Performance Evaluation of Routing Algorithms for Multicast Traffic

A Thesis Submitted to the College of

Graduate Studies and Research

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in the

Department of Computer Science

University of Saskatchewan

Saskatoon, Saskatchewan

by

Timothy George Harrison

August, 1995

PERMISSION TO USE

Head of the Department of Computer Science
University of Saskatchewan
Saskatoon, Saskatchewan, Canada
S7N 0W0

ABSTRACT

As network and internetwork usage continues to grow, a need is emerging to be able to multicast data transmissions efficiently to arbitrary groups of users. Multicasting is the selective broadcast of a message from a sender to a group of recipients that is a subset of all the potential recipients in a network. The main advantages of multicasting over broadcasting include conservation of network bandwidth, restriction of data transmission to paying subscribers, and control of distribution of sensitive information.

The choice of routing algorithm used in connecting multicast group recipients to the source has a significant impact on network link usage and on the probability of a successful multicast connection being made. Simplistic routing algorithms tend to waste network bandwidth in a multicast environment. Shared link algorithms, which route multicast calls along common link channels when possible, are generally more efficient.

This thesis evaluates the call blocking probability and link usage performance of several shared link and non-shared link multicast routing algorithms in a network environment, using simulation. It was found that the non-shared link routing algorithms did not scale well with network size or recipient group size. Link sharing was found to be an important heuristic that improves call blocking performance and reduces biases against large multicast groups. However, link sharing reduces multicast call blocking at the expense of increased unicast call blocking. Consideration must be given to service provisioning when multicast calls are introduced into a network, particularly with shared link routing algorithms.

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Carey Williamson, for his guidance and support during this research endeavor. I would also like to thank the other members of my committee, Mark Keil, Winfried Grassmann, and Wayne Grover (external examiner from TRLabs Edmonton) for their helpful input for both the proposal and final version of the thesis. Thanks also go to TRLabs Saskatoon for the use of their computing and office facilites for the research and preparation of the thesis.

My parents, Albert and Iris Sloboda, instilled in me the values of hard work. Without this teaching and their support, I would not have come this far.

Financial assistance was gratefully received from my parents, and from TRLabs.

Proverbs 3:5-6 KJV

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

As the era of the Information Superhighway draws closer with the increasing ubiquity of the Internet, there is appearing a need to be able to *multicast* data transmissions to arbitrary groups of users. Multicasting is the *selective* broadcast of a message from a sender to a group of recipients that is a subset of all the potential recipients in a network. In contrast, *broadcasting* is the transmission of a message from a sender to all possible recipients in a network, regardless of whether or not any of them wish to receive the information.

Multicasting has three advantages over broadcasting, particularly when the recipient group size is significantly smaller than the group of all possible recipients in a network. First, by routing a message only where it needs to go, multicasting conserves network bandwidth, which allows more efficient use of the network infrastructure, and can result in user toll charges that are lower than for a broadcast delivery. Second, data transmission can be restricted to only the paying subscribers for services such as video on demand. Third, the dissemination of sensitive information can be limited to a select group of recipients.

Multicasting from a single source to multiple recipients is referred to as *point to multipoint multicasting* or *single multicasting.* Multicasting that involves more than one sender in a group is called *multipoint to multipoint multicasting* or *group multicasting.* Some simple examples of multicasting include pay-per-view CATV telecasts (single multicasting), audio teleconferencing (group multicasting), and distance education (single or group multicasting). In recent years the Multicast Backbone (MBONE), a virtual multicast network layered on top of the Internet, has played host to audio and video teleconferencing of the Internet Engineering Task Force (IETF) meetings, using multicast [5, 6].

1.1 Performance Considerations of Multicasting

The versatility that multicasting provides is not without cost. Computational resources and network resources are required to set up multicast groups. The survey on network multicasting done by Frank *et al.* [16] outlines several cost criteria involved in routing packets from a source to a set of recipients. For single multicasting the following criteria are considered desirable:

- Low call setup time (*i.e.,* low complexity of the routing algorithm).
- Low cost of storing group state information (*e.g.,* group membership information) at either the sender or at the network routers, or even within the address fields of the individual data packets.
- Low path cost in terms of number of network links traversed (bandwidth used).
- Low delay in the time from when a packet is first transmitted by the source until the last copy is received by the last recipient. Low delay is desirable in time sensitive applications, such as audio transmission.

Of these, the path cost of multicast routing algorithms will be studied in this thesis.

The simplest solution to the multicast problem is to replicate the sender's transmission and send a copy to each recipient. This method is referred to as the N-Unicast technique. The Internet, for example, uses this technique for sending a single electronic mail message to multiple recipients via the Simple Mail Transfer Protocol (SMTP) [21]. The telephone industry handles a very similar problem in conducting conference calls by having a dedicated conference bridge serve as a centralized location where conference group members dial and interconnect [24].

Recent work in the area of Internet Protocol (IP) multicast has been done by Deering [10, 11, 12, 13]. Deering proposes several algorithms for multicast communication in arbitrary networks, where the network in question is represented as a graph, and trees of connectivity are generated by a variety of growing and pruning techniques.

Ballardie *et al.* [2] propose Core Based Trees (CBT) for efficient network-level multicast. Core based trees provide a different approach to setting up the multicast delivery tree by generating multiple delivery trees from some chosen point in the network that is relatively central to all the members of the multicast group. The advantage over using a delivery tree based at a single sender is the ease of implementing multipoint to multipoint teleconferencing. Core Based Trees offer link usage savings as well.

1.2 Future Considerations

Future uses of multicasting will include extensions of existing applications. For example, high quality video teleconferencing and multipoint interactive multimedia applications are expected to appear once large bandwidth connections become cost effective. Also, multidatabase applications requiring large bandwidths such as interactive seismic imaging are also candidates for large bandwidth multicasting [7].

The expected network environment for future large bandwidth data transfer is the Broadband Integrated Services Digital Network (BISDN) using Asynchronous Transfer Mode (ATM) [23]. ATM is a connection-oriented protocol that is well suited to handling bursty traffic on high speed networks. ATM is a high bandwidth, low delay packet switching technique that can support a wide variety of connection oriented and connectionless services. With ATM and new transmission infrastructures such as the Synchronous Optical NETwork (SONET) [3], the door is opened for emerging high bandwidth network applications such as video teleconferencing which were once considered impractical due to high bandwidth requirements. BISDN networks based on the ATM protocol will provide initial transmission rates of 155 Mb/s and 622 Mb/s, with gigabit speeds planned for the future.

The high bandwidth of ATM networks has important ramifications on multicasting. The simplistic N-Unicast method is undesirable for ATM multicast because it is very wasteful of network bandwidth and would greatly contribute to network congestion. The conference call method is also not desirable because the fixed location of the conference hub results in link distances that are often not optimal. Small multicast groups may be the only kind that can use these methods in an ATM network. Thus more efficient and scalable multicast routing algorithms are required for high speed ATM networks.

1.3 Objectives

The objective of this research is to evaluate the performance of multicast routing algorithms for an ATM network environment. More specifically:

1. What are the performance differences between the different multicast routing algorithms, and at what level of multicast load do these performance differences become significant?

2. How well do multicast routing algorithms scale with group size and network size?

3. What heuristics are most useful in developing efficient multicast routing? For example, does link sharing make any difference? How well do these heuristics work on different routing algorithms?

The evaluation of multicast routing algorithms is done using simulation. The routing models evaluated are the N-Unicast routing model, the Conference Call routing model, and the Core Based Tree routing model. Modified versions of the latter two models which attempt to use common network links when possible are also evaluated. This study of multicasting is restricted to the point to multipoint case with static group memberships.

1.4 Outline of Thesis

The layout of this document is as follows: Chapter 2 presents an overview of multicast transmission in various types of networks, while Chapter 3 outlines related work done in this field. Chapter 4 describes the experimental design of the research. Chapter 5 discusses the experimental results of the three basic routing models, and Chapter 6 discusses the results of the two modified models. Finally, Chapter 7 presents the conclusions from this work and discusses topics for future research.

CHAPTER 2

BACKGROUND

The purpose of this chapter is to discuss how multicasting is currently implemented in existing infrastructures such as Ethernet and the Internet Protocol (IP). In addition, ATM will be discussed in the context of multicasting and what multicasting support currently exists within ATM.

In order to discuss multicasting, some terminology must be defined. In this document, a *network* is defined as a collection of interconnected machines called *hosts*. For the purposes of this research, the hosts are assumed to be ATM switches, although they could be any type of network switching element whose behaviour matches the description given in Section 4.1. Each host has its own unique address identifier and may have any number (greater than zero) of links connecting it to other hosts in the network. In some multicasting schemes, a *group address* identifier may be employed to define the address of a multicast group so that a sender need only address a packet to the group address in order to dispatch the packet to its destination hosts.

2.1 Ethernet Multicast

The IEEE 802.3 standard, otherwise known as Ethernet, defines a broadcast-based technology for local area networks (LAN's) [26] which supports the ability to receive data link layer frames that are given a multicast destination address. The

Ethernet frame format contains a source address and destination address, each 6 bytes long. The presence of a 1 in the high order bit (bit 47 counting from zero) of the destination address denotes a multidestination address, whereas a 0 in this location specifies a single station address [8]. Furthermore, an Ethernet address can be global or local, depending on the status of the bit next to the high order bit (bit 46). The presence of a 0 in this location denotes a global address, while a 1 specifies an address confined to the local area network [8]. Global addresses are assigned by the IEEE to ensure that each station in the world has its own unique Ethernet address. Local addresses are administered locally and are of no significance outside the local network. A special broadcast address consists of all 1's and is delivered to all hosts in the local network and is relayed by all network bridges [26].

Ethernet interfaces are initially set up to receive frames destined to their own address, as well as the Ethernet broadcast address. Additionally, the hardware may be reconfigured via a software request to receive multicast frames with a particular multicast address identifier [14]. When a number of Ethernet stations are configured to receive frames of a given multicast address, an Ethernet multicast group is formed, which may be confined to the local network, or global, depending on the status of bit 46. Any station in the network may send frames to the multicast group, regardless of whether the station is configured to receive frames addressed to the group. It is up to the higher network layers to take care of address assignment and group management.

2.2 IP Multicast

The Internet Protocol (IP) is a connectionless network layer protocol that provides routing of datagrams across arbitrary internetworks [26]. IP defines four types of network layer addresses: Type A, B, and C are used for large, medium, and small networks, respectively, and Type D addresses are used for multicast. All IP addresses are 32 bits in length. The type classification is determined by the number of consecutive 1's in the high order bits of the IP address.

IP multicasting uses Class D addresses in the IP address range from 224.0.0.0 to 239.255.255.255, corresponding to 2^{28} possible multicast addresses [11]. IP multicast groups may be temporarily or permanently allocated. Permanent groups are allocated a "well known" address, such as 224.0.0.1 for the group of all IP hosts in the local network, and 224.0.0.2 for all routers in the local network [21]. When implemented on Ethernet, the low order 23 bits of the IP multicast address are placed in the low order 23 bits of the Ethernet multicast address (01-00-5E-00-00-00 hex).

The actual management of multicast groups in IP is handled by the Internet Group Management Protocol (IGMP) [11]. In Level 2 (full) support for IGMP, hosts are allowed to send and receive multicast packets. Multicast routers send out host membership queries within their local networks to determine which hosts belong to the known multicast groups. The hosts respond by sending out host membership reports, which contain the IP destination address of the multicast group being reported on. Host reports are randomly staggered and are configured such that only a single group member host in the local network submits a report to avoid bursts of simultaneous host reports [11]. The multicast router need only know if at least one host in the local network belongs to a particular multicast group. If at least one multicast group member exists in the network, IP packets from remote networks are forwarded into the local network for receipt by the group member(s).

2.3 Asynchronous Transfer Mode (ATM)

ATM is a connection-oriented protocol for "on demand" bandwidth allocation. Connections in ATM are point to point, or point to multipoint (multicast). ATM packets are referred to as cells. ATM cells are 53 bytes long and consist of a 5 byte information header and a 48 byte data payload. The cell header contains a connection identifier that maps the cell to its unique Virtual Channel on the physical link [25]. The ATM cell connection identifier consists of two fields: the Virtual Channel Identifier (VCI) and Virtual Path Identifier (VPI). The VCI denotes a unique virtual channel link between two switching points in an ATM network, while the

VPI denotes a unique virtual path link between two switching points in an ATM network. Virtual Paths (VP's) contain groups of Virtual Channels (VC's) that are switched together as a unit on common paths [20]. Virtual paths are used to reduce the connection setup and management overhead on the network switches, since control functions performed on a VP apply to all virtual connections on that path [25]. ATM switches are responsible for performing the switching of cells from VC to VC and VP to VP as a cell is routed from source to destination in an ATM network.

An ATM switch can be described as a black box with M inputs and M outputs that performs two main switching functions: cell multiplex switching and cell label switching [22]. Cell multiplex switching is the routing of a cell from an input port to an output port on the switch (see Figure 2.1), while cell label switching is the translation of the cell's connection identifier as read at the switch input port to its new value at the switch output port [22]. The cell routing information, which includes the input to output port mappings and label translations, is stored in the switch as tables. These tables are established at the time of connection setup or are predefined [9, 20].

The actual low-level support for multicasting in ATM switches depends on the architecture of the switch. Some designs are better suited to multicasting than others, and Turner [28] outlines some of the issues involved. For a general survey and performance comparison of ATM switches, the reader is referred to survey papers by Tobagi [27], Ahmadi [1], and Melen [22].

2.4 ATM Multicasting

Multicast extensions to the Internet Protocol (IP) exist via the Internet Group Management Protocol (IGMP) [11]. These multicast capabilities map on to the multicast capabilities of the underlying hardware such as Ethernet. However, there is no direct mapping of IP multicast or even IP broadcast functions directly onto ATM functionality [19]. Other methods are needed to implement an ATM multicast

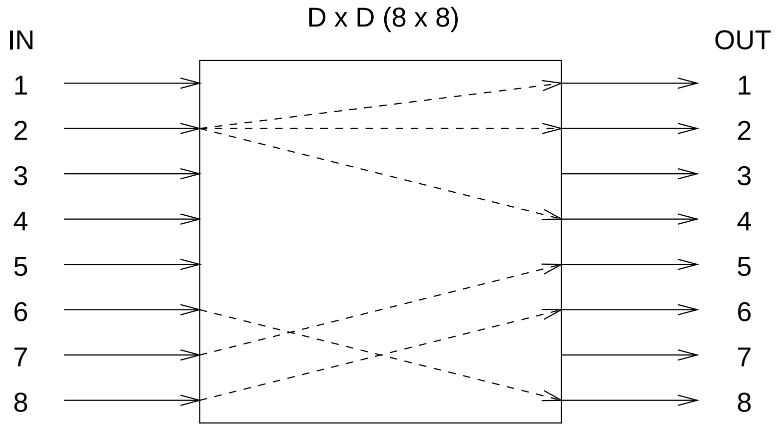

Figure 2.1: ATM switch, showing multicast routing from input port 2 to output ports 1, 2, and 4, and unicast routing on three other input ports.

group.

