consisted of evenly distributing ground nodes over populated regions of Europe, Africa, North America, Asia, South America, and Oceania. Figures 4.5 and 4.6 illustrate the resulting metrics for the two populating methods.

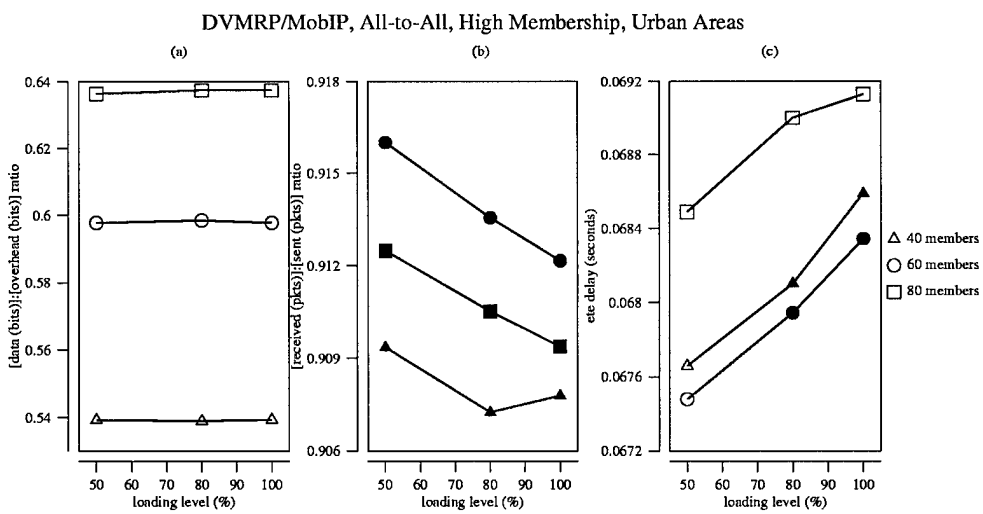

Figure 4.5. DVMRP All-to-all, High Membership, Urban Areas

4.3.2.1 Data-to-Overhead Analysis. At these high membership levels, there is no effect on the data-to-overhead metric as a function of loading levels. In all six

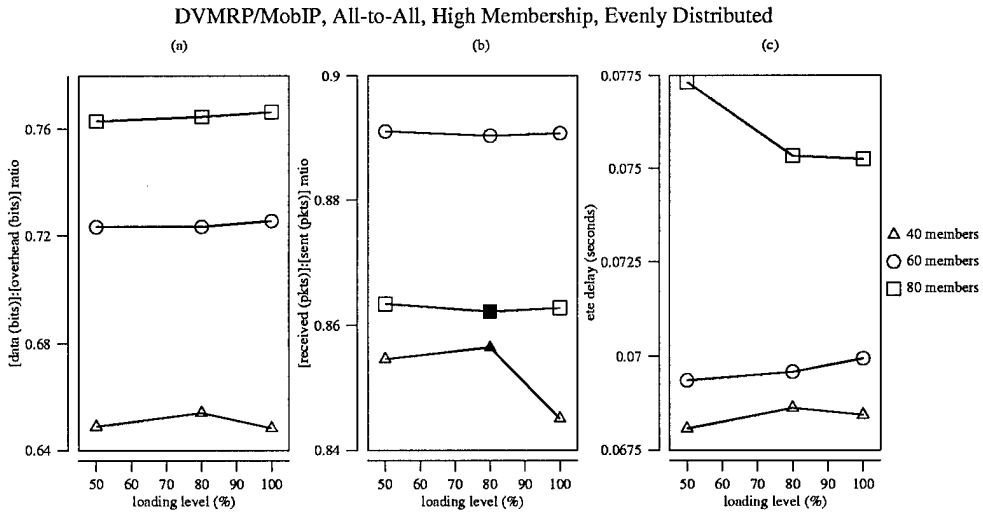

Figure 4.6. DVMRP All-to-all, High Membership, Evenly Distributed

cases, the values are statistically identical at the 90% confidence interval across the entire workload range. Furthermore, the F-test in conjunction with ANOVA shows the 0.01% contribution to the variance by the workload is insignificant.

Increasing the membership, however, does have a distinct effect on the ratio. Specifically, 34% of the variance comes from the membership. In both scenarios, increasing membership continues to increase the ratio of data-to-overhead. Figure 4.7 illustrates this trend of increasing data-to-overhead as a function of members over the entire sparse DVMRP data set. Because of the closeness of the lines and the goal of portraying data trends, unlike other figures, no information on the 90% confidence intervals is contained in these plots.

Plot (a) clearly shows there is a increasing relationship between membership size and the efficiency of the algorithm. This relationship is logarithmic. Using linear regression, the data fits the curve

$$r = m \log_{10} n + b \tag{4.11}$$

where r is the data-to-overhead ratio and n is the membership levels. Values for m, b and the associated R^2 of the regression can be found in Table 4.1.

A continuous approximation can be made of the ratio as a function of membership level and loading by fitting the relationship between m, b and membership to a linear

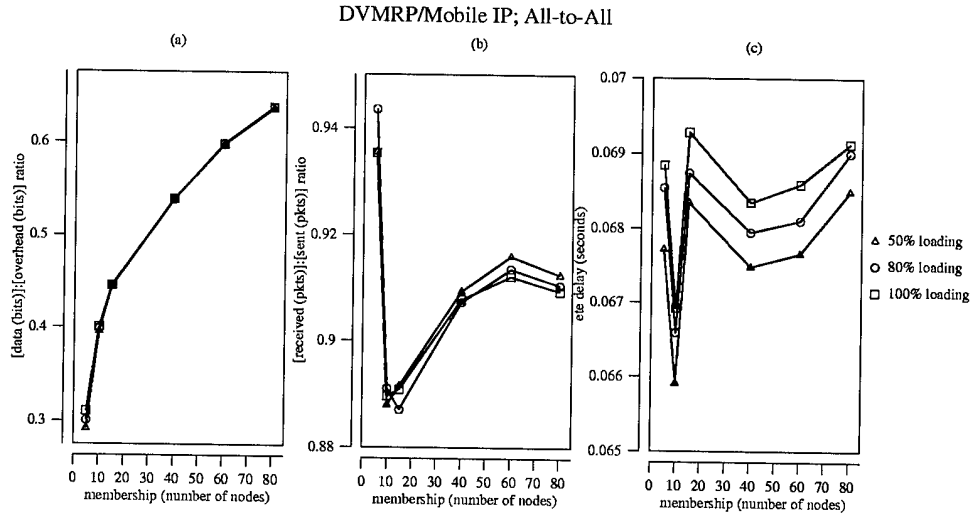

Figure 4.7. DVMRP All-to-all Full Comparison

Table 4.1. Regression Analysis, data-to-overhead, DVMRP

Loading level	m	b	R^2
50	0.27453	0.11088	0.99441
80	0.26911	0.12055	0.99517
100	0.26245	0.13162	0.99523

regression. The approximations are shown in Table 4.2. Thus the complete approximation

Table 4.2. Regression Analysis, Loading, DVMRP

Parameter	m	b	R^2
b	4.08×10^{-4}	0.08978	0.97650
m	-2.37×10^{-4}	0.28685	0.96998

for the sparsely distributed, all-to-all DVMRP scenario is

$$r = (-2.37 \times 10^{-4} w + 0.28685) \log_{10} n + (4.08 \times 10^{-4} w + 0.08978) \tag{4.12}$$

where w is the total workload, in percentage of a single inter-satellite link bitrate and n is the number of members.

Sixty-five percent of the variance comes from the density of the multicast members. Having a dense distribution places allows more ground stations to co-located with more

satellites, which again amortizes the cost of the distance vector updates over more routes. This causes the increase in data-to-overhead between the sparse and dense distributions.

4.3.2.2 Received-to-Sent Analysis. At higher membership levels, there is very little change in the received-to-sent ratio, especially in the sparse configuration. At the 90% confidence interval for this distribution there is no statistical difference between the metrics either as a function of loading level or membership level. The dense mode showed a lower number of packets were successfully transmitted and a much higher difference between the membership levels. With the exception of the 40 member group, the protocol showed no response to loading level regardless of membership. This was confirmed by ANOVA, which shows high membership levels had no significant contribution from the workload and only 14% from the membership level.

In contrast, 74% of the variance comes from the density. Additionally, when compared to the lower membership levels in Figure 4.7, the sparse scenario has a greater received-to-sent ratio. Both these trends occur because of the increase in redundancy found in dense mode versus the sparse mode and the high membership level versus the low membership levels.