Point to multipoint Virtual Channel connections are supported in the Phase 1 Signalling specification of the ATM UNI 3.0 specification [15]. One ATM link is designated as the Root Link in a tree topology which sends information to all the other hosts in the multicast group. The leaf hosts are not allowed to directly communicate with each other, and Phase 1 Signalling does not allow the leaf nodes to send messages to the root node. Furthermore, multipoint to multipoint connections are not supported in Phase 1 [15]. A multipoint to multipoint connection is achieved by using multiple point to multipoint connections.

2.5 Summary

This chapter has discussed support for multicast transmission in Ethernet, IP, and ATM networks. While IP multicast is directly supported in Ethernet hardware as per the IEEE 802.3 specification, no direct multicast support exists as yet for ATM. Although single multicasting is supported in the ATM UNI 3.0 specification, its actual hardware implementation is left up to the switch vendor.

CHAPTER 3

MULTICAST ROUTING

The goal of multicasting is to route a network packet from a source to a group of recipients while using as little of the total network bandwidth as possible. The motivation for this is to minimize transmission costs, and to allow for maximum sharing of the communications infrastructure. For this reason, multicasting is essentially a routing problem, and in this chapter several multicast routing methods are studied.

3.1 Evaluation Criteria

Frank *et al.* [16] describe in their work some of the qualitative criteria used for rating the various multicast routing methods. For a group with a single multicast source, the criteria are:

- Bandwidth: the cost of setting up the multicast in terms of total hop-weighted link bandwidth used. In this thesis, this is interpreted as link cost.
- Delay: the time from when the first multicast packet is sent from the source until the last packet is delivered to all the recipients. For example, a multicast method that makes use of parallel packet deliveries will have a lower overall delay than a method that sends a packet to each recipient sequentially. Also, this criterion would need to assume that the multicast group has some kind of "typical" dispersion throughout the network under study.

- State: the total cost of storing the state information of the multicast group, such as group identifiers and group membership lists. The multicast group state may either be stored centrally, or be distributed among the network routers or switches.
- Computation: the computational cost of setting up a single multicast group.

Different multicast methods incur different costs for each of these criteria. Low costs are desired for all four rating criteria. The following sections describe the typical multicast methods used and their relative costs.

3.2 Point to Point Multicasting Methods

3.2.1 Simplistic Approaches to Multicasting

Two simplistic methods of multicasting are flooding and the N-Unicast method. In flooding, the multicast packet is initially sent from the source node to all immediately adjacent nodes, which then forward copies to all their adjacent nodes except the node from which the packet arrived. The multicast group members eventually receive the packet; the other nodes discard the packet after forwarding it to other nodes. This method is extremely wasteful of bandwidth, since it is more like broadcasting than multicasting, and because packets can be bounced around the network indefinitely unless packet lifetimes are bounded to a certain number of repetitions. On the positive side, flooding is very simple since there is no state information required, and no path computation is needed. However, due to its inefficiency, this method will not be investigated further.

In N-Unicast, the source node makes N copies of the outgoing packet and addresses and routes each copy separately to each recipient. This method is still quite wasteful of link bandwidth, since multiple copies of a packet may traverse much of the same path to get to the recipients. On the other hand, N-Unicast is reliable, simple to implement, and has low costs in terms of computation and maintaining state. The N-Unicast method is used in the distribution of mail to multiple recipients on the Internet via the Simple Mail Transfer Protocol (SMTP) [21].

3.2.2 Telephony Approaches to Multicasting

Telephone conference calling is a form of multicast transmission, where conferees participate in a multipoint to multipoint audio exchange. One common arrangement is called the "meet-me" teleconference where the conferees dial in to a special number assigned by the telephone company on the conference bridge at a prearranged time [24]. The conference bridge replicates incoming messages from each of the participants and sends the copies to all the other participants. The conference bridging may be provided by the exchange carrier end-office or tandem-office switching systems or by the interexchange carriers [4]. Since the conference bridge is a specialized piece of equipment, the conference hub must exist in a fixed location. The hub for a given conference call is then chosen based on the geographic location of the conference originator. The net effect of the conference call model of multicasting is a star topology with a single point of failure (the conference bridge). Due to the individual replication and routing of messages, the bandwidth usage of this approach is similar to that of the N-Unicast method. Furthermore, the fixed location of the hub may be considerably less than optimal for cases where a tightly clustered group of participants exists a long distance from the hub. However, this method of multicasting is simple to implement, has low costs in terms of state and computation, and has a low variance in delay due to the direct connection of each participant.

3.3 Graph Theory Approaches to Multicasting

Graph theory has been used to solve a wide variety of optimization problems ranging from project scheduling to determining maximum flows in pipeline systems. Graph theory is particularly useful in finding shortest paths and least cost paths in networks. Graph theory methods facilitate the construction of an optimal or near-optimal multicast delivery tree according to some cost metric such as least total number of links traversed. The multicast routing problem falls into the Steiner Tree and shortest path class of problems in networking. In this section the Steiner Tree and shortest path problems are studied as well as the IP-Multicast and Core-Based Tree solutions to the multicast routing problem.

3.3.1 Steiner Trees

The Steiner Tree problem in graphs can be summarized as finding the lowest cost subtree or subnetwork spanning some subset of points in a network graph [31]. In his survey paper on the Steiner Tree problem in networks, Winter [33] formally defines the problem as follows:

Given: An undirected network $G = (V,E,c)$ with n vertices, m edges, an edge cost function $c : E \rightarrow R$, and the subset $Z \subseteq V$ with p vertices.

Find: A subnetwork G_Z of G such that there exists a path between every pair of vertices in subset Z and the sum of the edge costs of G_Z is a minimum.

Figure 3.1 illustrates a weighted six node network graph and the Steiner tree for the node subset {1,2,4,5}. All links are of equal distance, and node 1 is the sender with nodes 4 and 5 being the recipients.

Unfortunately, the solution to the Steiner Tree problem has been shown to be NP-complete [17, 31]. To derive algorithms which can solve the Steiner Tree problem in polynomial time, heuristics are used in constructing the spanning tree.

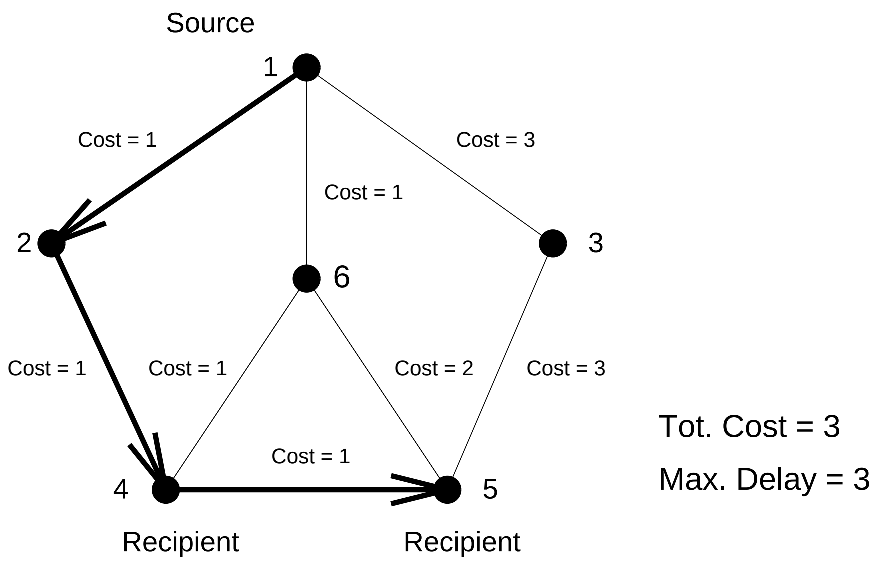

Figure 3.1: Steiner Tree minimum cost multicast route. All links have distance 1.

3.3.2 Shortest Path Methods

The shortest path problem in undirected network graphs corresponds to finding the minimum delay path from a source node to a group of recipient nodes. Shortest paths can be easily computed using a polynomial time method such as Dijkstra's algorithm. However, Wall [30] noted that the minimum delay path does not always coincide with the lowest cost path. Delay critical applications such as voice traffic over ATM may perform better with a shortest path algorithm than with a lowest cost based Steiner Tree solution. Figure 3.2 illustrates the same network and node subset {1,2,4,5} as Figure 3.1 routed according to a shortest path algorithm for each of the recipients.

3.3.3 Core (Center) Based Tree Methods

As described earlier, Wall [30] indicated that while minimum Steiner Trees are cost effective, they do not necessarily optimize routing delay. Conversely, a shortest path tree provides minimum delay, but does not guarantee lowest route cost. Wall proposed a compromise approach which uses a single tree rooted at some node central to the multicast subnetwork. Ballardie *et al.* [2] discuss the practical aspects

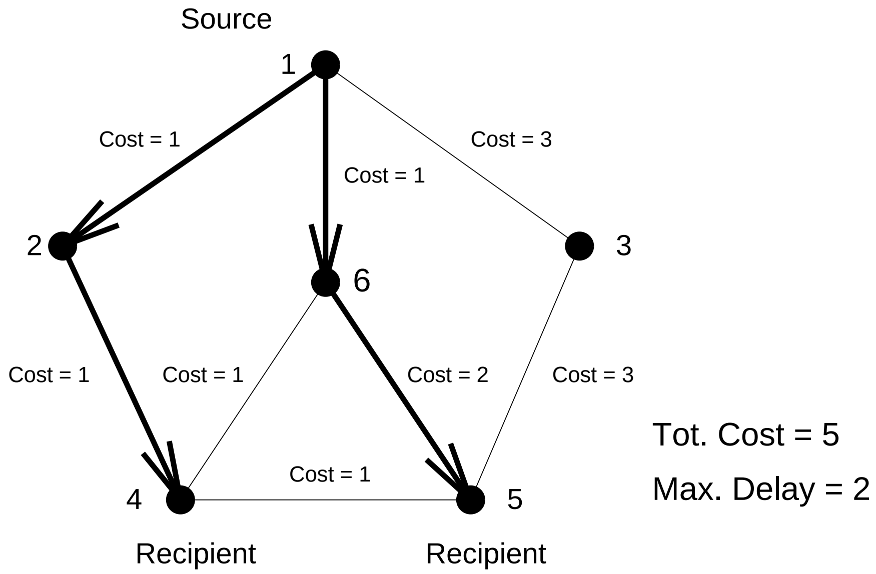

Figure 3.2: Minimum delay shortest path multicast route. All links have distance 1.

of Core Based Trees (CBT's) in their paper. An example of a CBT for the node subset {1,2,4,5} is shown in Figure 3.3. Unresolved issues with CBT's include determining optimal core placement, robustness (the core node and every span up to it is a single point of failure), and the possibility of using multiple cores in a multicast group.

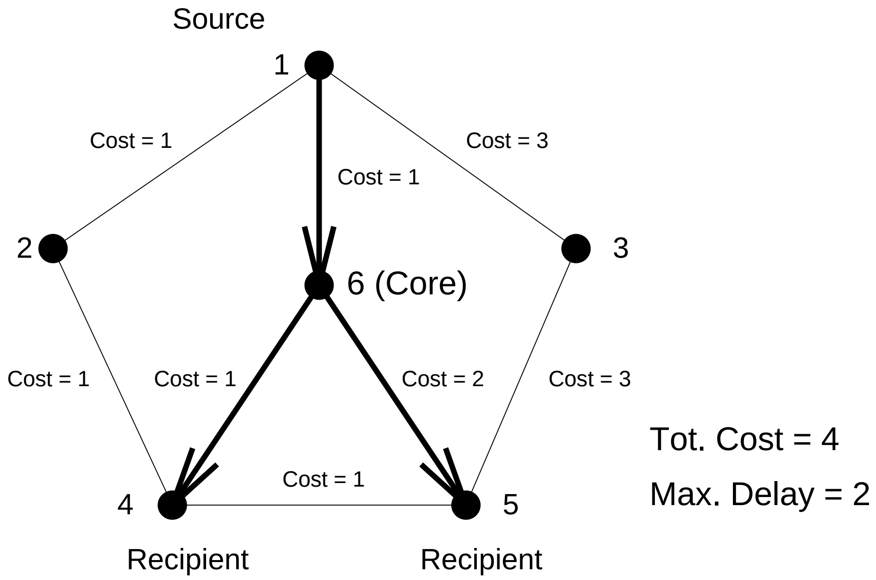

Figure 3.3: Simple Core Based Tree. All links have distance 1.

3.3.4 IP-Multicast Methods

The MBONE IP multicast audio and video transmissions mentioned in Chapter 1 utilize the Distance Vector Multicast Routing Protocol (DVMRP) developed by Deering [10, 12, 13, 29]. DVMRP's routing is based on generating shortest path routes from "leaf routers" (whose local subnetwork contains the multicast recipients) to the multicast source. Each multicast router in the network must determine its proper place in the multicast tree, and determine which of its multicast connections not in the shortest path tree are to be removed from the route. The Truncated Reverse Path Broadcasting algorithm is used to track which routers need to belong to the multicast group. Unnecessary routers are "truncated" from the multicast tree by routers located closer to the multicast source.

3.4 Summary

This chapter has given an overview of point to point multicast routing methods such as flooding, N-Unicast, and the Conference Call method. Also covered are more sophisticated tree-based routing methods such as DVMRP, Steiner Trees, shortest path trees, and Core Based Trees. While the latter group is more difficult to implement than the former, tree based methods promise more efficient link usage than point to point methods. It now remains to experimentally evaluate how much of an improvement the tree based multicast routing methods offer, and under what circumstances.