Redundancy occurswhen multiple satellites provide service to the multicast group, raising the received-to-sent ratio by spreading the probability of misrouting among more satellites and thus more routes. At low membership levels, fewer members have redundancy. Similarly, at sparse density, the ground stations are physically limited from having a large number of satellites service a specific location. However, the coverage area provided by a satellites overlaps on the edges with coverage from other satellites. This allows for greater redundancy in urban areas. However, it takes larger numbers of subscribers in that area to exploit that redundancy.

4.3.2.3 End-to-End Analysis. The end-to-end delay increased both as a function of membership and density. This is supported by ANOVA data which indicates both density, membership and the interaction between density and membership provide the large portion of variation. The dense distribution averages nearly 10 millisecond slower

per packet than the sparse distribution. Additionally, the 80 member multicast group has the longest end-to-end delay, the culmination of following a trend towards higher delay with the addition of additional members.

However, for the sparse, urban scenario, the dependence on delay to membership level is small, increasing delay by less than 2 milliseconds. This is again attributable to the weighting of the distribution. The 40 member case has a weighting of 0.2857, and the 60 and 80 member cases a weighting of 0.4286. This increase in weighting causes the delay to grow slightly.

The dense case is not really analyzable in this fashion, since each addition of members adds more locations. As such, "weighting" is not comparable across membership. However, as the addition of more members creates a larger number of reception points, the number of inter-satellite links in use increases, causing queuing to occur on more links. This in turn increases the delay.

4.4 ODMRP Scenarios

The ODMRP scenarios were executed at similar levels as the low membership DVMRP protocol, with both the all-to-all and one-to-all transmission modes. Geographic positioning was achieved with users placed round-robin in the seven urban areas.

Like DVMRP, ODMRP converged to steady-state results in a consistent amount of time. However, with few exceptions, the protocol converged much more quickly than DVMRP. The samples exhibited in Figures 4.8 and 4.9 are typical, with the transient response lasting less than 1000 seconds. The 10 and 15 member all-to-all scenarios converged in less than 100 seconds. Also, like DVMRP, the all-to-all scenarios exhibited a "smoothness" not found in the one-to-all case. This smoothness is again attributable to the averaging effect of multiple sources.

Figures 4.10 and 4.11 present the metrics as functions of loading level and number of members. As before, clear symbols indicate a statistically unique value at the 90% confidence interval. Two or more filled symbols represent values that, for the particular

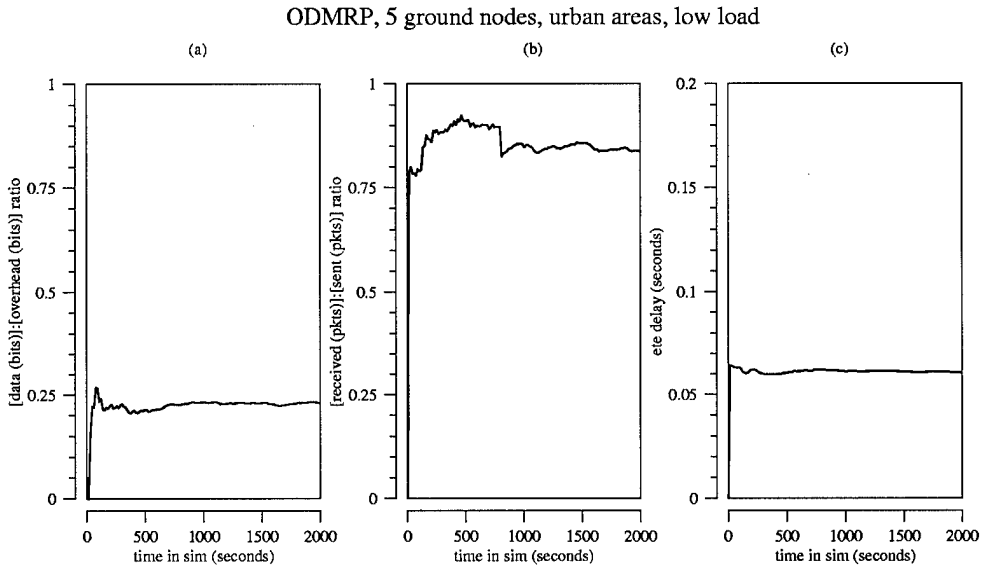

Figure 4.8. Sample ODMRP One-to-all Simulation Run

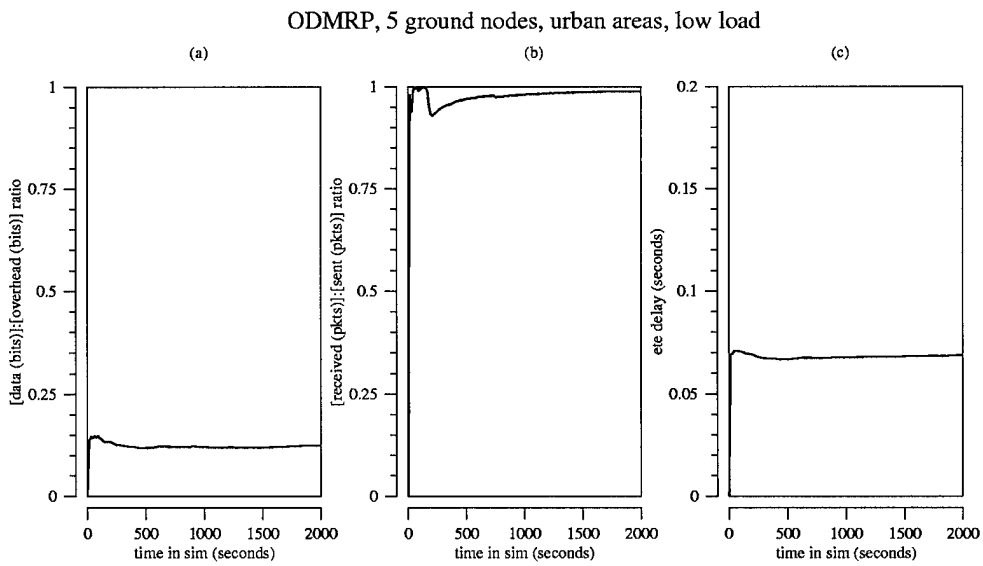

Figure 4.9. Sample ODMRP All-to-all Simulation Run

loading level, are identical at the 90% confidence interval. Table A.10 presents the results of the ANOVA analysis for the ODMRP metrics.

4.4.1 Data-to-Overhead Analysis. Both the one-to-all scenario and the all-to-all scenario exhibited a high coefficient of variation in the data-to-overhead ratio, which is reflected in the statistical similarity of many of the data points. Unlike the DVMRP pro-

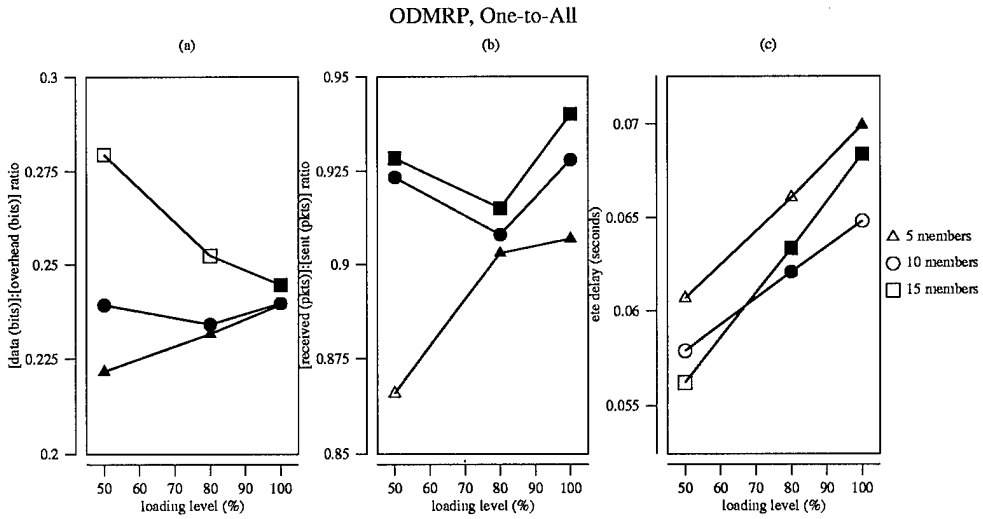

Figure 4.10. ODMRP One-to-all

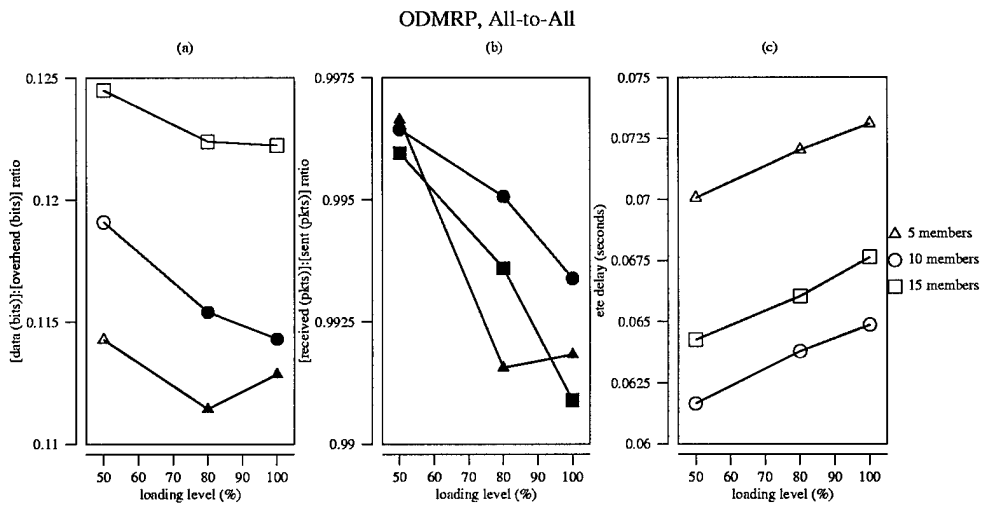

Figure 4.11. ODMRP All-to-all

tocol, the ODMRP data-to-overhead ratio converges at higher loading levels. Specifically, four of the six scenario's ratios decrease. The two cases in which the ratios appear to increase (the 5 and 10 member one-to-all scenarios) are statistically identical across the offered workload, meaning there is no conclusive evidence the metrics are increasing or decreasing.

The ratios associated with the all-to-all scenario are grouped very tightly, varying by less than 0.0128 from the highest to lowest reported ratio. This tightness is also present in the higher loading levels of the the one-to-all scenario. As such, the simulation showed

ODMRP is affected more by transmission scenario than by workload or membership, since changing from an all-to-all to a one-to-all transmission scenario more than doubled the data-to-overhead ratio.

Not surprisingly, the ANOVA analysis supports this observation, showing 96% of the variation of data-to-overhead comes from the mechanism of transmission. Only 1.5% comes from membership, and the rest from various interactions. The less than 1% contribution from workload is not even significant according to the F-test.

Regardless of the percent of total variation, increasing membership causes an increasing trend to occur to the data-to-overhead ratio. This occurs for similar reasons as for DVMRP. By utilizing more than one ground station per satellite, each packet that reaches a satellite with n ground stations can be sent n times at the same cost as one packet. This makes for a more efficient transmission of data, thus increasing the ratio of data-to-overhead.

4.4.2 Received-to-Sent Analysis. The two scenarios exhibited distinctly different responses to received-to-sent ratio metric. The all-to-all scenario has an extremely tight group of measurements, with 99.18 to 99.66% of the packets being received. The received to sent values for this scenario are statistically identical both with respect to the workload and the membership level. Effectively, this means with 90% confidence, if all members of a group are transmitting, greater then 99% of all packets transmitted will be received, regardless of membership level and traffic load.

In contrast, the one-to-all scenario has a much wider range of ratios. The wide confidence intervals prevent any solid trends from emerging. The 10 and 15 member groups are statistically identical at each workload, and the 5 member group is identical across the workload.

Unlike the other metrics, the data collected for the received-to-sent metric does not pass the first assumption of ANOVA. The residual values, found by

$$r(x_i) = \bar{x} - x_i \tag{4.13}$$

are not normally distributed. Figure 4.12 (a) shows the quantile-quantile plot for the metric has longer tails than a normal distribution. Figures 4.12 (b) and (c) show the quantile-quantile plots for one-to-all and all-to-all cases, when taken individually, do represent a normal distribution.

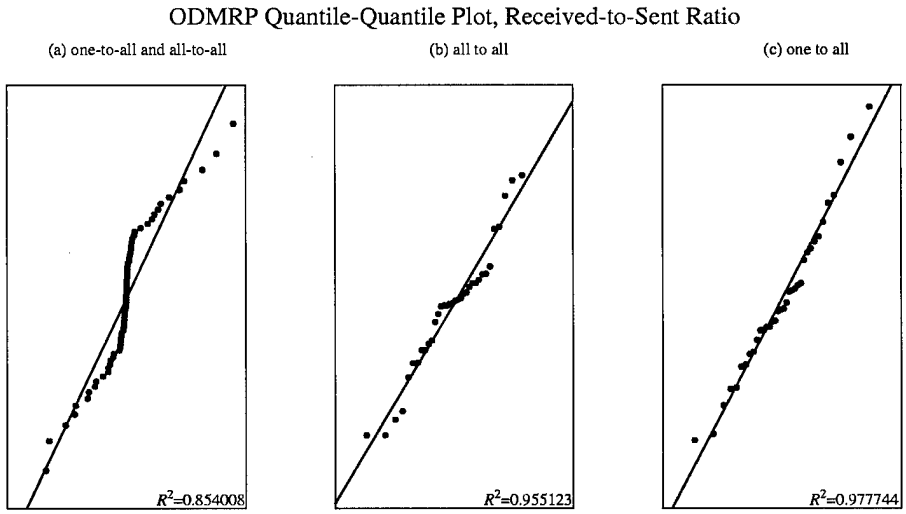

Figure 4.12. ODMRP received-to-sent Quantile-Quantile Plot

This discrepancy occurs because of a bimodality in the received-to-sent metric in the one-to-all transmission configuration. Figure 4.13 shows the different outcomes from two identical trials, with only the random seed changing between the two. The quantile-quantile plot for this configuration, Figure 4.12 (c) shows the errors are normally distributed when examined separately from the all-to-all configuration.

Although the results from the ANOVA analysis are questionable once the residual normality assumption has been broken, the results agrees with what is visually apparent: the protocol variance is primarily influenced by mechanism of transmission. The mechanics of ODMRP quickly explain the source of this influence.

Figure 4.14 presents a visual verification of the inequity in forwarding group membership. Recall from Section 2.4.2.1 that ODMRP sources periodically broadcast JOIN QUERY packets. Receivers of these packets become forwarding group members if they later receive JOIN REPLY from a multicast member with their IP address in the list of

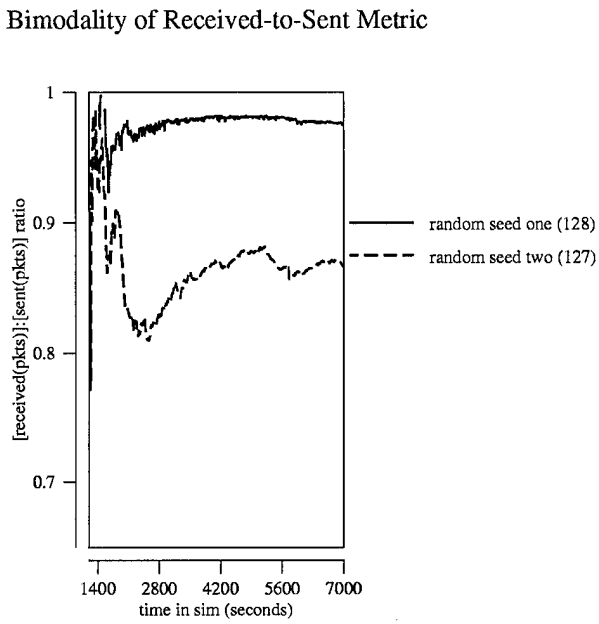

Figure 4.13. ODMRP One-to-All, Received-to-Sent Metric Bimodality

Source-Next field. With more members transmitting JOIN REPLIES, the number of satellites that become forwarding group members increases.

Thus in the example in Figure 4.14, the one-to-all scenario has, on average, less than half the forwarding group members of the all-to-all scenario. With more forwarding group members, the ODMRP mesh obtains more paths from source to destination. If a particular ISL is turned off because of the polar region, there are many alternate routes for the packet to take. The number of dropped packets decreases, and the ratio of received-to-sent increases.

This forwarding group member difference also explains the differences in the data to overhead ratio between the two transmission scenarios. The presence of so many forwarding group members in the all-to-all case generates many extraneous transmissions of packets, which in turn creates a high level of overhead. By keeping the number small, the one-to-all ODMRP scenario manages to obtain a higher data-to-overhead ratio at the expense of received packets.