CHAPTER 4

EXPERIMENTAL DESIGN

This chapter describes the setup of the simulation experiments for this thesis. Also discussed are workload generation and experimental design issues. Section 4.1 discusses the network model used in the simulations, while Section 4.2 describes in detail the five multicast routing models used in the experiments. Section 4.3 outlines the characteristics of the simulated workloads. Section 4.4 describes the performance metrics for the simulation experiments, while Section 4.5 lists the experimental factors and levels. Section 4.6 discusses the experimental design, and Section 4.7 discusses the validation of the synthetic workloads and simulation models. Finally, Section 4.8 provides a summary of this chapter.

4.1 Network Model

The network topology used in this thesis was an regularly connected D by D rectangular mesh, for different values of D (D = {4,6,8,10}). An example mesh is illustrated in Figure 4.1. Each node in the mesh was assumed to be identical in capability, differing only in degree of connectivity with other nodes, depending on location within the mesh. Each node was capable of sending and receiving any number of connections from other nodes. The limiting factor was that the interconnecting links in the mesh were of equal capacity, expressed as a finite number of virtual call channels. Whenever a link was used by a call, the available capacity of that link was decremented by one channel for the duration of the call. If one or more members of a multicast call group could not be reached due to link blocking by other calls, the multicast call was considered to have failed or have been *blocked* (blocked calls are discarded immediately). This arrangement was designed to emphasize efficient link allocation in routing a multicast group.

The experiments were performed using simulation. This technique was chosen over trace analysis since a real multicast ATM network was not available at the time of the study. Analytic approaches were not used in this investigation due to the complexity of the system under test (particularly for large meshes).

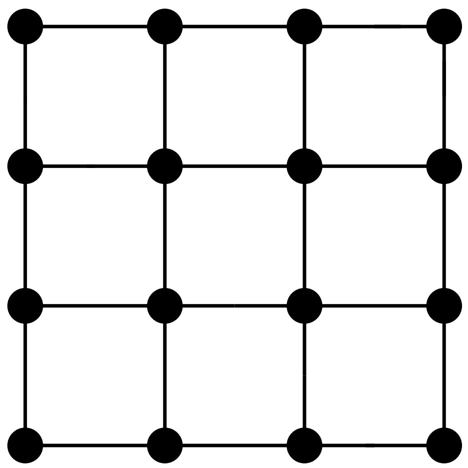

Figure 4.1: Example Network Topology (Size = 4 x 4)

4.2 Multicast Models

Five multicast call routing models were investigated: N-Unicast, Conference Call, Core Based Tree, and modified versions of the Conference Call and Core Based Tree models. Except where noted, all models used a randomized breadth-first search (BFS) to route from source to destination. The BFS method was chosen because it finds a shortest path from source to destination if a path exists. The use of randomization in the choice of routing direction balances load in the network and eliminates routing biases.

4.2.1 N-Unicast Model

The N-Unicast model was simulated by having the source node make N separate connections, one to each of the recipient nodes. An obvious limitation to the maximum number of recipients in this case was N $\leq 4m$ for interior nodes where m is the maximum channel capacity of the link. Corner nodes have a limit of N $\leq 2m$ and non-corner edge nodes are limited to N $\leq 3m$ in this model. BFS routing was used to determine the path from the source node to each of the recipient nodes.

4.2.2 Conference Call Model

The Conference Call model was implemented by having the conference hub(s) located in the interior nodes of the network. As in the case with telephone systems, the conference hubs' location in the network was fixed. Multicast calls were first routed via BFS from the sender to the conference hub, and were then replicated and separately routed via BFS from the conference hub to each recipient.

The hubs were distributed as evenly as possible in the networks, as shown in Figure 4.2. To minimize the effects of having a wide range of nodes being served by a single hub, networks of size D = 6 were served by 2 regional hubs, and networks of size D = {8,10} were served by 4 regional hubs. Depending on the dimension of the mesh, there were at least 16 nodes for every hub and at most 25 nodes per hub. Multicast calls originating from within a hub's region (for $D > 4$) could only be routed through that one hub. To promote routing fairness, the regions were made equal in size.

Table 4.1 shows the number of conference hubs and nodes per hub for mesh sizes from D = 4 to D = 10. Meshes smaller than 4 x 4 were not modeled with conference hubs since their simulation would be uninteresting. Mesh sizes larger than 10 x 10 were not considered due to simulation run time considerations.

The Conference Call model eliminates the problem of the low maximum number of recipients for corner and edge sources, but still suffered from the N $\leq 4m$ restriction for the maximum number of recipients since hubs always have 4 outgoing links, each of capacity m. It was assumed that network link channels were not shared by group recipients with similar routing paths.

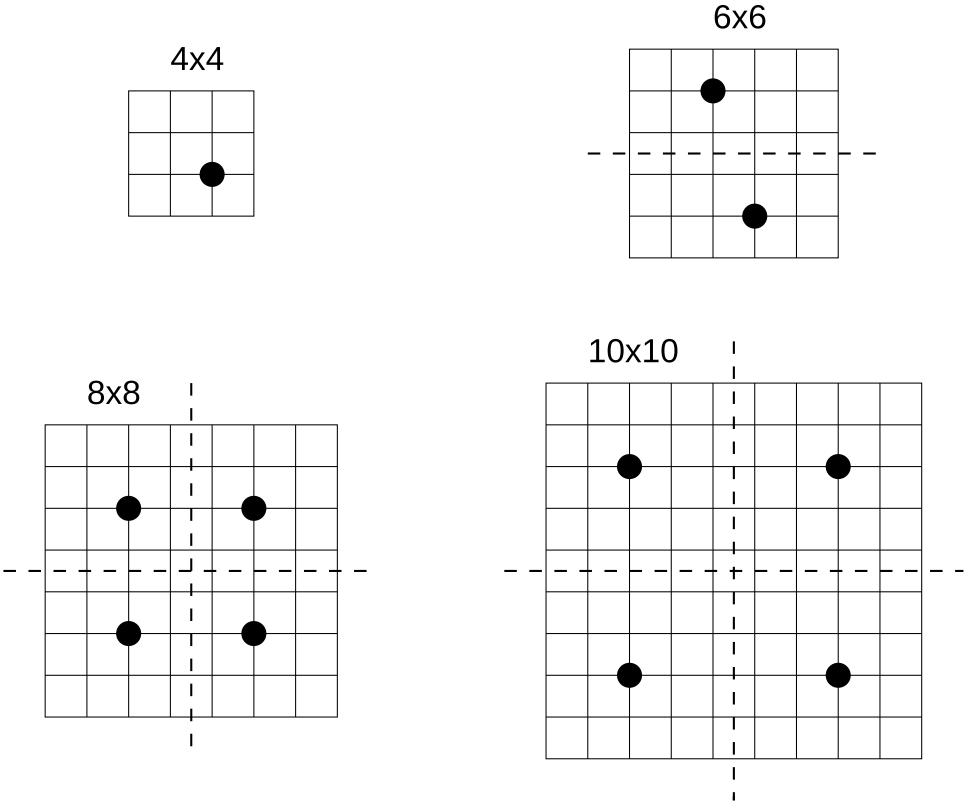

Figure 4.2: Conference Call Hub Placement for D = 4, 6, 8, 10

Table 4.1: Distribution of Conference Hubs

Mesh Size D	Nodes	Hubs	Nodes per Hub
4	16	1	16
6	36	2	18
8	64	4	16
10	100	4	25

4.2.3 Core Based Tree Model

The basic Core Based Tree model was an extension of the Conference Call model. Recall that in the Conference Call model, the location of the conference hub(s) were *static* (fixed). In the Core Based Tree model, the placement of the core (hub) was chosen *dynamically* for each multicast group based on where in the network the group members reside.

The dynamic core placement was implemented by placing the network nodes in a two-dimensional integer coordinate system and calculating the geometric center of the multicast group. The coordinates $< C_{\overline{x}}, C_{\overline{y}} >$ of the center of a multicast group with a single sender and $n-1$ recipients were calculated as follows,

$$C_{\overline{x}} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$C_{\overline{y}} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

where $< x_i, y_i >$ are the coordinates of the individual group members and the originating node. It was assumed that all network nodes were uniformly spaced on a two-dimensional x-y plane and all nodes were of equal unit weight. The center node location is rounded to the nearest integer x and y coordinates since the network topology did not support fractional coordinates. An example 8 x 8 network with coordinate system and core placement for a multicast group with 3 recipients is depicted in Figure 4.3.

The path from the source to the core was determined by the BFS routing algorithm as were the paths from the core to the recipients. It is possible, in some cases, for the core node to coincide exactly with one of the recipients, or even the source node. These cases do not present difficulties for the algorithm.

In a manner similar to the Conference Call model, the basic Core Based Tree model assumed that network links from the core to the recipients were not shared by the recipients. The reason for this assumption was to separate out the effects of further modifications outlined in the next subsection, and to emphasize the effect of using a dynamically placed core versus a statically placed core.

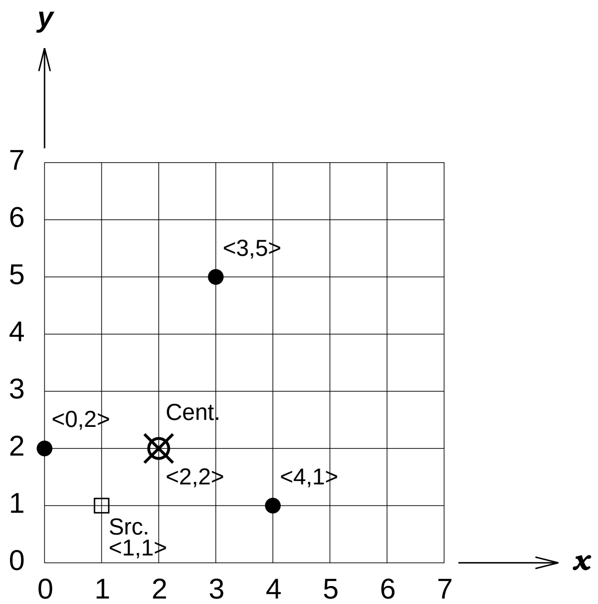

Figure 4.3: Coordinate System and Core Centroid for D = 8 Mesh and N = 3 Group Members

4.2.4 Core Based Tree Model with Link Sharing

A modified Core Based Tree model was implemented in the same manner as the basic Core Based Tree model. The only change was the use of link sharing in routing from the core node to the recipients to conserve link bandwidth. With link sharing, multicast calls that use common links in routing a multicast group use only one unit of channel capacity for all recipients on those common links, regardless of how many recipients use that link. This was implemented by taking advantage of common paths when possible during BFS routing from the core node to the recipients. That is, the calls will route as if there was no link sharing, but the link sharing algorithm checks for duplicate channels on the network links. All duplicate channels for a call on any link are merged into one channel and redundant the channels are made available for other calls.

4.2.5 Conference Call Model with Link Sharing

Preliminary results showed that using link sharing with the Core Based Tree model resulted in significant reductions in multicast call failures when compared with the non-shared link Core Based Tree model. It was decided to investigate the performance impact of using link sharing on the Conference Call Model as well to see if similar reductions in multicast call failure could be realized.

4.3 Workload Generation

A workload generator was used to construct the input load to the network simulator. The following assumptions were made about the workload:

- Calls arrived according to a Poisson process
- Call holding times were exponentially distributed
- Multicast group size was geometrically distributed
- Recipients were picked uniformly at random

A geometric distribution for the multicast group size was chosen because it was the discrete equivalent of the exponential distribution. The memoryless property of the geometric distribution was desirable in modelling multicast group size. The formula [18] used to obtain the geometric distribution $G(p)$ from the uniform random number u from the $U(0,1)$ distribution is:

$$G(p) = \left\lceil \frac{ln(u)}{ln(1-p)} \right\rceil$$

where $\lceil \cdot \rceil$ denotes the ceiling function and p is the probability of success. For the geometric distribution, the mean multicast group size g is equal to $\frac{1}{p}$.

The distributions were created with a pseudo-random number generator. Only a specified fraction of the total calls were multicast, and these were selected at random. Multicast calls with fewer than two recipients were not permitted, and multiple occurrences of the same recipient in a group were also not permitted. Some assumptions about the network models were also made to govern generator behaviour. A recipient was able to receive multiple calls simultaneously, and an originator could receive calls while sending. Calls of zero distance (originator to originator) were not allowed.

The workload parameters were:

- *Mesh size (D).* This was required since each mesh size had different numbers of potential recipients.

- *Percentage of multicast traffic.* The average percentage of multicast calls in the generated call workload.

- *Mean multicast group size (g).* The average number of recipients in a multicast group.

- *Mean call holding time (duration).* The average duration of a unicast or multicast call.

- *Mean call interarrival time.* This parameter is expressed relative to the mean call holding time, which is always fixed at 500 seconds in the experiments in this thesis. The call interarrival parameter thus affects the average number of active calls in the network.

The randomly generated workload was represented as lines of data in a file as shown in Figure 4.4. The purpose of using workload files was to provide exactly the same workload for each routing algorithm. Each line of data corresponded to a unicast or multicast call. The information contained in each call consisted of the call start time in seconds, the duration of the call in seconds, the number of the node which originated the call, the number of group recipients, and the list of recipient node numbers. Each of the five (logical) columns of data was generated using a separate independent random number stream. While the design of the workload generator did not allow more than one call to start at the same instant in time, calls were able to overlap each other in an unrestricted manner. Since the mesh size placed an upper bound on the number of group recipients, it was decided to generate separate workloads for each size of mesh rather than attempt to use one workload for all mesh sizes. All workload files had 100,000 calls. This value was chosen as a compromise value between minimizing start-up effects and executing the simulations with reasonable computation time.

```
16708.941   122.551   31   1   20
17591.751   174.804   24   1   15
17638.253   389.275   17   8   24 15 3 25 5 33 30 35
17714.681   893.331   22   1   4
17835.537   416.170   11   1   26
18561.058   744.052    2   5   20 35 8 9 4
```

Start Time (sec.) | Duration (sec.) | Origin | Group Size | List of Recipients

Figure 4.4: Example of Workload File, D = 6

4.4 Performance Metrics

The performance metrics for the simulation evaluation of multicast routing algorithms were:

- *Percentage of failed calls (PFC).* The percentage of failed (blocked) calls was used as one indicator of how efficient a call routing model was. Both the overall (cumulative) value of this metric for all (unicast and multicast) calls and the individual values broken down according to multicast group size were recorded. The cumulative value was useful for comparing routing models, while the detailed values identified group size dependency in the call blocking behaviour.

- *Average links used per call.* The average number of links used per successful call in unicast and multicast calls was another indicator of how efficient a call routing model was. The data from this metric was also broken down according to multicast group size.