It is also interesting to note the maximum number of forwarding group members is 71. This is a result of the fact the ground nodes themselves can become a forwarding group

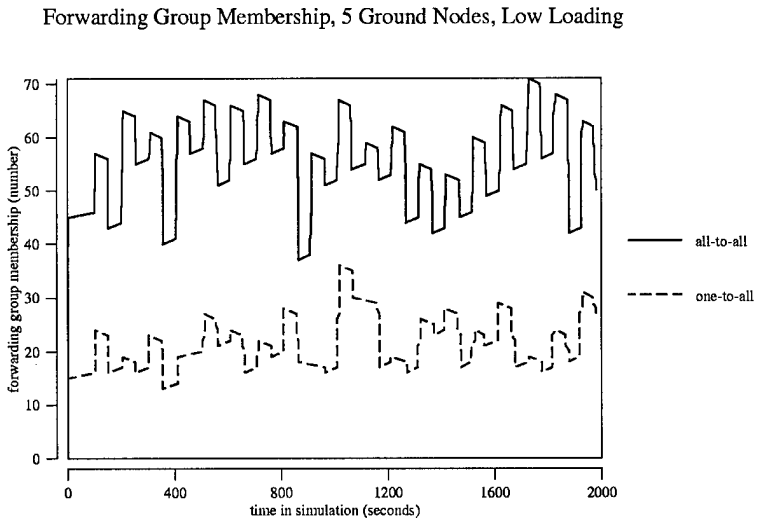

Figure 4.14. ODMRP Forwarding Group Membership

member. Trials show ODMRP uses these ground stations to cross the counter-rotating plane or skip diagonally between both in-plane and inter-plane ISL's.

4.4.3 End-to-End Analysis. Similar trends observed in the DVMRP protocol are found in the ODMRP trials. Again the 10 member case exhibits (with one exception) the smallest end-to-end delay. As the locations were the same for both protocols, this is consistent with the same location skewing argument used before. ODMRP also exhibits a small range in values (less than 15 millisecond difference between the slowest and fastest delays) and narrow confidence intervals are indicative of consistent, stable operation.

Unlike the other metrics, the end-to-end delay has no clear factor of influence. Eighteen percent of the variation comes from the transmission mechanism, 34% the workload, and 34% the membership. Having such an even contribution indicates the metric is sensitive to all three factors.

4.5 Reliability Scenarios

To best analyze the effect of a sudden, complete satellite failure on each of the protocols, the technique described in Section 3.8 was used. By eliminating the most traversed node, the network will be disrupted in potentially the worst case. This critical satellite was calculated for all membership levels.

There are a few parameters that could be associated with the failure timing: the length of the failure, start time of the failure, and time between failures. To achieve steady state results, the failures were started at time t_0 and continued until the simulation terminated. Thus there was no intermittent behavior.

To determine the appropriate t_0, the DVMRP protocol was simulated for three failure start times: 500 seconds, 1000 seconds and 1500 seconds. The results of these trials are shown in Figure 4.15. While at first it may appear there is a dramatic effect on the metrics from the satellite failure, the variation is actually caused by a ground station that is in the footprint of the failed satellite. From approximately the 750 second mark to the 1300 second mark (a period of approximately 10 minutes), a ground station was without an uplink satellite.

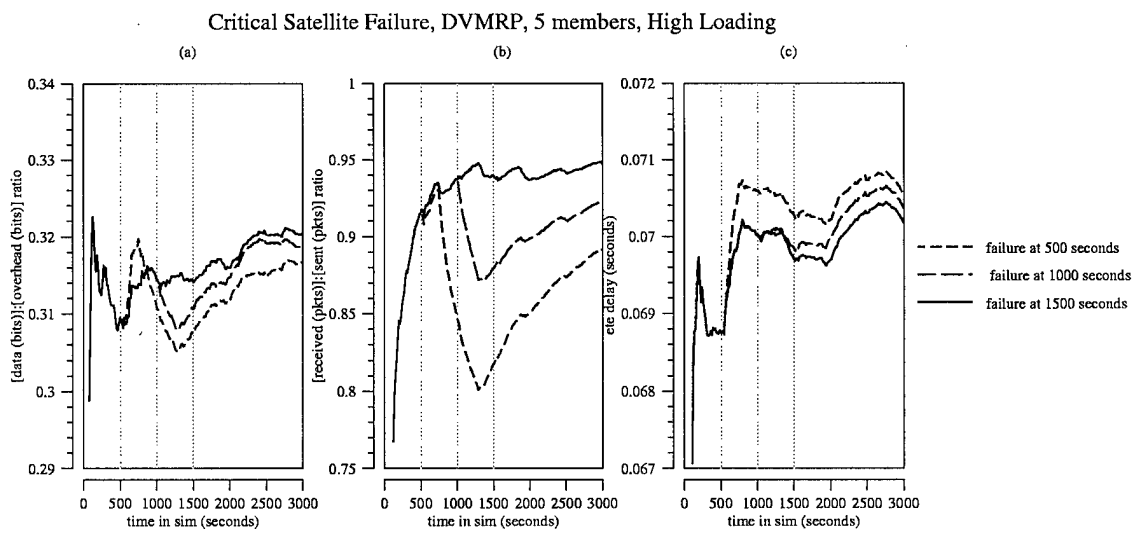

Figure 4.15. Effect of Start Time on DVMRP Satellite Failure

Later simulations showed the 500 second failure, which occurs during the transient period of DVMRP, provided the most variance from the no-fail case. This time was chosen for both the ODMRP and the DVMRP protocols to provide an equal comparison. Since t_0 was chosen in the transient period, and since the satellite failure should introduce a destabilizing effect to the protocol, the metrics were recorded for the entire simulation period. This is in contrast to the previous trials in which only the steady-state period was analyzed.

When comparing the fail case against the no-fail case in Figure 4.16, a slightly higher data-to-overhead ratio was observed. However, ANOVA analysis reveals less than 0.3% of the variance observed comes from the failure. The slight rise in data-to-overhead is attributable to the satellite loss, which means less vector data will need to be transferred over the network.

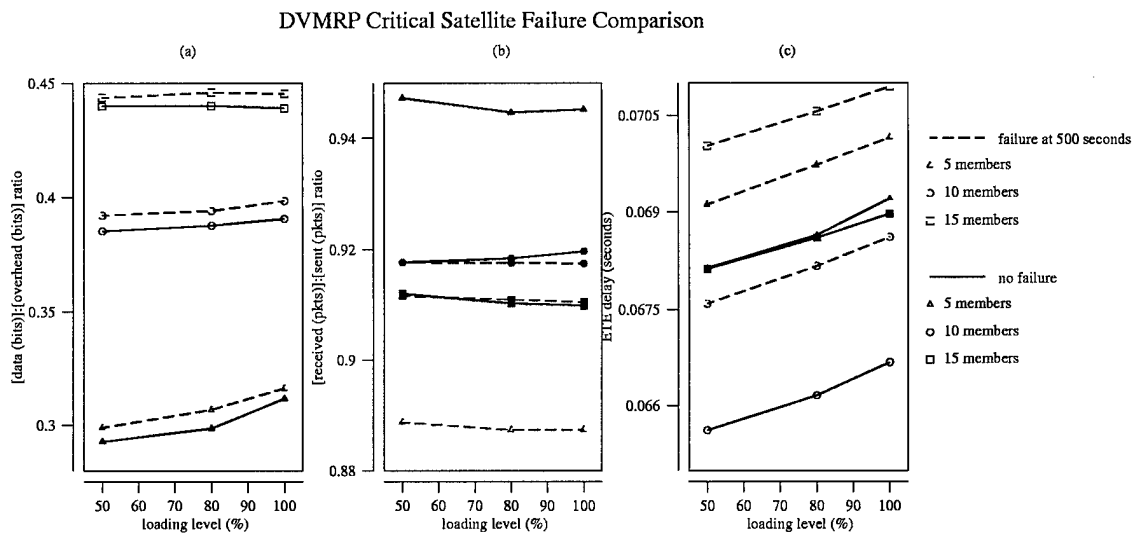

Figure 4.16. DVMRP Critical Failure

The received-to-sent ratio, while appearing to having a third of the variation come from the failure, is misleading. The 5 member group, which has a difference of 0.06 from the no-fail to fail case, is the same group shown in Figure 4.16. The presence of a dead satellite directly above a ground station prevents it from receiving packets, and reduces the received-to-sent ratio. If this did not occur, there would be no statistical difference in the metric between the fail and no-fail cases.

The one DVMRP metric with a strong effect from the failure is the end-to-end delay. Like the received-to-sent ratio, ANOVA indicates approximately a third of the variance comes from the failure, which is verified by the nearly 0.03 second increase in all scenarios. This added delay is a direct effect of the protocol routing around the failure.

Unlike DVMRP, ODMRP exhibited no response to the satellite failure in any metric. Figure 4.17 shows for all scenarios, the data was statistically the same. ANOVA analysis concurs with this, indicating failure does not play a significant role in the variance.

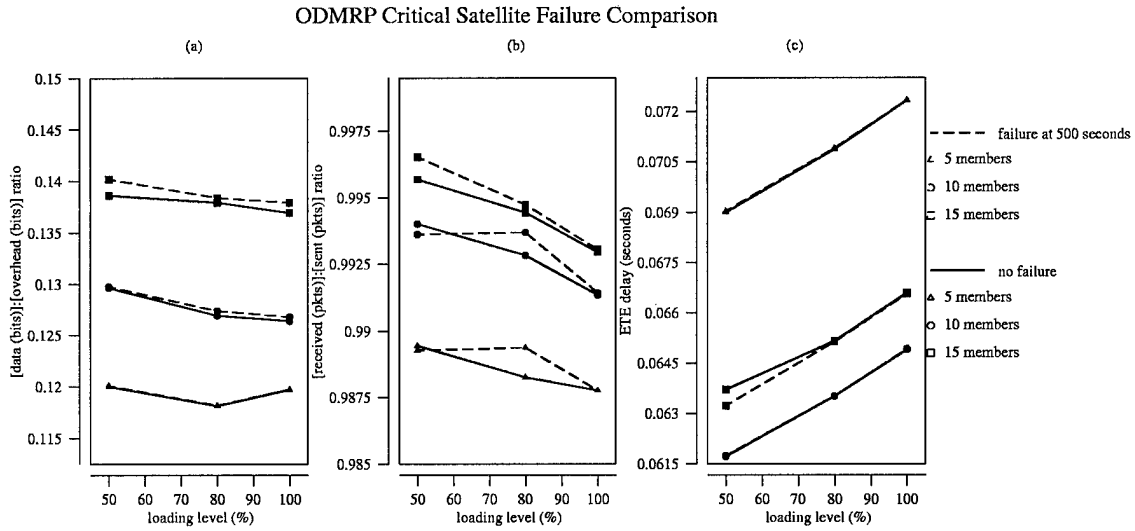

Figure 4.17. ODMRP Critical Failure

Like DVMRP, ODMRP experienced variability when a ground station was in the footprint of a malfunctioning satellite. Figure 4.18 shows the difference in response between a satellite that loses a satellite acting as a forwarding group member.

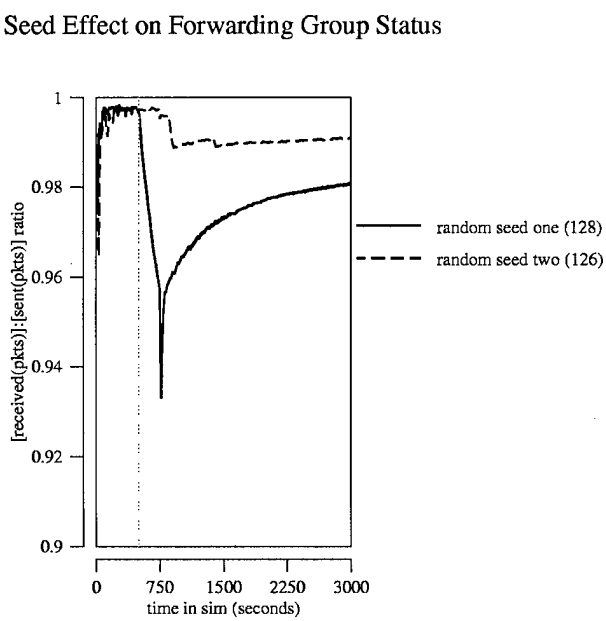

Figure 4.18. Effect of Seed on ODMRP Satellite Failure

These two responses come from the same experiment with different random number seed. The first replication, with random seed 128, sends out a JOIN QUERY message at

some time earlier than 500 seconds. Satellite "a" forwards the message and ends up a forwarding group member. At time 500 seconds, satellite "a" fails. The ground station blindly continues to send packets to satellite "a" and not until the route expires and it sends another JOIN QUERY does it realize that no packets are being received. It then sends JOIN QUERY messages until satellite "b" comes into view, forwards the query and becomes a new forwarding group member.

In contrast, the second experiment broadcasts its JOIN QUERY at a slightly different time and gets satellite "b" immediately. When satellite "a" fails, the ground station is not even aware of it. The difference in the timing for the JOIN QUERY results from the random seed. Although all ODMRP members use the same expiration times for the routes and forwarding group members, this time is the mean value of a normally distributed (with a very small variance) timing outcome.

The reliability of these two protocols is not surprising, given that by the very nature of the constellation, satellites partially "fail" during their time above and below the 60 degree meridians. Adding a complete failure only reduces the coverage by one footprint and eliminates some of the redundancy.

4.6 Protocol Comparison

The ANOVA data presented in Table A.8 verifies the differences between the two protocols in each metric is the result of using different metrics, rather than an effect of some other factor. The only exception is the end-to-end metric, which is discussed in more detail in Section 4.6.3.

4.6.1 Data-to-Overhead Analysis. The data-to-overhead ratio varied widely between ODMRP and DVMRP, with ODMRP having between two to four times as much overhead. The ANOVA analysis confirmed this, indicating 68% of the variance came from the choice of protocol. However, these findings of higher overhead in the ODMRP protocol differ those of Bae, Lee, Su, and Gerla. Specifically, in [24] a comparison between ODMRP and DVMRP shows DVMRP has a data-to-overhead ratio of 5.138 and ODMRP has a

data-to-overhead ratio of 28.4118. This discrepancy come from the difference in definition of what is overhead.

Bae et al. [24], define overhead from the frame of reference of the source. Overhead is therefore any information the source transmits that is not data. This is in contrast to the system-wide frame of reference used for this study, where overhead is any data packet that does not end up being received by a multicast member. By the definition of Bae, only the overhead from control packets JOIN REQUEST and JOIN REPLY count as overhead. When this is compared with the amount of overhead in the FLASH UPDATE, GRAFT, and PRUNE packets, as well as the entire Mobile IP scheme, the amount of data from a DVMRP/Mobile IP source is large.

From the system frame of reference, DVMRP's significantly lower data-to-overhead ratio is caused by the same properties that made ODMRP lower in Bae's source-based frame of reference. The on-demand aspect of the protocol is unlike DVMRP, in that instead of paying a fixed cost to determine routing data, ODMRP pays the cost to route over every single transmission. Figure 4.19 shows a sample unicast transmission utilizing both DVMRP 4.19a and ODMRP 4.19b. In this figure, F represents a forwarding group member, S the source, R the receiver, and V a satellite with distance vector information.

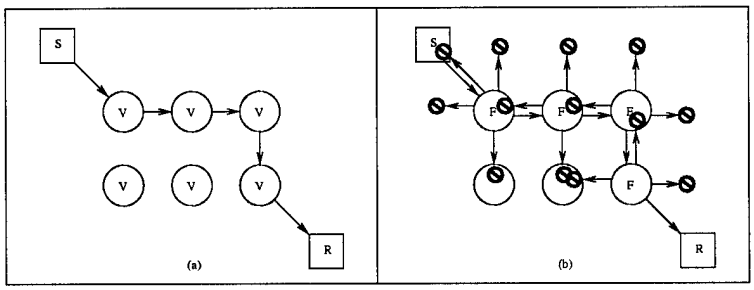

Figure 4.19. Overhead comparison of ODMRP, DVMRP

In 4.19b, the circles with a line through them indicate packets forwarded but destroyed. In the case of the non-forwarding group members, these packets were destroyed because of their state as non-forwarding group members. The forward group members destroy the packet because the packet has already been received, whereas the DVMRP protocol in 4.19a had no overhead involved in the actual transmission 12 of the 17 packet

transmissions in the ODMRP cases were pure overhead. Thus, this explains both the reason for the lower ratios, as well as the tendency for the ratio to actually decrease as loading level increase.

4.6.2 Received-to-Sent Analysis. None of the factors have a large effect on the received-to-sent ratio alone. However, when the protocol is combined with the mechanism of transmission, 70% of the variance is explained. This interaction is the single most influential factor in determining the number of packets successfully received.

While the on-demand aspect of ODMRP may cost it in terms of efficiency, it has a profound effect on it in terms of reliability. DVMRP is dependent on maintaining fresh vector routing to deliver packets. If a vector becomes outdated during a packet transmission, the information is lost. In contrast, ODMRP is capable of devising a new route because of it's on-demand and mesh-based features. The more satellites acting as forwarding group members, the better the protocol is able to correct from unexpected route failures.

Since ODMRP performs more reliably with more forwarding group members, it has a higher ratio of received-to-sent when the all-to-all communication method is used. DVMRP, in contrast, performs better when the system is multicasting from a single source. As such, there is a difference of 5 to 10% between ODMRP and DVMRP in terms of packets delivered correctly.