- *Average number of active calls per link in the network.* This metric was used to measure link utilization in the network (*i.e.,* how busy each link was during the simulation). These values are useful as an additional indicator of how effective a call routing model was. They also serve to identify hot spots in the network.

4.5 Factors and Levels

The primary factors and corresponding levels used in the experiment were as follows:

Multicast routing model:	N-Unicast (N-UNI)
	Conference Call (CC)
	Core Based Tree (CBT)
	CC with Link Sharing (CC-LS)
	CBT with Link Sharing (CBT-LS)
Mesh size:	4x4, 6x6, 8x8, 10x10
Percent multicast traffic:	10%, 20%
Mean multicast group size (g):	4, 8

In addition, three non-factor parameters were used in the experiment:

- Link capacity: 5 channels per link. Preliminary investigations revealed this value to be a good value for creating call blocking in the non-shared link routing models. This value was chosen to stress the routing models yet provide some capacity to handle multiple calls on a single link.
- Mean call duration: 500 seconds (chosen arbitrarily).
- Mean call interarrival time: This value was chosen so that the number of active calls in the network was equal to half the number of network nodes (assuming that no call blocking occurred). Table 4.2 contains the values of this parameter with respect to mesh size. Note that these parameter settings

result in significantly different workloads for each mesh size. Thus it does not make sense to compare call blocking performance for a given routing algorithm across different mesh sizes, but it does make sense to compare the call blocking probability of different routing algorithms for the same mesh size.

Table 4.2: Mean Call Interarrival Time as a Function of Mesh Size

Mesh Size (D):	4	6	8	10
Number of Calls in Network:	8	18	32	50
Mean Interarrival Time (sec.):	62.5	27.8	15.6	10.0

Each test run was simulated with 100,000 calls. This value was chosen arbitrarily to achieve a balance between statistically significant results and reasonable computation time.

The simulation models were started in an initial state of zero active calls in the network, *i.e.,* there was no preloading of the network with initial calls. Start-up effects in the simulations were considered to be negligible for two reasons. First, the set of workloads used by each algorithm were identical, which resulted in identical call behaviour being offered to each algorithm. Second, preliminary tests showed that the simulation models typically reached steady state within the first 1000 calls of the simulation. This start-up phase corresponded to only 1% of the total workload, and was thus considered to have minimal effect on the results.

4.6 Experimental Design

A full factorial experimental design with no replications was used. A total of 80 test runs were performed, corresponding to five levels of routing models, four levels of mesh size, two levels of multicast traffic fraction, and two levels of mean multicast group size. Sixteen workloads were utilized, four for each size of mesh. One pair of each workload quartet corresponded to the 10% multicast call input load factor and the other pair corresponded to the 20% multicast call input load factor. Within each of these pairs, one workload had a mean multicast group size of 4 recipients and the other workload had a mean multicast group size of 8 recipients. Each workload was run on all five routing models.

4.7 Validation

4.7.1 Workload Validation

The sixteen workloads generated were checked for accuracy with respect to average number of calls input to the network. Table 4.3 summarizes the time weighted average number of calls offered to the network by the workloads, as well as the observed average number of active (*i.e.,* accepted) calls during the simulation for workloads with 10% multicast traffic and an average multicast group size g of 4. The time weighted averages were determined by summing the (simulated) durations of all calls during the simulation and dividing this by the (simulated) duration of the entire 100,000 call test run. The mean number of active (*i.e.,* accepted) calls is always less than or equal to the mean number of offered calls because of the chance of call blocking. Due to the modular design of the workload generator, there was no difference in the arrival times, call duration, and source node of calls between workloads that differed only in mean multicast group size. Thus the average number of calls offered to the network was the same between workloads with a mean multicast group size of 4 and their counterparts with a mean group size of 8. The average number of calls offered to the network was 7.98 for the 4 x 4 mesh, 17.93 for the 6 x 6 mesh, 31.80 for the 8 x 8 mesh and 49.56 for a mesh of size 10. All values are within 1% of the desired workload characteristics shown in Table 4.2.

A profile of the number of calls offered to the network appears in Figure 4.5 for the 4 x 4 mesh with a 10% multicast load and a mean multicast group size of 4. Similar traffic offered plots, all for 10% multicast load, appear in Figures 4.6, 4.7, and 4.8 for mesh sizes of 6, 8, and 10, respectively. All plots were taken from the middle of the workload and only represent approximately a 1000 call segment of each workload. Table 4.4 lists the mean and variance in the number of calls offered to the network corresponding to the entire 100,000 call workloads in Figures 4.5 through 4.8. For these four cases, the average and variance were not time weighted *i.e.,* they were calculated using the raw number of calls in the mesh. For all four mesh sizes, the mean number of calls in the network was roughly the same as the variance in the number of calls in the network, which was the expected behaviour for the prepared workloads (Poisson arrivals, exponentially distributed holding times). The traffic offered plots for the 20% multicast workloads appear similar to their 10% multicast counterparts and are not shown. The workload profiles show that the number of calls offered to the network at any given point in time is a random variable with mean close to that given in Table 4.2. Thus the workload generator appears to be working as expected.

The histogram of multicast group size produced by the workload generator was also evaluated. For example, a profile of the number of calls generated with two or more recipients for the 4 x 4 mesh workload with 10% multicast calls and a mean multicast group size of 4 appears in Figure 4.9. In this workload, the number of unicast calls generated was 89,758 and the total number of calls generated was 100,000. The profile of the group size for the multicast calls matches well with the geometric distribution (a modelling assumption). The other workloads displayed similar profiles and are not shown. A similar profile for the 4 x 4 mesh workload with a mean multicast group size of 8 appears in Figure 4.10.

4.7.2 Simulation Model Validation

Validation of the simulators was conducted to verify correctness of simulation results. The first test performed was a test run on a 2 x 2 mesh. Since there are only four links in the mesh, and all nodes were connected by two links, all non-shared link algorithms were expected to have identical call blocking probabilities, and all shared link algorithms were expected to share a different call blocking probability. The validation test produced the expected results.

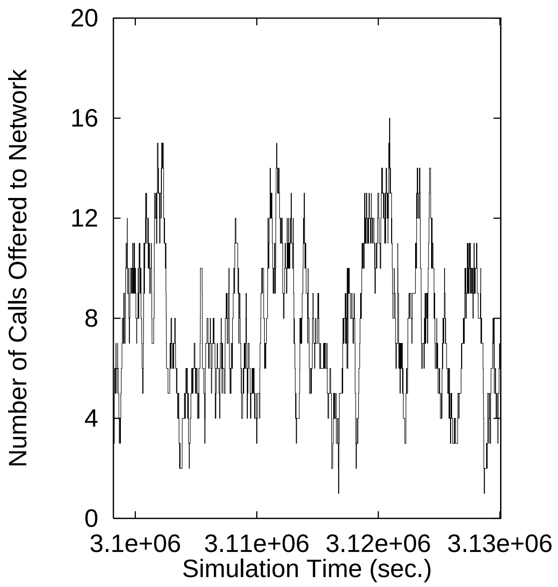

Figure 4.5: Workload sample for 4 x 4 mesh, 10% multicast load, g = 4.

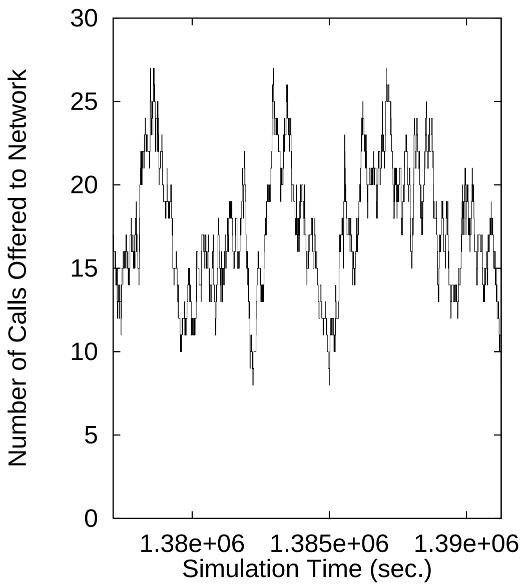

Figure 4.6: Workload sample for 6 x 6 mesh, 10% multicast load, g = 4.

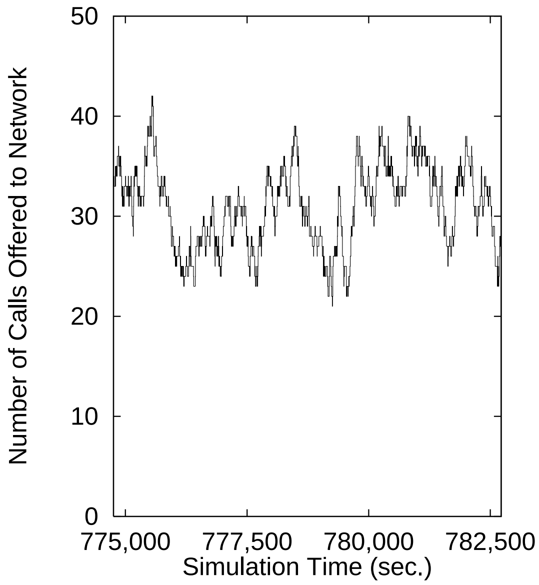

Figure 4.7: Workload sample for 8 x 8 mesh, 10% multicast load, g = 4.

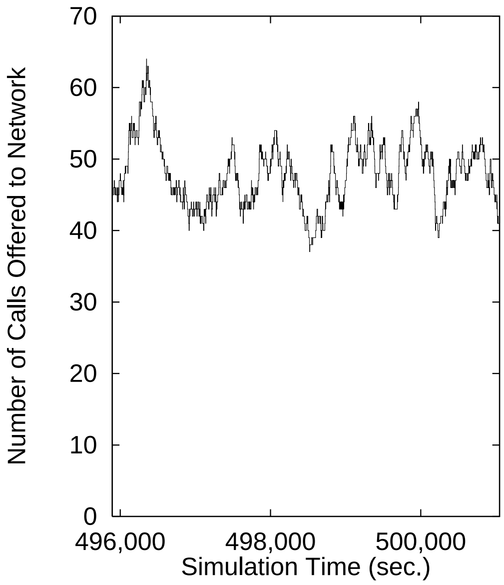

Figure 4.8: Workload sample for 10 x 10 mesh, 10% multicast, g = 4.

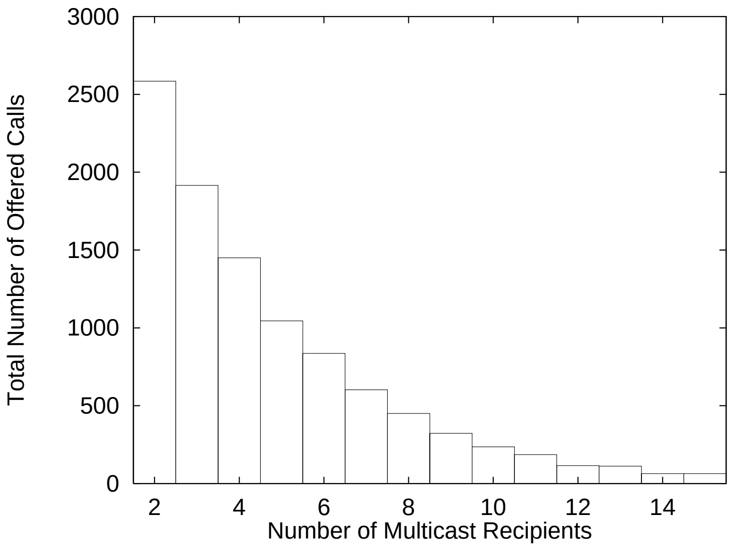

Figure 4.9: Multicast Call Distribution Profile for D = 4 Workload with 10% Multicast Traffic, g = 4.

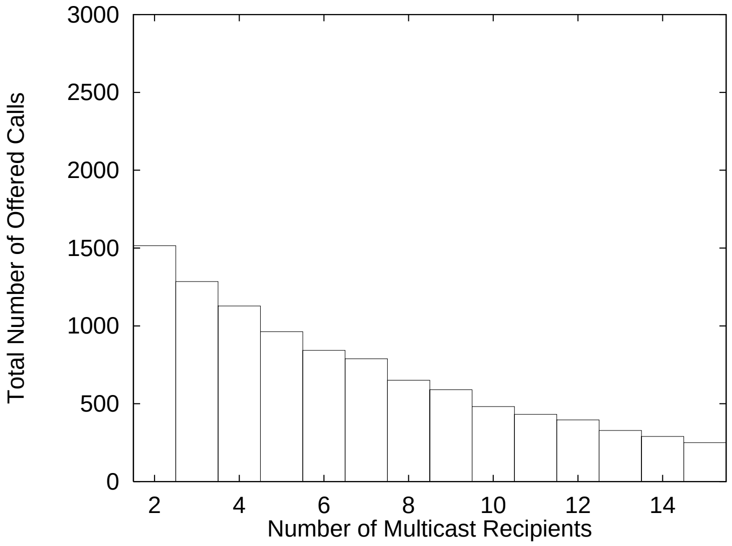

Figure 4.10: Multicast Call Distribution Profile for D = 4 Workload with 10% Multicast Traffic, g = 8.

Table 4.3: Average Input and Active Calls to Mesh for 10% Multicast Traffic, g = 4.

Mesh Size	Routing Model	Avg. Input Calls to Net	Avg. Active Calls
4	N-Unicast	7.98	7.90
	Conf. Call	7.98	7.90
	CBT	7.98	7.94
	Conf. Call w/ LS	7.98	7.98
	CBT w/ LS	7.98	7.98
6	N-Unicast	17.93	17.68
	Conf. Call	17.93	17.61
	CBT	17.93	17.67
	Conf. Call w/ LS	17.93	17.92
	CBT w/ LS	17.93	17.93
8	N-Unicast	31.80	31.09
	Conf. Call	31.80	30.95
	CBT	31.80	31.01
	Conf. Call w/ LS	31.80	31.65
	CBT w/ LS	31.80	31.67
10	N-Unicast	49.56	47.45
	Conf. Call	49.56	47.21
	CBT	49.56	47.45
	Conf. Call w/ LS	49.56	48.13
	CBT w/ LS	49.56	48.37

Table 4.4: Non-time weighted mean and variance in offered number of calls to network, 10% multicast load, g = 4.

Mesh Size (D):	4	6	8	10
Mean:	8.48	18.44	32.40	50.36
Variance:	8.01	17.46	30.30	47.41

Another test used to validate the unicast and multicast routing simulator was to perform test runs on all models using a workload with no multicast calls. All five routing models returned the same results, indicating that the unicast portion of the simulators were working correctly, and that the multicast component of the simulators were not evaluating unicast calls as multicast calls.

The third validation test used extreme values for channel capacity (*i.e.,* link bandwidth) to test and debug the call failure detection algorithms of the routing models. With all network link bandwidths set to a very high value, such as 100, there should be no call blocking. Tests were also conducted with all network link bandwidths set to zero, in which case every call would be blocked. All the routing models used passed these tests before the final simulation runs were conducted.

The link bandwidth tests also provided insight as to the proper amount of link bandwidth to use for the thesis experiments. Preliminary tests indicated that there was a considerable range of call blocking and link usage performance among the five different routing models. It was found that low link bandwidths such as 2-3 channels per link yielded a good range of results for the shared link models, but caused the non-shared link models to have high call blocking probabilities, even with small multicast group sizes. Raising the link bandwidth to 10 channels per link improved the non-shared link call blocking performance, but this caused the shared link call blocking probability for small mesh sizes to approach zero. Comparison between shared link routing models became difficult as a result. A compromise bandwidth of 5 channels per link was found to provide a reasonable level of call blocking in both shared link and non-shared link models, facilitating comparison.

The calls per link metric (used in hot spot analysis) proved to be very useful in debugging, testing, and validating the simulator. In fact, it helped detect a routing bias problem in an early version of the simulator. Prior to detection, the routing algorithms always tried downward links first when doing the breadth first search (BFS) between network nodes. A randomized BFS, which made each outgoing link equally likely to be chosen, eliminated the routing bias. The resulting symmetries in the hot spot analyses (presented in Chapter 5 and Chapter 6) provide confirmation that the randomized BFS algorithm is working as intended.

4.8 Summary

This chapter has served to provide background on the thesis experiments. The design of the network model was outlined, as were the five multicast routing algorithms: N-Unicast, Conference Call, Core Based Tree (CBT), Conference Call model with Link Sharing, and CBT with Link Sharing. The simulation workload was described, along with the performance metrics, factors and levels, and the experimental design. Validation of the workload and simulation models was also discussed. The next two chapters present and discuss the simulation results.

CHAPTER 5

EXPERIMENTAL RESULTS: NO LINK SHARING

This chapter presents the simulation results for the non-shared link multicast routing models. The results for these models are characterized by call blocking and link usage performance curves that readily allow comparisons between these models. In contrast, the shared link routing models had distinctly different call blocking and link usage characteristics. The simulation results for the latter models are discussed in Chapter 6.

The layout of this chapter is as follows. Section 5.1 presents the results for the N-Unicast routing model, while Section 5.2 presents the results for the Conference Call model, and Section 5.3 discusses the Core Based Tree model. Section 5.4 focuses on the comparison of routing models. Section 5.5 evaluates the effect of multicast group size on all three multicast routing models, while Section 5.6 compares the three routing models from the standpoint of network hot spot analysis. Finally, Section 5.7 summarizes the findings of this chapter.

5.1 N-Unicast Model

In the N-Unicast routing model, the source of a multicast call makes a separate unicast transmission to each of the N recipients of the call. This section presents the simulation results for the N-Unicast algorithm when the mean multicast group size is 4 recipients (g = 4).

For the 10% multicast traffic workload, the N-Unicast routing model resulted in an average call blocking probability of 1.0% on a 4 x 4 mesh. The average call blocking probabilities for the other mesh sizes (for different workloads, of course) were 1.5% for a 6 x 6 mesh, 2.6% for an 8 x 8 mesh, and 4.8% for a 10 x 10 mesh.

Figure 5.1 illustrates a breakdown of the call blocking probability as a function of the number of recipients for each call generated. Large multicast groups clearly had a much higher call blocking probability than small groups. For example, unicast calls on a 4 x 4 mesh had a call blocking probability of 0.3%, while multicast calls with groups of size 2, 5, and 10 had call blocking probabilities of 0.7%, 3.4%, and 36.9%, respectively. The bias against large multicast groups makes sense intuitively: the larger the multicast group, the less likely it is that all group members can be reached successfully when routing the call. If all recipients cannot be successfully reached, the call is blocked (and cleared).

For a given multicast group size, the probability of call blocking usually increased with mesh size. For example, for groups with 4 recipients, the D = 4 and D = 6 meshes had a 2.3% and 3.2% call blocking probability, respectively, while the D = 8 and D = 10 meshes blocked 9.6% and 16.1% of the incoming calls, respectively. The reason for this behaviour is that the higher average number of calls in the larger meshes overloaded the network (since the offered number of calls to the network was scaled based on the number of nodes in the mesh). This effect is referred to as autoscaling in this thesis. There is one anomaly, however. For groups of 7 to 13 and 15 recipients the D = 6 call failure probabilities were less than or equal to the call failure probabilities for D = 4.

The latter anomaly was investigated further using confidence interval analysis. The method of *batch means* [18] was used to determine the mean and standard deviation of the call blocking probability for each possible recipient group size. The batch size was chosen as 10,000 calls to provide a reasonable tradeoff between the number of batches (*i.e.,* 100,000 calls ÷ 10,000 calls/batch = 10 batches) and the number of blocked (multicast) calls per batch (*i.e.,* 10,000 calls $\times \approx$ 1% call blocking probability = 100 blocked calls per batch, of which approximately 10 are multicast calls). The call blocking probabilities observed in each of the 10 batches were then used to determine the 95% confidence intervals for the results.

The results of the batch means analysis are shown in Figure 5.5 for the D = 4 and D = 6 cases from Figure 5.1. As can be seen, the confidence intervals are tight for small group sizes (where the number of generated calls is large) and are much looser for large group sizes (where the number of generated calls is small). With the exception of groups of size 9 and 10, the confidence intervals for D = 4 and D = 6 overlap for group sizes greater than 6. Except for groups of size 9 and 10, the anomaly reported above for Figure 5.1 is statistically inconclusive and *may* be an artifact of the workload file used. However, the behaviour for groups of size 9 and 10 is likely the result of edge and corner effects in the 4 x 4 mesh. The lower number of routes for edge (3 paths) and corner (2 paths) nodes compared to interior nodes (4 paths) as well as the larger percentage of these nodes in the D = 4 network model (*i.e.,* 25% corner nodes vs. 11% for D = 6, and 50% edge nodes vs. 44% for D = 6) serve to increase the call blocking probability for large group sizes.

The N-Unicast routing model was found to require an average of 3.5 links per call for a 4 x 4 mesh with a 10% multicast traffic workload. Similarly, the 6 x 6 mesh needed 5.4 links per call on average. The 8 x 8 mesh used an average of 7.2 links per call, and the 10 x 10 mesh averaged 9.1 network links per call. The plots of link usage with respect to recipient group size are shown in Figure 5.2. The link usage grows nearly linearly for all mesh sizes. This linearity was expected because in a non-shared link routing model such as N-Unicast, the number of network links used increases in direct proportion to the number of group recipients. The irregularities in the link usage plots for large group sizes can be attributed to small samples of data for these group sizes.

Figure 5.2 also shows that larger meshes used more network links for a given multicast group size than smaller meshes. This was reasonable given that larger meshes have greater average distances from source to recipient than smaller networks. Section 5.4 will cover link usage trends for this and the other non-shared link routing models in greater detail.

Increasing the fraction of multicast calls increased the percentage of failed calls. For the 20% multicast traffic workload, the N-Unicast routing model attained an average call blocking probability of 2.9% on a 4 x 4 mesh, 4.6% for a 6 x 6 mesh, 7.3% for an 8 x 8 mesh, and 11.4% for a 10 x 10 mesh. This increase in call blocking probability was expected since multicast calls use more links per call than unicast calls. The graphs of call blocking probability as a function of group size in Figure 5.3 were similar to those in Figure 5.1, except that most of the corresponding call blocking probabilities were higher for the 20% multicast load than for the 10% multicast load. The anomaly where the D = 6 call failure probabilities were less than the D = 4 probabilities did not appear in this case. Using the same example of groups with 4 recipients, the D = 4 and D = 6 meshes had a 6.3% and 8.3% call blocking probability, respectively, while the D = 8 and D = 10 meshes had a 15.6% and 26.6% call blocking probability, respectively.

The percentage of multicast traffic at first appeared to have very little effect on the links used per successful call, since Figures 5.2 and 5.4 were very similar in appearance. However, the average number of links per call was greater for the case of 20% multicast traffic than for the 10% multicast traffic workloads. With 20% multicast traffic offered to a 4 x 4 mesh, 4.4 links were used per call on average while 6.5 links were used to route a typical call on a 6 x 6 mesh. For a mesh of size 8, 8.7 network links per call were needed, while a 10 x 10 network used an average of 10.8 links per call. This behaviour was not surprising since it was expected that more links per call would be required for the 20% multicast traffic workload over the 10% multicast traffic workload.

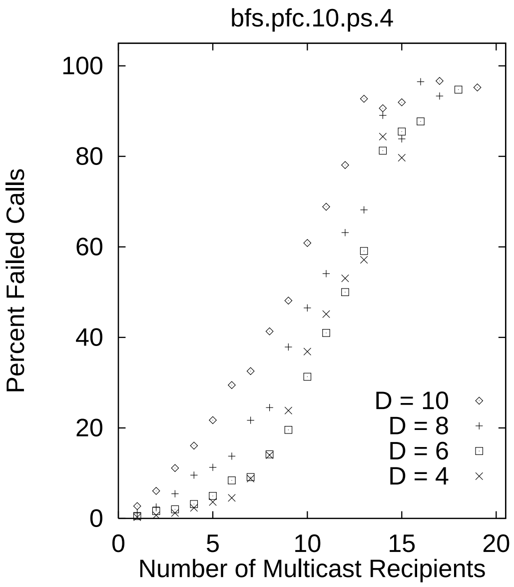

Figure 5.1: Percent failed calls, N-UNI model, 10% MC traffic, g = 4.

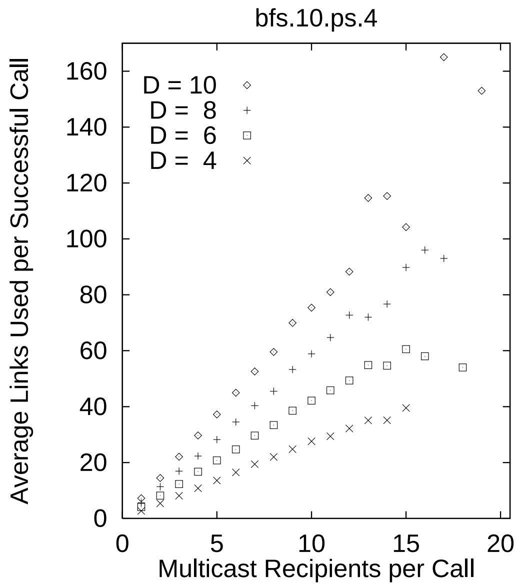

Figure 5.2: Avg. links per call, N-UNI model, 10% MC traffic, g = 4.

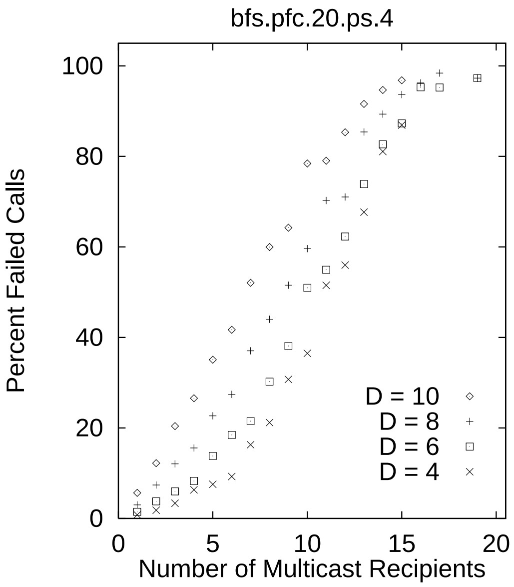

Figure 5.3: Percent failed calls, N-UNI model, 20% MC traffic, g = 4.

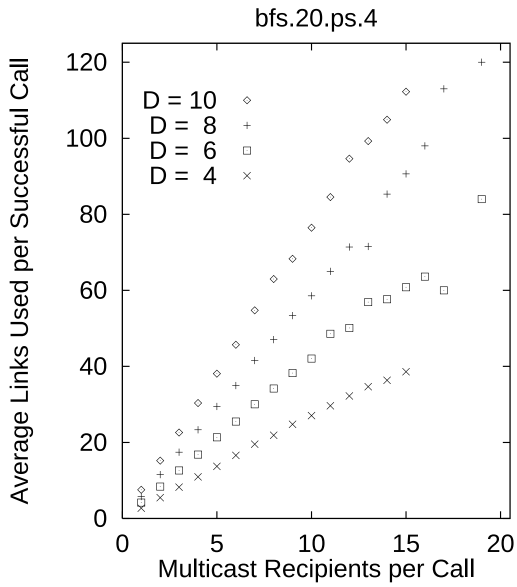

Figure 5.4: Avg. links per call, N-UNI model, 20% MC traffic, g = 4.

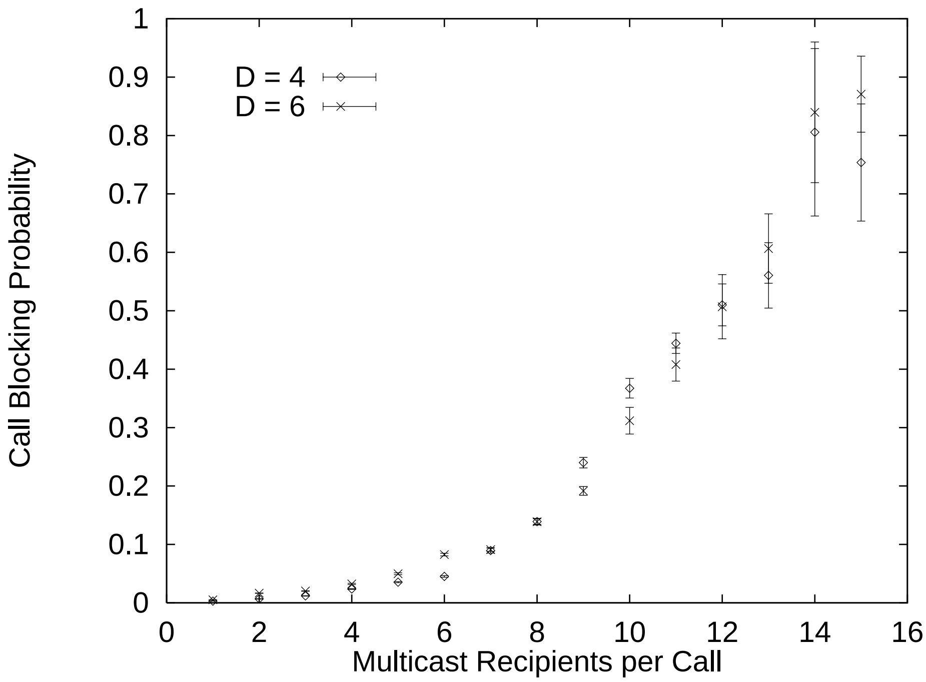

Figure 5.5: A 95% confidence level plot of call blocking behaviour for the N-Unicast routing model with 10% multicast traffic, g = 4.