4.6.3 End-to-End Analysis. The ANOVA values show of all the metrics, end-to-end delay was least dependent on the protocol. Fully 78% of the variation was attributable to causes outside of the protocol chosen. However, 12% of the variance was from the interaction between the protocol and other factors. This interaction is where the difference between DVMRP and ODMRP appears.

While both DVMRP and ODMRP show similar delay values, ODMRP exhibits a much stronger response to loading than the DVMRP model. This is illustrated in Figure 4.20.

In comparison with the DVMRP protocol, delay associated with ODMRP increases 2 to 9 milliseconds more than DVMRP over the same workload increase. This trend

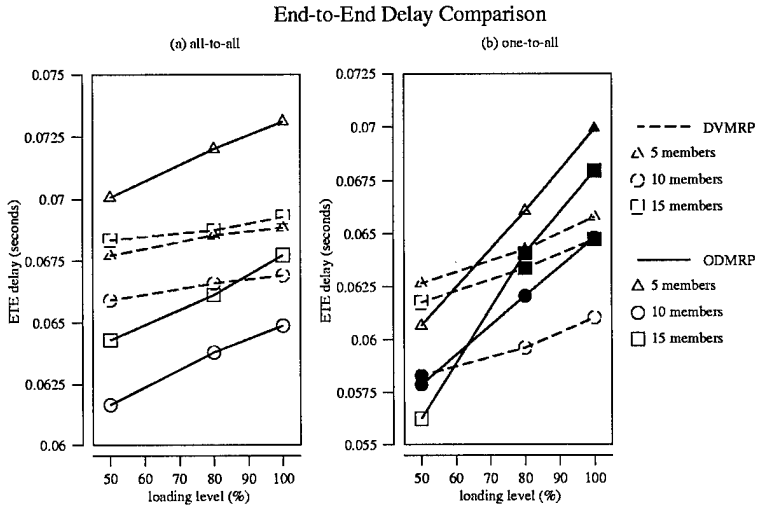

Figure 4.20. Comparison of low membership level end-to-end delay metrics

culminates in four of the six delays at the 100% workload being higher for ODMRP than DVMRP. This is despite the fact that at the 50% workloads the five of the six ODMRP scenarios have lower delays than DVMRP.

This difference in sensitivity to workload is associated with the overhead explained in Section 4.6.1. ODMRP responds to workload increases by sending even more data as overhead. This is due to the policy of forwarding group members transmitting on all outgoing interfaces regardless of the forwarding state of the receiver.

4.6.4 Conclusions. Both DVMRP and ODMRP have strengths. For use of bandwidth, the DVMRP protocol outperforms ODMRP by over 200% in every case. The ability to selectively choose the next hop of a multicast route requires less overhead than the mini-broadcast ODMRP performs at every hop. Even with the additional overhead of determining the shortest route from every source to every destination, DVMRP maintains a data-to-overhead ratio that increases as more subscribers join the multicast group.

The extra overhead of ODMRP, however, pays off well when it comes to it's nearly 100% reliability. With many sources, the replication of the mesh-based architecture provides enough redundancy that nearly every packet created is successfully delivered. Similarly, the protocol is not affected by a single critical failure, so long as the satellite doesn't provide critical connectivity for a ground station to the satellite network.

Both protocols provide adequate performance in the end-to-end metric. ODMRP has a slight advantage at low membership and low loading levels, while the lower response of DVMRP to loading and membership level gives it an edge at higher loading and membership levels.

V. Conclusions

5.1 Restatement of Research Goal

The focus of the research was to perform a comparative analysis of two multicasting protocols, one optimized to work in a fully mobile, ad hoc network and the other configured for a fixed-topology network with mobile components. These protocols are examined under various group membership, density and loading levels as well as satellite failures. In particular, the Distance Vector Multicast Routing Protocol (DVMRP) utilizing mobile IP is compared against the On Demand Multicast Routing Protocol (ODMRP) ad hoc multicasting protocol

5.2 Research Contributions

This work is the first to analyze DVMRP and ODMRP in a Low Earth Orbit (LEO) satellite environment. It is also the first to examine these two protocols against each other in a large complicated topology with high workloads. The work follows on to previous research into LEO constellation routing by Fossa, Janoso and Pratt [5, 6, 4] as well as mobile IP multicast research by Muller [7].

5.3 Conclusions

Neither protocol clearly outperformed the other in all situations. Depending on the LEO constellation application, either protocol can be an acceptable choice. For small groups where reliability is a high priority, such as real-time tactical intelligence, ODMRP will reliably transmit data throughout the multicast group. By maintaining a greater than 99% received-to-sent ratio, ODMRP is the clear choice for applications that demand reliability. However, the system-wide overhead of more than 8 overhead bits for every 1 data bit makes it a high bandwidth solution

For groups with a larger membership or with less of a demand of reliability, DVMRP seems a logical choice. Group video or voice conferencing could be provided with a lower overhead than ODMRP. DVMRP is also ideal for multicasting in one-to-all scenarios, a weakness in ODMRP. Since the data-to-overhead ratio has a trend to increase logarith-

mically with membership, it is more scalable than ODMRP. Additionally, the end-to-end delay is not effected by levels, loading and transmission mechanism, with less than a 5% spread from the lowest to highest delay.

In time of crisis, both protocols can survive damage to the constellation so long as the failed satellite's footprint does not provide the only link for a ground station into the constellation. The ODMRP protocol has no statistical difference between the failure and non-failure modes, whereas the DVMRP protocol only has an increase in the end-to-end delay.

5.4 Future Research

There were many areas of potential future research brought up by this study. The most significant would involve modifying ODMRP or DVMRP to optimize their performance in a LEO constellation. Further examination of alternate multicasting protocols would also prove beneficial, as well as examining alternate mobile IP solutions, such as tunneling and MoM. Furthermore, a more thorough analysis of the timings for both protocols could have a profound effect on the metrics. This is one area the research did not cover completely.

The area of reliability under satellite failure should be pursued more vigorously, as the strength of ODMRP seems to be in the fault tolerance built into the routing mesh it employs. Failures in multiple satellites would show whether this approach is effective in more stressful scenarios.

5.4.1 ODMRP Modifications. There are several obvious ODMRP modifications that could improve the performance of ODMRP in a LEO environment. The most obvious would be to not send the packets to the same interface from where they were received. In an IEEE 802.11 network, which ODMRP was designed around, there is only one outgoing interface – the broadcast interface. In the satellite constellation, every incoming ISL is also an outgoing ISL. Thus it makes sense not to send packets back to the source, as the packets will simply be destroyed.

Another modification would be a "poison packet" from a receiver of a packet. If that receiver was not a forwarding group member, it would reply with a poison packet that would stop the source from forwarding on that interface for a set amount of time. This would exploit the large amount of time that the network is static. The amount of time until the poison wears off would need to be synchronized with the forwarding group update times, otherwise the mesh that the ODMRP creates would be weakened.

5.4.2 DVMRP Modification. The boundary between the DVMRP protocol and the Mobile IP is potentially an area of future research. Modifying DVMRP to dynamically co-locate with the ground stations could eliminate the need for mobile IP's mobility management. This modification has the potential to increase the performance of the protocol.

Appendix A. Data

Table A.1. DVMRP, All to All, Low Membership, Urban Areas

Members	Loading level	**Data to Overhead**		**Received to Sent**		**End to End Delay**	
		μ	σ	μ	σ	μ	σ
5	50	0.2918	0.0042	0.9355	0.0065	0.0677	0.0003
	80	0.3004	0.0041	0.9435	0.0108	0.0685	0.0005
	100	0.3102	0.0083	0.9354	0.012	0.0688	0.0007
10	50	0.3954	0.0022	0.8879	0.0028	0.0659	0.0002
	80	0.3986	0.002	0.891	0.0026	0.0666	0.0001
	100	0.4006	0.0042	0.8896	0.0038	0.0669	0.0003
15	50	0.4444	0.0016	0.8915	0.0032	0.0683	0.0001
	80	0.4449	0.0031	0.8871	0.0038	0.0687	0
	100	0.4455	0.0047	0.8908	0.0046	0.0693	0.0003