5.2 Conference Call Model

The Conference Call model uses statically located hubs to route a multicast call. As described in Section 4.2.2, the number of hubs and the location of the hubs are determined by mesh size. This model was expected to have lower call blocking probabilities for multicast calls in small meshes since the point to multipoint connections originated from fully connected interior mesh nodes. The simulation results presented in this section are for the Conference Call model with a mean multicast group size of 4 recipients.

For the 10% multicast traffic workload, the Conference Call model attained an average call blocking probability of 1.1%, on a 4 x 4 mesh. The average call blocking probabilities for the other mesh sizes were 2.0% for a 6 x 6 mesh, 3.1% for an 8 x 8 mesh, and 5.4% for a 10 x 10 mesh. As with the N-Unicast model, call blocking probability increased with mesh size.

A breakdown of call blocking probability as a function of group size is shown in Figure 5.6. These values were slightly higher than the values for the N-Unicast model. The call blocking probability again shows a strong bias against large groups. For a given group size, call blocking probability increased with mesh size as was typical for the N-Unicast model. The autoscaling effect for call blocking in the Conference Call model was universal, except for some of the largest group sizes where sample sizes were very small (and the variances in the results were large). In fact, the autoscaling effect for call blocking probability was universal for all shared link and non-shared link routing models except for the anomaly noted for the N-Unicast model.

The call blocking values for the Conference Call model were slightly higher than the N-Unicast model on mesh sizes of 8 and 10 for recipient group sizes of 2 through 15. For a mesh of size 6, the Conference Call model had a higher call blocking probability for group sizes of up to 10 recipients, and a lower call blocking probability beyond this value. The size D = 4 case displayed similar behaviour, but the cross-over point was at groups with 9 recipients. The lower blocking probabilities for small meshes with large recipient groups was expected since edge and corner nodes were not a limiting factor as they were in the N-Unicast model. One possible explanation for the higher overall call blocking probability for the large mesh sizes was that the links surrounding the fixed conference hubs likely became bottlenecks for routing multicast calls.

Three trends present in the N-Unicast model regarding link usage were observed in the Conference Call model. First, the average link usage per call increases with group size (see Figure 5.7). Second, the relationship of average link usage per call to multicast group size is linear. Finally, for a given multicast group size, link usage increases with mesh size. For example, with the Conference Call model and a 10% multicast traffic workload, the 4 x 4 mesh used an average of 3.5 links per call, and the 6 x 6 mesh used 5.3 links per call. The 8 x 8 mesh required 6.9 links per call, and the 10 x 10 mesh needed 8.9 links per call on average. These values were equal to or slightly lower than their counterparts in the N-Unicast model with 10% multicast traffic, with the difference becoming larger with increasing mesh size.

With 10% multicast traffic, the Conference Call model appeared to reduce link usage slightly when compared with the N-Unicast model. For a 4 x 4 mesh, both the N-Unicast and Conference Call models required 3.5 links per call on average. The savings in link usage became more apparent in the large meshes, since the Conference Call model averaged 8.9 links per call for a 10 x 10 mesh, compared with 9.1 links per call with the N-Unicast model. A small savings in link usage over the N-Unicast model was expected for the Conference Call model, since the link intensive one-to-many multicast routing is deferred to a conference hub. The conference hub is usually closer to the recipients than the source node, particularly in large meshes with regional hubs. Section 5.4 presents a more detailed discussion and comparison of the link usage measurements.

For the 20% multicast traffic workload, the Conference Call model attained an average call blocking probability of 4.0% on a 4 x 4 mesh, 6.2% for a 6 x 6 mesh, 8.2% for an 8 x 8 mesh, and 11.8% for a 10 x 10 mesh. As expected, these values were larger than for the 10% multicast load. A breakdown of call blocking probability as a function of group size is shown in Figure 5.8. As was the case with the 10% multicast workloads, call blocking probability increased with group size and mesh size.

The Conference Call model with 20% multicast traffic nearly always had a higher call blocking probability than the N-Unicast model on mesh sizes of 8 and 10 for recipient group sizes of 1 through 14. For a mesh of size 6, the Conference Call model had a higher call blocking probability for group sizes of up to 12 recipients, and a lower call blocking probability beyond this value. The size D = 4 case displayed similar behaviour, but the cross-over point was at groups with 13 recipients. This overall increase in call blocking probability can be attributed to the same reasons as for the 10% multicast traffic workload.

The link usage savings realized by the Conference Call model were more apparent with the 20% multicast traffic workloads than with their 10% multicast counterparts (refer to Figure 5.9). With 20% multicast traffic, the Conference Call model averaged 4.2 links per call for a 4 x 4 mesh compared with 4.4 links per call for the N-Unicast model. Again, link usage savings increased with group size. For a

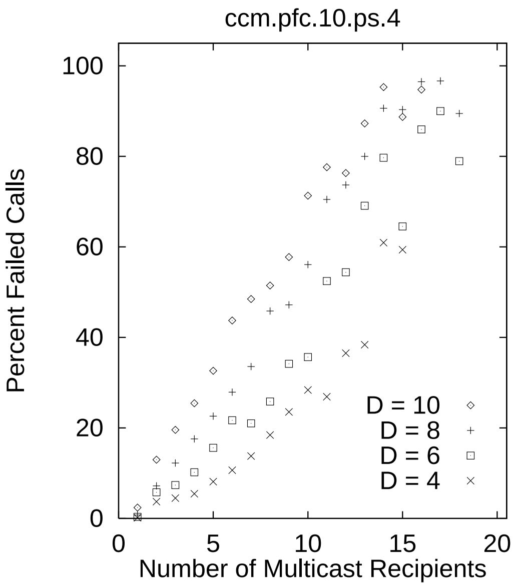

Figure 5.6: Percent failed calls, CC model, 10% MC traffic, g = 4.

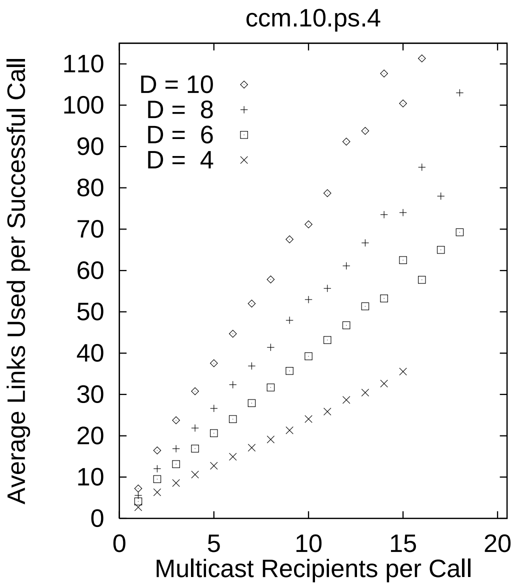

Figure 5.7: Avg. links per call, CC model, 10% MC traffic, g = 4.

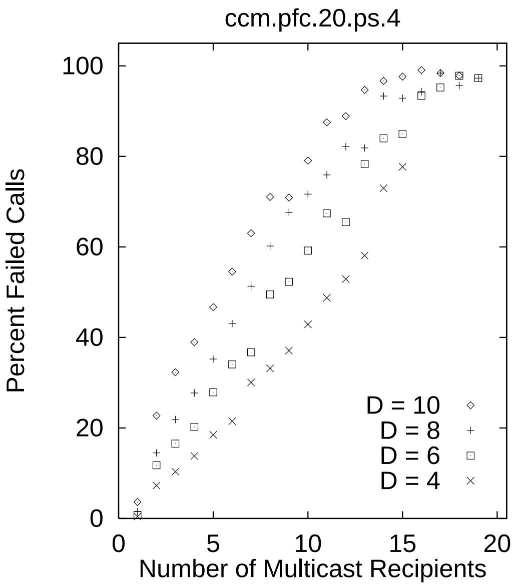

Figure 5.8: Percent failed calls, CC model, 20% MC traffic, g = 4.

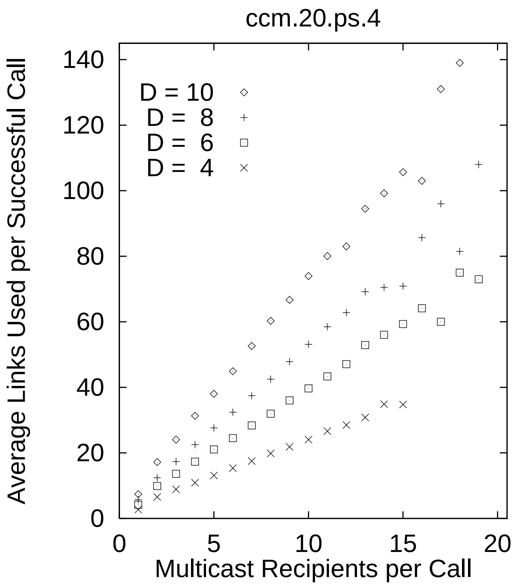

Figure 5.9: Avg. links per call, CC model, 20% MC traffic, g = 4.

10 x 10 mesh, the Conference Call model used 10.2 links per call on average, while the N-Unicast model average 10.8 links per call. As with the case of 10% multicast workload, this small link usage saving was expected.

5.3 Core Based Tree Model

The Core Based Tree (CBT) model is similar to the Conference Call model in that a designated multicast node other than the originator is used to replicate cells of the call and dispatch them to the recipients. In the CBT model, the multicast (core) node location is dynamically chosen based on the locations of the recipients (refer to Section 4.2.3). This model was expected to achieve lower call blocking probabilities and link usage than the Conference Call model since the core node would always be placed relatively central to the group of nodes containing the originator and the recipients. In this section, the simulation results presented are for the Core Based Tree model with a mean multicast group size of 4.

For the 10% multicast traffic workload, the non-shared link Core Based Tree model attained an average call blocking probability of 0.5% on a 4 x 4 mesh. The average call blocking probabilities for the other mesh sizes were 1.6% for a 6 x 6 mesh, 2.9% for a mesh of size 8 x 8, and 5.0% for a 10 x 10 grid (refer to Figure 5.10). As with the other two non-shared link models, the call blocking probability increased with mesh size. The performance results are shown in Figures 5.10 through 5.13.

The call blocking probability of the CBT model in relation to the N-Unicast and Conference Call model depended on the mesh size. For a 4 x 4 mesh, the CBT model had lower call blocking probabilities for all group sizes than either the N-Unicast or Conference Call model, with the differences becoming greater with increasing group size. For a mesh of size 6, the CBT model had a higher percentage of failed calls than the N-Unicast model for all group sizes except for sizes 8, 10, and 17. When compared with the Conference Call model, the CBT algorithm had a lower percentage of failed calls for all but the largest group sizes. In the case of an 8 x 8 mesh, the CBT model had a higher call blocking probability than the N-Unicast model for all multicast group sizes, but achieved lower call blocking probabilities than the Conference Call model for recipient groups of size 2 through 5. For the 10 x 10 mesh, the CBT model had higher call blocking probabilities than the other two routing models for almost all multicast group sizes. These results appear to be counter-intuitive, since it was expected that the greatest reductions in call blocking would appear in the larger mesh sizes and not in the smaller meshes. However at 1.2% unicast call blocking for a 10 x 10 mesh, the CBT model does have a much lower unicast call blocking probability than the other two models (2.7% for the N-Unicast and 2.4% for the Conference Call model for the same size mesh). The net result is overall call blocking performance that is better than the N-Unicast model, but slightly worse than the Conference Call model for the D = 10 mesh. It appears that the non-shared link CBT model routes unicast calls at the expense of multicast calls in large meshes.

As expected, the CBT model consistently had the lowest average link usage of the three non-shared link routing models. For workloads with 10% multicast traffic, the CBT model averaged 3.4 links per call for a 4 x 4 mesh, compared with 3.5 links per call for both the N-Unicast and Conference Call models. Link usage savings with the CBT model also improved with mesh size. For a 10 x 10 mesh, the CBT model averaged 8.0 links per call while the N-Unicast and Conference Call models averaged 8.9 and 9.1 links per call, respectively. The graph of links used per multicast call as a function of group size appears in Figure 5.11. It was again observed that the average links used per call increased linearly with group size, and for a given group size, link usage increased with mesh size.

For the 20% multicast traffic workload, the Basic CBT model attained an average call blocking probability of 1.9% on a 4 x 4 mesh. The average call blocking probability for the other mesh sizes were 4.8% for a 6 x 6 mesh, 7.8% for an 8 x 8 mesh, and 10.4% for a 10 x 10 grid. A breakdown of call blocking probability as a function of group size is shown in Figure 5.12. As expected, these values were larger than for the 10% multicast load, and the relationship between call blocking probability and mesh size remained unchanged. In addition, the general trend of call blocking being lower in the CBT model than the N-Unicast and Conference Call for the smaller meshes but greater in the larger meshes reappeared in the 20% multicast traffic workload.

With 20% multicast traffic workloads, the CBT model continued to have lower average link usage than the other two routing models. The increase in multicast traffic made the differences in link usage more apparent. With a 4 x 4 mesh and 20% multicast traffic, the CBT model averaged 4.1 links per call compared with 4.4 links per call for the N-Unicast model and 4.2 links per call for the Conference Call model. For a 10 x 10 mesh, the CBT model required 8.8 links per call on average, compared with 10.8 links per call and 10.2 links per call for the N-Unicast and Conference Call model, respectively. The graph of average link usage per call as a function of group size for 20% multicast traffic appears in Figure 5.13.

5.4 Comparison of Models

The call blocking and link usage performance of the three non-shared link routing models were compared using confidence intervals to facilitate a more rigorous evaluation of the models. For link usage, comparisons were also made with theoretically optimal link usage values for all three models, and with a formula for the N-Unicast and Conference Call models. This section presents the details of these analyses.

5.4.1 Call Blocking

The same batch means analysis technique used in Section 5.1 was employed to determine confidence intervals for call blocking and link usage results for each routing algorithm. For each model, the mean and standard deviation of the call blocking probability were tabulated according to group size for unicast calls, multicast calls, and all calls. These statistics were derived from batch means analysis with 10 batches, each of size 10,000 calls, as in Section 5.1. A comparison of the three routing models broken down according to unicast, multicast, and combined calls

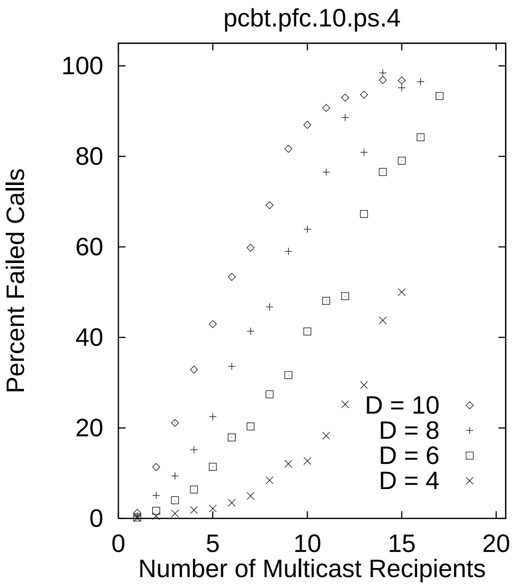

Figure 5.10: Percent failed calls, CBT model, 10% MC traffic, g = 4.

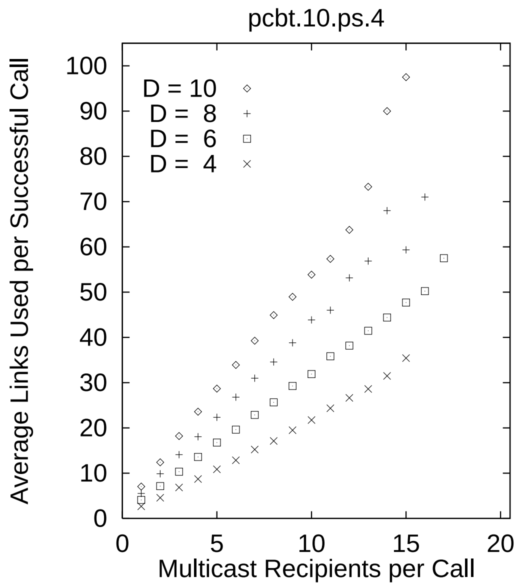

Figure 5.11: Avg. links per call, CBT model, 10% MC traffic, g = 4.

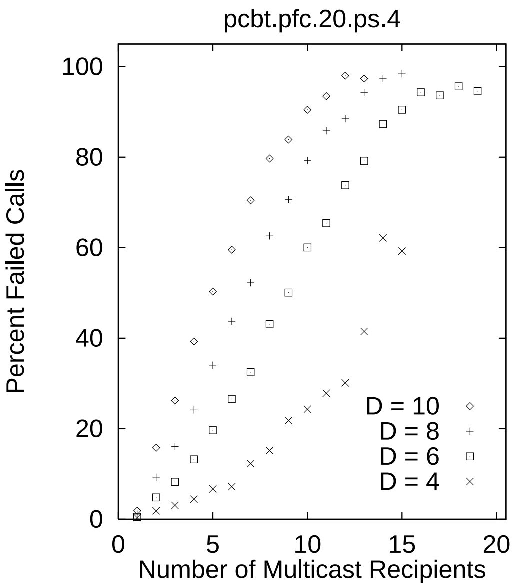

Figure 5.12: Percent failed calls, CBT model, 20% MC traffic, g = 4.

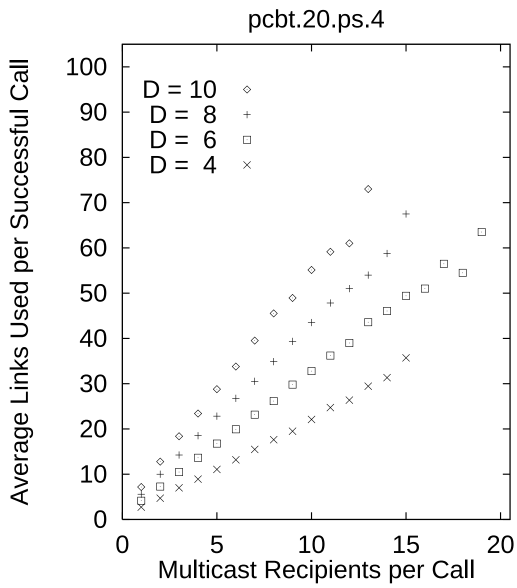

Figure 5.13: Avg. links per call, CBT model, 20% MC traffic, g = 4.

will be covered in this subsection followed by more specific analysis of the individual routing algorithms.

The 95% confidence interval plots of unicast call blocking probability for all 4 mesh sizes appear in Figure 5.14. The workload consisted of 10% multicast traffic with a mean multicast group size of 4. The confidence intervals overlapped in only three instances. Two comparisons between the Conference Call and the CBT model were inconclusive for the D = 4 and D = 6 mesh sizes, although they were not as inconclusive as between the N-Unicast and Conference Call model for the D = 10 mesh size. In any case, the trend was that the CBT model had the lowest unicast call blocking probability, followed by the Conference Call model, and finally the N-Unicast model with the highest call failure rate. This behaviour was expected, although these results would be best used as a relative indicator of how many free channels there are in the mesh capable of routing a unicast call.

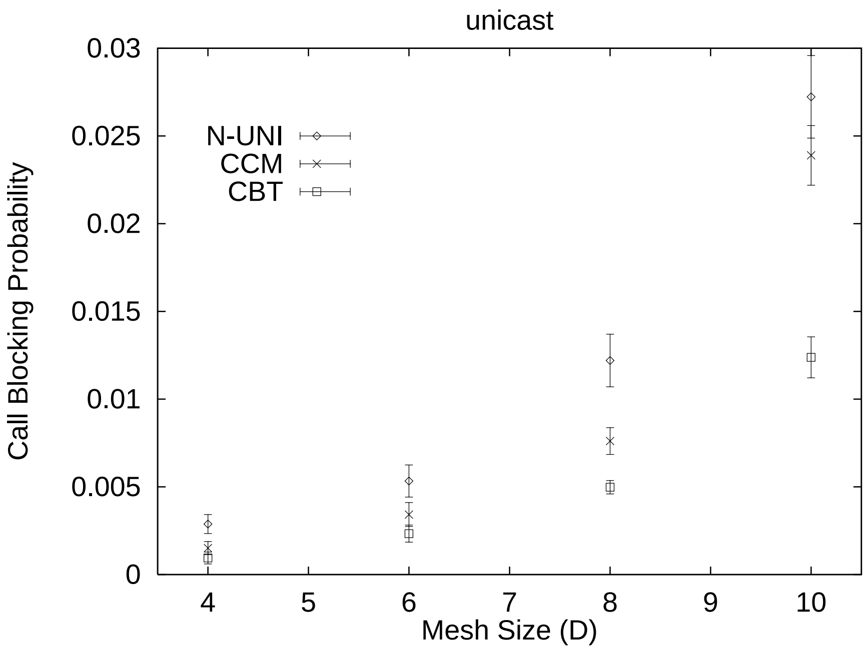

Figure 5.14: Unicast call blocking probability: observed values with 95% confidence intervals for the N-Unicast, Conference Call, and CBT models (all mesh sizes with 10% multicast traffic, g = 4).

The 95% confidence interval comparison plots of call blocking probability for multicast calls of all call sizes appear in Figure 5.15. All intervals are distinct except for the interval overlap between the CBT model and the Conference Call model for the 8 x 8 mesh size. The N-Unicast model consistently has a lower multicast call blocking probability than the Conference Call model. The CBT model, however, starts out with the lowest multicast call blocking probability at D = 4, falls between the N-Unicast and Conference Call model at D = 6, is about the same as the Conference Call model at D = 8, and has the highest blocking probability for D = 10. The most likely cause of this behaviour is that the CBT model tends to favour unicast calls, particularly in larger meshes (as mentioned in Section 5.3). This would impair the successful completion of multicast calls and therefore increase the multicast call blocking probability.

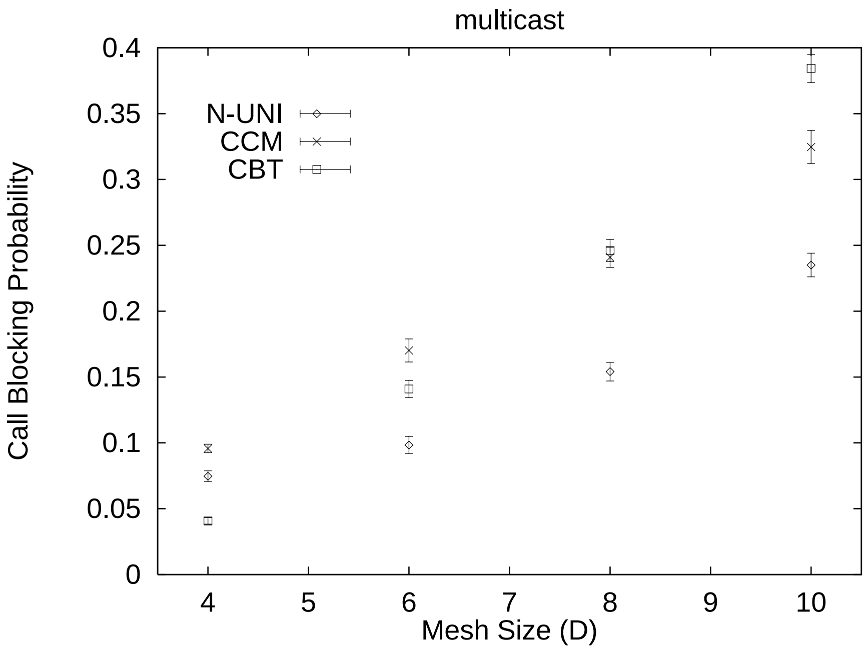

Figure 5.15: Multicast call blocking probability: observed values with 95% confidence intervals for the N-Unicast, Conference Call, and CBT models (all mesh sizes with 10% multicast traffic, g = 4).

The 95% confidence interval comparison plots for overall call blocking probability appear (not surprisingly) to be a hybrid of the unicast and multicast plots previously discussed (refer to Figure 5.16). Although the multicast calls had roughly an order of magnitude higher call blocking probability than the unicast calls, the combination of 90% unicast and 10% multicast calls results in the composite plot shown. Since the confidence intervals overlap slightly, conclusive comparisons cannot be made. For D = 4, the confidence intervals between the N-Unicast and Conference Call model overlap, while for D = 6, the confidence intervals overlap for the Conference Call and the CBT models. For D = 8 and D = 10 there is an overlap between the confidence intervals of the N-Unicast and Conference Call model, and between the Conference Call model and the CBT model. If the confidence limits were relaxed, some general trends can be observed. At D = 4, the CBT model has the lowest call blocking probability, followed by the Conference Call model and the N-Unicast model. For the remaining mesh sizes, the Conference Call model has the lowest call blocking probability, followed by CBT model, and finally the N-Unicast model with the highest probability of call blocking. The poor multicast performance of the CBT model was offset by its good unicast performance, resulting in its second-place showing for the three largest mesh sizes.

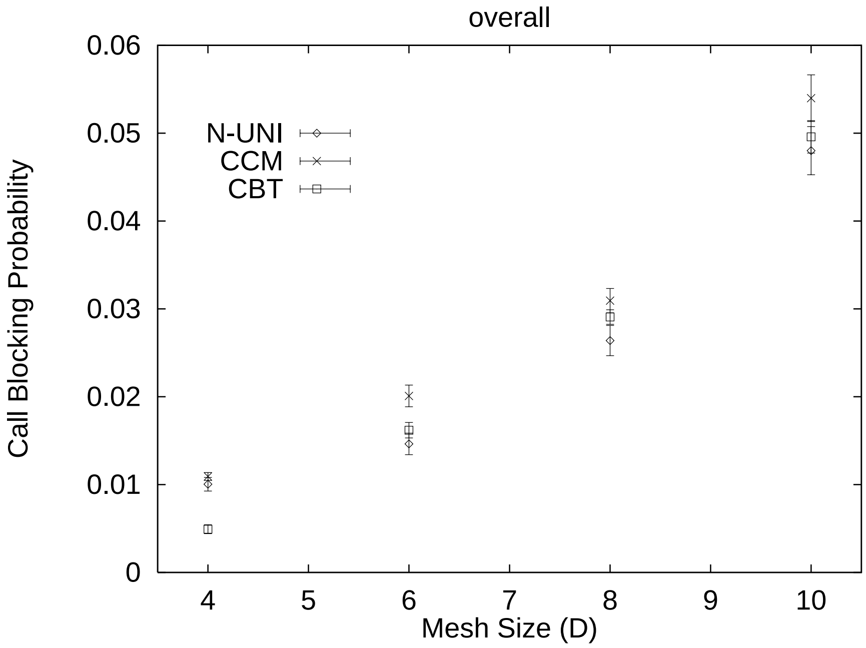

Figure 5.16: Overall call blocking probability: observed values with 95% confidence intervals for the N-Unicast, Conference Call, and CBT models (all mesh sizes with 10% multicast traffic, g = 4).

N-Unicast Model

The call blocking results for the N-Unicast routing algorithm are summarized in Table 5.1. As expected, the mean call blocking probability is much lower for the unicast calls than for the multicast calls. For all categories of calls, the standard deviation of the call blocking probability was about an order of magnitude smaller than the corresponding mean call blocking probability. These results imply tight confidence intervals when comparing call blocking performance between routing models. These confidence interval bounds would be reasonably tight for all but the largest recipient group sizes.

Table 5.1: Mean and Standard Deviation of the Call Blocking Probability for the N-Unicast routing model with 10% multicast traffic (mean multicast group size = 4).

Mesh Size	Call Blocking Probability for N-Unicast Model					
	Unicast Calls		Multicast Calls		All Calls	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
4	0.0029	0.0008	0.0746	0.0066	0.0100	0.0012
6	0.0053	0.0015	0.0983	0.0106	0.0146	0.0020
8	0.0122	0.0024	0.1541	0.0115	0.0264	0.0028
10	0.0272	0.0038	0.2351	0.0145	0.0480	0.0044

Conference Call Model

A summary of the call blocking performance of the Conference Call model appears in Table 5.2. As with the N-Unicast model, the standard deviation of each call blocking probability was approximately an order of magnitude smaller than its corresponding mean value.