Table A.2. DVMRP, One to All, Low Membership, Urban Areas

Members	Loading level	**Data to Overhead**		**Received to Sent**		**End to End Delay**	
		μ	σ	μ	σ	μ	σ
5	50	0.2556	0.0031	0.9778	0.0062	0.0627	0.0004
	80	0.2697	0.0053	0.9787	0.003	0.0643	0.0007
	100	0.2908	0.0102	0.9844	0.0064	0.0658	0.0009
10	50	0.3446	0.0018	0.9556	0.001	0.0583	0.0004
	80	0.347	0.0033	0.9567	0.005	0.0596	0.001
	100	0.3538	0.0059	0.9563	0.0042	0.061	0.0008
15	50	0.3714	0.0014	0.9584	0.0049	0.0618	0.0001
	80	0.372	0.0018	0.958	0.0023	0.0634	0.0002
	100	0.3728	0.0016	0.9572	0.0017	0.0647	0.0002

Table A.3. DVMRP, All to All, High Membership, Urban Areas (Sparse)

Members	Loading level	Data to Overhead		Received to Sent		End to End Delay	
		μ	σ	μ	σ	μ	σ
40	50	0.5392	0.0013	0.9094	0.0017	0.0675	0.0001
	80	0.5389	0.0022	0.9073	0.0029	0.0679	0.0001
	100	0.5393	0.0016	0.9078	0.0039	0.0683	0.0001
60	50	0.5978	0.0009	0.916	0.0014	0.0677	0.0001
	80	0.5985	0.0012	0.9136	0.0035	0.0681	0.0002
	100	0.5978	0.0032	0.9122	0.0031	0.0686	0.0002
80	50	0.6364	0.0032	0.9125	0.0033	0.0685	0.0001
	80	0.6375	0.0038	0.9105	0.0038	0.069	0.0001
	100	0.6374	0.0029	0.9094	0.0049	0.0691	0.0001

Table A.4. DVMRP, All to All, High Membership, Even Distribution (Dense)

Members	Loading level	Data to Overhead		Received to Sent		End to End Delay	
		μ	σ	μ	σ	μ	σ
40	50	0.6489	0.0016	0.8546	0.0028	0.0681	0
	80	0.6541	0.0051	0.8565	0.0025	0.0686	0.0002
	100	0.6483	0.0051	0.8451	0.0144	0.0684	0.0003
60	50	0.7232	0.0008	0.891	0.0018	0.0694	0.0012
	80	0.7232	0.0007	0.8903	0.0022	0.0696	0.0006
	100	0.7254	0.0018	0.8907	0.0036	0.07	0.0006
80	50	0.7629	0.0024	0.8634	0.0015	0.0773	0.0041
	80	0.7646	0.0027	0.8622	0.0097	0.0753	0.0027
	100	0.7664	0.0033	0.8627	0.0086	0.0753	0.0026

Table A.5. ODMRP, All to All, High Membership, Urban Areas

Members	Loading level	Data to Overhead		Received to Sent		End to End Delay	
		μ	σ	μ	σ	μ	σ
5	50	0.1143	0.0042	0.9966	0.0006	0.0701	0.0003
	80	0.1114	0.0038	0.9916	0.0056	0.072	0.0007
	100	0.1129	0.0055	0.9918	0.0024	0.0731	0.0008
10	50	0.1191	0.0023	0.9964	0.0002	0.0617	0.0002
	80	0.1154	0.0022	0.9951	0.0003	0.0638	0.0002
	100	0.1143	0.0007	0.9934	0.0018	0.0649	0.0008
15	50	0.1242	0.0009	0.9954	0.0013	0.0643	0.0001
	80	0.1225	0.0002	0.9937	0.0019	0.0661	0.0002
	100	0.1224	0.0003	0.9918	0.0019	0.0677	0.0002

Table A.6. ODMRP, One to All, Low Membership, Urban Areas

Members	Loading level	Data to Overhead		Received to Sent		End to End Delay	
		μ	σ	μ	σ	μ	σ
5	50	0.2215	0.0071	0.8658	0.0461	0.0607	0.0018
	80	0.2314	0.0071	0.9029	0.039	0.0661	0.0027
	100	0.2391	0.0095	0.9069	0.0439	0.0699	0.0032
10	50	0.2391	0.0192	0.9233	0.0213	0.0579	0.0011
	80	0.2339	0.012	0.9026	0.0154	0.0621	0.0011
	100	0.2395	0.0078	0.928	0.0251	0.0648	0.0043
15	50	0.2781	0.0039	0.9289	0.0106	0.0562	0.0005
	80	0.2508	0.0052	0.9186	0.0132	0.0641	0.0016
	100	0.2423	0.0065	0.9288	0.0249	0.068	0.0016

Table A.7. DVMRP Failure Data

State	Members	Loading level	Data to Overhead		Received to Sent		End to End Delay	
			μ	σ	μ	σ	μ	σ
Fail	5	50	0.2988	0.0023	0.8887	0.0025	0.0691	0.0002
		80	0.3067	0.0016	0.8873	0.0016	0.0697	0.0002
		100	0.3161	0.0016	0.8874	0.0033	0.0702	0.0003
	10	50	0.3919	0.0023	0.9176	0.0027	0.0676	0.0001
		80	0.3940	0.0028	0.9175	0.0015	0.0682	0.0001
		100	0.3983	0.0014	0.9175	0.0026	0.0686	0.0002
	15	50	0.4437	0.0040	0.9116	0.0018	0.0700	0.0001
		80	0.4459	0.0021	0.9109	0.0016	0.0706	0.0001
		100	0.4454	0.0035	0.9105	0.0019	0.0709	0.0001
No Fail	5	50	0.2927	0.0027	0.9473	0.0036	0.0681	0.0003
		80	0.2985	0.0033	0.9447	0.0022	0.0686	0.0002
		100	0.3115	0.0039	0.9452	0.0036	0.0692	0.0003
	10	50	0.3851	0.0018	0.9177	0.0017	0.0656	0.0001
		80	0.3875	0.0009	0.9184	0.0018	0.0662	0.0000
		100	0.3905	0.0026	0.9196	0.0016	0.0667	0.0001
	15	50	0.4399	0.0016	0.9120	0.0025	0.0681	0.0001
		80	0.4403	0.0027	0.9103	0.0024	0.0686	0.0001
		100	0.4390	0.0024	0.9099	0.0023	0.0690	0.0001

Table A.8. ODMRP vs DVMRP ANOVA

* - not significant according to F-test		**data to overhead**	**received to sent**	**end to end**
Main effects	Protocol	67.7 %	4.84 %	0 %*
	Mechanism	2.96 %	1.71 %	34.5 %
	Workload	0.03 %	0.18 %* %	17 %
	Membership	7.32 %	1.76 %	22.52 %
Second order	Protocol-Mechanism	16.59 %	70.41 %	1.05 %
	Protocol-Workload	0.09 %	0.19 %	4.36 %
	Protocol-Membership	4.37 %	7.61 %	4.12 %
	Mechanism-Workload	0.01 %*	0.36 %*	3.68 %
	Mechanism-Membership	0.08 %	1.99 %	0.3 %*
	Workload-Membership	0.16 %	0.24 %*	0.35 %*
Third order	Protocol-Mechanism-Workload	0.01 %*	0.2 %*	0.9 %
	Protocol-Mechanism-Membership	0.33 %	0.04 %*	3.59 %
	Mechanism-Workload-Membership	0.07 %	0.26 %*	0.39 %*
	Protocol-Workload-Membership	0.01 %*	0.13 %*	0.38 %*
Fourth order	Protocol-Mechanism-Workload-Membership	0.03 %	0.24 %*	0.27 %*
	Unaccounted	0.25 %	9.85 %	6.56 %

Table A.9. Low Membership DVMRP ANOVA

* - not significant according to F-test		**data to overhead**	**received to sent**	**end to end**
Main effects	Mechanism	18.68 %	71.34 %	68.32 %
	Workload	0.68 %	0.03 %*	6 %
	Membership	77.1 %	23.51 %	19.7 %
Second order	Mechanism-Workload	0.07 %	0.04 %*	1.41 %
	Mechanism-Membership	2.38 %	2.91 %	2.76 %
	Workload-Membership	0.59 %	0.11 %*	0.02 %*
Third order	Mechanism-Workload-Membership	0.06%*	0.21 %*	0.03 %*
	Unaccounted	0.44 %	1.85 %	1.76 %

Table A.10. ODMRP ANOVA

* - not significant according to F-test		**data to overhead**	**received to sent**	**end to end**
Main effects	Mechanism	95.97 %	78.39 %	18.04 %
	Workload	0.12 %*	0.59 %*	29.58 %
	Membership	1.46 %	1.83 %*	30.36 %
Second order	Mechanism-Workload	0.02%*	0.9 %*	6.27 %
	Mechanism-Membership	0.3 %	1.67 %*	4.51 %
	Workload-Membership	0.51 %	0.55 %*	1.12 %*
Third order	Mechanism-Workload-Membership	0.55 %	0.72 %*	1 %*
	Unaccounted	1.07 %	15.34 %	9.12 %

Table A.11. High Membership DVMRP ANOVA

* - not significant according to F-test		**data to overhead**	**received to sent**	**end to end**
Main effects	Density	65.63 %	73.57 %	24.14 %
	Workload	0.01 %*	0.28 %*	0.1 %*
	Membership	34 %	14.12 %	36.69 %
Second order	Density-Workload	0 %*	0.06 %*	0.67 %*
	Density-Membership	0.24 %	7.98 %	22.34 %
	Workload-Membership	0.01 %*	0.16 %*	0.91 %*
Third order	Density-Workload-Membership	0.01 %*	0.32 %*	0.74 %*
	Unaccounted	0.1 %	3.5 %	14.42 %

Table A.12. DVMRP Failure ANOVA

* - not significant according to F-test		**data to overhead**	**received to sent**	**end to end**
Main effects	Failure	0.29 %	32.43 %	33.24 %
	Workload	0.33 %	0.06 % *	8.15 %
	Membership	98.85 %	3.33 %	55.08 %
Second order	Failure-Workload	0 % *	0.01 % *	0.01 % *
	Failure-Membership	0 % *	62.61 %	2.51 %
	Workload-Membership	0.29 %	0.1 % *	0.05 % *
Third order	Failure-Workload-Membership	0.01 % *	0.03 % *	0.01 % *
	Unaccounted	0.23 %	1.44 %	0.96 %

Table A.13. ODMRP Failure ANOVA

		data to overhead	received to sent	end to end
Main effects	Mechanism	0.09 % *	0.7 % *	0 % *
	Workload	1.29 %	7.42 % *	14.71 %
	Membership	95.55 %	31.41 %	84.71 %
Second order	Mechanism-Workload	0.01 % *	0.2 % *	0.01 % *
	Mechanism-Membership	0.08 % *	3.01 % *	0.02 % *
	Workload-Membership	0.44 % *	0.72 % *	0.01 % *
Third order	Mechanism-Workload-Membership	0.05 % *	0.33 % *	0.02 % *
	Unexplained/error	2.49 %	56.21 %	0.51 %

Bibliography

[1] S. Deering, "Host extensions for IP multicasting." RFC 1112, August 1989.

[2] S.-J. Lee, W. Su, and M. Gerla, "On-demand multicast routing protocol (ODMRP) for ad hoc networks." `draft-ietc-manet-odmrp-02.txt`, January 2000.

[3] T. Pusateri, "Distance vector multicast routing protocol." `draft-ietf-idmr-dvmrp-v3-10.txt`, August 2000.

[4] S. R. Pratt, "A performance analysis of dynamic routing algorithms in an Iridium-like low earth orbit satellite system," Master's thesis, Air Force Institute of Technology, 1999. AFIT/GCE/ENG/99M-04.

[5] C. E. Fossa, "A performance analysis of the iridium low earth orbit satellite system," Master's thesis, Air Force Institute of Technology, 1998. AFIT/GCE/ENG/98M-04.

[6] R. F. Janoso, "Performance analysis of dynamic routing protocols in low earth orbit satellite data network," Master's thesis, Air Force Institute of Technology, 1996. AFIT/GE/ENG/96D-08.

[7] A. Muller, "A comparitive analysis of proposed mobility support schemes for IP multicast," Master's thesis, Air Force Institute of Technology, 2000. AFIT/GCS/ENG/00J-02.

[8] M. Ramalho, "Intra– and inter– domain multicasting routing protocols: A survey and taxonomy," *IEEE Communications: Surveys and Tutorials*, January-March 2000.

[9] J. D. Solomon, *Mobile IP: the Internet Unplugged.* Upper Saddle River, New Jersey: Prentice Hall, 1998.

[10] T. G. Harrison, C. L. Williamson, W. L. Mackrell, and R. B. Bunt, "Mobile multicast (MoM) protocol: Multicast support for mobile hosts," in *MOBICOM 97*, pp. 151–160, 1997.

[11] W. Fenner, "Internet group management protocol, version 2." RFC 2236, November 1997.

[12] S. Deering and J. Smith, "Protocol independent multicasting version 2: Dense mode specification." `draft-ietc-pim-v2-dm-03.txt`, March 1999.

[13] J. Moy, "Multicast extensions to OSPF (MOSPF)." RFC 1584, March 1994.

[14] T. Ballardie, P. Francis, and J. Crowcroft, "Core based trees (CBT) - an architecture for scalable interdomain multicast routing," in *Proceedings of ACM SIGCOMM'93*, pp. 85–95, 1993.

[15] C. Perkins, "IP mobility support." RFC 2002, October 1996.

[16] C. Perkins, "IP encapsulation within IP." RFC 2003, October 1996.

[17] C. Perkins, "Minimal encapsulation within IP." RFC 2004, October 1996.

[18] J. D. Solomon, "Applicability statement for IP mobility support." RFC 2005, October 1996.

[19] D. Cong, M. Hamlen, and C. Perkins, "The definitions of managed objects for IP mobility support using SMIv2." RFC 2006, October 1996.

[20] M. S. Corson, "Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations." RFC 2501, January 1999.

[21] G. Xylomenos and G. C. Polyzos, "IP multicast for mobile hosts," *IEEE Communications Magazine*, pp. 54–58, January 1997.

[22] V. Chickarmane, C. L. Williamson, R. B. Bunt, and W. Mackrell, "Multicast support for mobile hosts using mobile IP: Design issues and proposed architecture," *Mobile Networks and Applications*, vol. 3, no. 4, 1998.

[23] J. Broch, D. Maltz, D. Johnson, Y.Hu, and J.Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *MOBICOM '98*, pp. 85–97, 1998.

[24] 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," *IEEE Network*, pp. 70–77, January/February 2000.

[25] S.-J. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia, "A performance comparison study of ad hoc wireless multicast protocols," in *Proceedings of IEEE Infocom 2000*, 2000.

[26] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade, "AMRoute: Ad hoc multicast routing protocol." `draft-ietc-manet-amroute-00.txt`, August 1998.

[27] W. W. Wu, E. F. Miller, W. L. Pritchard, and R. L. Pickholtz, "Mobile satellite communications," *Proceedings of the IEEE*, pp. 1431–1445, September 1994.

[28] E. D. Re, R. Fantacci, and G. Giambene, "Efficient dynamic channel allocation techniques with handover queuing for mobile satellite networks," *IEEE Journal on Selected Areas in Communication*, pp. 397–404, February 1995.

[29] R. Cheng and Kumar, "A loop free Bellman Ford routing protocol without bouncing effect," in *SIGCOMM '89*, pp. 224–236, 1989.

[30] K. Tsai and R. Ma, "Darting: A cost effective routing alternative for large space-based dynamic topology networks," in *MILLCOM '95*, pp. 682–687, 1995.

[31] K. Chua, Y. Li, and C. Foo, "On a Linux implimentation of mobile IP and its effects on TCP performance," *Computer Communications*, no. 22, pp. 568–588, 1999.

[32] Y. C. Hubbel, "A comparison of the Iridium and AMPS systems," *IEEE Network*, vol. 11, pp. 52–59, March-April 1997.

[33] V. Paxson and S. Floyd, "Wide-area traffic: The failure of poisson modeling," *IEEE/ACM Transactions on Networking*, pp. 226–244, June 1995.

[34] S. McCreary and K. Claffy, "Trends in wide area IP traffic patterns." `http://www.caida.org/outreach/papers/AIX0005/`, May 2000.

[35] K. Park, G. Kim, and M. Crovella, "The relationship between file sizes, transport protocols, and self-similar network traffic," in *Proceedings of IEEE International Conference on Network Protocols*, pp. 171–180, October 1996.

[36] G. M. Comparetto, "A technical comparison of several global mobile satellite communications systems," *Space Communications*, vol. 11, no. 3, pp. 97–104, 1993.

[37] S. Deering, D. L. 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, pp. 153–162, April 1996.

[38] R. Jain, *The Art of Computer Systems Performance Analysis.* New York: John Wiley and Sons, 1991.

[39] D. C. Montgomery, *Design and Analysis of Experiments.* New York, New York: John Wiley and Sons, 1997.

Vita

Second Lieutenant Ryan W. Thomas was born in Portland, Oregon. He received his commission through AFROTC after graduating with honors and a Bachelors of Science in Engineering from Harvey Mudd College in Claremont, California. He is a member of Tau Beta Pi and Eta Kappa Nu. His first assignment in the Air Force was to the Air Force Institute of Technology. As a follow-on assignment, he will be working for the Air Force Research Lab at Hanscom AFB, MA.

Permanent address: 2950 P Street
Wright Patterson AFB, OH 45433