CBT Model

The mean and standard deviations of the call blocking behaviour for the CBT model appears in Table 5.3. As with the N-Unicast and Conference Call models,

Table 5.2: Mean and Standard Deviation of the Call Blocking Probability for the Conference Call routing model with 10% multicast traffic (mean multicast group size = 4).

Mesh Size	Call Blocking Probability for the Conf. Call Model					
	Unicast Calls		Multicast Calls		All Calls	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
4	0.0015	0.0006	0.0957	0.0052	0.0109	0.0007
6	0.0034	0.0011	0.1702	0.0141	0.0201	0.0020
8	0.0076	0.0012	0.2409	0.0123	0.0309	0.0022
10	0.0239	0.0027	0.3247	0.0203	0.0540	0.0042

the standard deviation of each call blocking probability was approximately an order of magnitude smaller than its corresponding mean value.

The values reported in Tables 5.1, 5.2, and 5.3 facilitate a more rigorous comparison of the call blocking performance of the three routing models. For example, for a 4 x 4 mesh, the CBT model provides call blocking performance superior to that of the N-Unicast model for unicast calls, for multicast calls, and for overall calls. This result applies with a very high degree of statistical significance (> 95%). As another example, the bias of the CBT algorithm against multicast calls on large meshes (D = 10) is statistically significant when compared to the N-Unicast algorithm. On the other hand, there is very little statistical difference in the overall call blocking part of the three algorithms on 10 x 10 meshes.

Table 5.3: Mean and Standard Deviation of the Call Blocking Probability for the CBT routing model with 10% multicast traffic (mean multicast group size = 4).

Mesh Size	Call Blocking Probability for the CBT Model					
	Unicast Calls		Multicast Calls		All Calls	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
4	0.0009	0.0005	0.0407	0.0040	0.0049	0.0008
6	0.0023	0.0007	0.1409	0.0104	0.0162	0.0014
8	0.0050	0.0006	0.2461	0.0136	0.0291	0.0013
10	0.0124	0.0019	0.3844	0.0173	0.0496	0.0030

5.4.2 Link Usage

Comparisons of link usage per call were performed between routing models. In general, great care must be taken in such comparisons, since the blocking probabilities may differ for each algorithm. Although the actual successful calls being processed by the routing models were different due to differences in call blocking characteristics, the comparisons made in this work were considered valid for three reasons. First, the offered workload to each algorithm is identical. This means that all three algorithms, despite their routing differences, experience busy and idle periods at the same point in time. Thus a specific call is quite likely to be accepted by all three algorithms or rejected by all three algorithms. While this behaviour is obviously not universally true, it does apply much of the time, particularly during start-up periods, at regeneration points, and for many unicast calls (the bulk of the workload). Thus there is a large intersection in the set of successful calls for each algorithm. For example, a comparison was made of the successful calls (based on a call identifier) for each of the three routing models on a 4 x 4 mesh. The largest difference found was between the N-Unicast model and the Conference Call model, which differed on 1343 calls (1.34% of the total calls offered to the network). The other model combinations differed by 1% or less. Second, in small mesh sizes the call blocking probability for small and medium group sizes is small. Thus, the differences in the actual successful calls being processed in relation to the entire workload would be small. Third, comparisons between routing models are only made when there is a sufficient number (> 50) of calls of a given multicast group size to make statistically significant comparisons (using confidence intervals).

A three-fold analysis of the link usage for the three non-shared link routing algorithms was conducted. First, 95% confidence intervals for the links per call metric were established to determine the group sizes for which differences in this metric were statistically significant. To achieve this, the mean and standard deviation in link usage for each recipient group size were computed over the entire run of 100,000 simulated calls. Second, the observed link usages were compared with empirically determined optimal values to evaluate the efficiency of the routing algorithms. Finally, the observed values were compared with an analytical formula for the N-Unicast and Conference call models to determine if link usages matched the analytical results.

For each algorithm the optimal link usage was determined using a two-pass method for each workload. During the first pass, the blocked calls were determined and excluded from the workload during the second pass. In the second pass, the number of links required to complete each call was calculated by determining the difference in the x and y coordinates between each pair of nodes instead of using the BFS routing algorithm. The result is the minimum possible number of links needed to complete a given call for each of the three non-shared link routing algorithms, assuming an unloaded network. This analysis provides a lower bound on the number of links required for each call. The actual routing algorithms may use more than this number of links because of congestion within the network, or suboptimality in the routes chosen.

The final part of the link usage analysis used a formula derived by Williamson [32], which calculates the average distance from any point in a mesh network to all other points in the network, as well as the average distance between any two points in the network. For an r x c sized rectangular mesh, the average distance d from a point (i,j) (where $1 \leq i \leq r$ and $1 \leq j \leq c$) to all other points in the mesh is: [32]

$$d = \frac{rc(r + c - 2(i + j - 1)) + 2rj(j - 1) + 2ci(i - 1)}{2(rc - 1)}$$

The average distance between any two (distinct) random points in the mesh can be computed by summing the distances and taking the average. The equation thus becomes:

$$\bar{d} = \frac{1}{rc} \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{rc(r + c - 2(i + j - 1)) + 2rj(j - 1) + 2ci(i - 1)}{2(rc - 1)}$$

where $\bar{d}$ is the average value of d over the whole mesh. For the N-Unicast model, the average number of links per call for a given recipient group size N is equal to $N\bar{d}$. For the Conference Call model the average number of links per multicast call is $\bar{d}_1 + N\bar{d}_2$, where $\bar{d}_1$ is the average distance from the source node to the conference hub in that node's region (if applicable) and $\bar{d}_2$ is the average distance from that hub to all other points in the mesh. The CBT model does not easily lend itself to analysis of the average links used per call because the core is not statically located. Thus the CBT model is excluded in this part of the analysis.

These formulae do not consider points that coincide, *i.e.,* source and destination nodes are the same. While this situation is not allowed by the workload generator, there is the possibility that the source of a multicast call may coincide with a hub node for the conference call model, or that a recipient in a multicast call may coincide with a hub node in the same model (the N-Unicast model does not have this problem). However, these occurrences are relatively rare. For example, in a 4 x 4 mesh, the chance of this occurring in a multicast call is 1 in 16 for a source node coinciding with a hub node, and 4 in 16 for a recipient coinciding with a hub node when the mean multicast group size is 4. Since these two events are mutually exclusive, the probability of having a zero distance path in a multicast call would be 5 in 16 or 0.3125. For workloads with a mean multicast group size of 4, the probability of having a zero distance path in a multicast call for D = 6, D = 8, and D = 10 would be 0.1389, 0.0781, and 0.05, respectively. In any case, the error caused by these cases is small.

Link Usage Comparison Between Routing Models

Figures 5.17, 5.18, 5.19, and 5.20 depict the link usage per call confidence intervals for the 4 x 4, 6 x 6, 8 x 8, and 10 x 10 meshes, respectively, for the N-Unicast, Conference Call, and CBT models. The workloads used had 10% multicast traffic and a mean multicast group size of 4.

The confidence intervals were small for group sizes smaller than 8 recipients on all mesh sizes. Of course, the size of the confidence intervals grew rapidly with

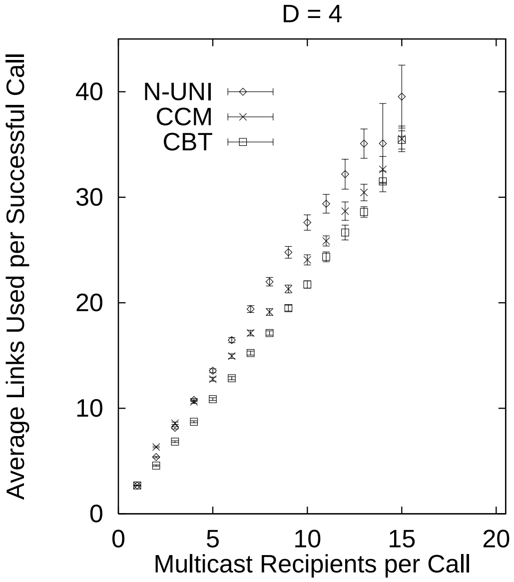

Figure 5.17: Avg. links per call 95% confidence intervals for D = 4, 10% MC traffic, g = 4.

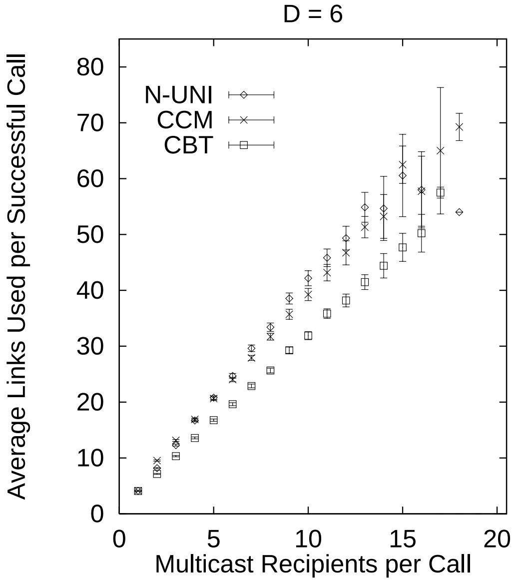

Figure 5.18: Avg. links per call 95% confidence intervals for D = 6, 10% MC traffic, g = 4.

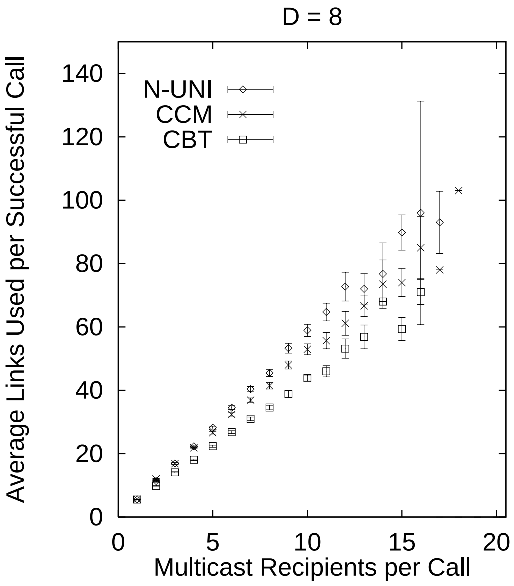

Figure 5.19: Avg. links per call 95% confidence intervals for D = 8, 10% MC traffic, g = 4.

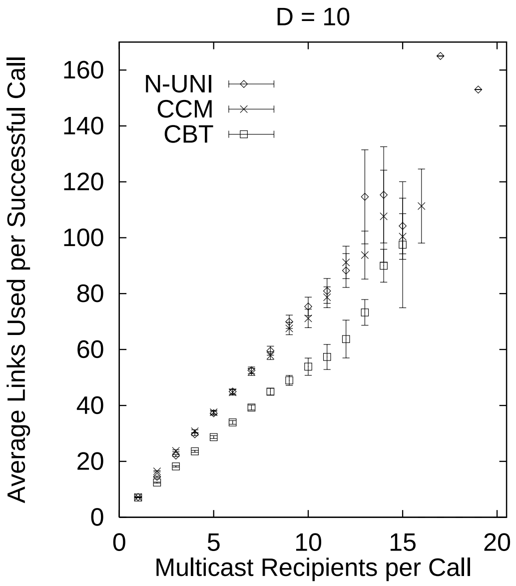

Figure 5.20: Avg. links per call 95% confidence intervals for D = 10, 10% MC traffic, g = 4.

increasing group size for larger groups, since there were few samples at these sizes, resulting in large variances and large confidence intervals. The largest group sizes for the D = 8 and D = 10 meshes had much larger confidence intervals than their counterparts in the D = 4 and D = 6 sizes. This could be attributed to the larger meshes allowing a wider range of routing possibilities than the smaller meshes, which increased the variance in the number of links required per call.

The link usage of the N-Unicast and Conference Call model displayed what appeared to be unusual behaviour for small recipient group sizes. In Figures 5.17 through 5.20 the Conference Call model actually requires more links to complete a call for groups of 2 and 3 recipients than the N-Unicast model. The plots then return to their expected behaviours for larger group sizes. If confidence intervals are considered, this overlap ends at 4, 5, 5, and 8 recipients for D = 4, 6, 8, and 10, respectively. Beyond these group sizes for the D = 4, 6, and 8 mesh sizes, and until the largest group sizes (where the confidence intervals become large enough to overlap), the Conference Call model used fewer links per call than the N-Unicast model. The confidence intervals for the Conference Call and N-Unicast model for the D = 10 model overlapped for all groups larger than 4, resulting in an inconclusive comparison between these two models.

The small group size link usage anomaly can be attributed to the additional link overhead required for routing via the conference hub in the Conference Call model, which becomes especially noticeable in the links per call metric for small group sizes. In larger mesh sizes the effect is more spread out due to the use of multiple regional hubs which result in a lower additional link cost for small recipient group sizes, but a larger total effect for large group sizes due to the non-central location of the regional hubs (recall that only the one regional hub closest to the source node is used for multicast calls in large meshes).

For all but the largest multicast group sizes, the CBT model's confidence intervals did not overlap the confidence intervals of the other two models for all mesh sizes tested. This confirms that the CBT model consistently requires the lowest average number of links to complete a call.

N-Unicast Link Usage

The plots of average links used per multicast call versus the number of multicast recipients per call were close to linear for both the 10% and 20% multicast traffic cases on all four mesh sizes for a mean multicast group size of 4 (see Figures 5.2 and 5.4). Linear regression analysis by the least-squares method [18] was performed on this and all other link plots to quantify the linearity of the link usage by the simulation models.

Table 5.4 summarizes the results of the regression analysis for the N-Unicast routing model with a mean multicast group size of 4 members. All link usage results fit very well with a linear model, since the R^2 coefficient for the regression is consistently larger than 0.9. Table 5.4 also shows the slopes and y intercepts determined by the linear regression. For comparison, the slopes of the link usage based on the average node distance formulas are also listed with the observed values. These values will be referred to as the “expected average” links per call in the ensuing comparisons. Note that in the case of the N-Unicast model, the expected average links per call is equal to $\frac{2D}{3}$.

Table 5.4 shows that the N-Unicast routing model routed better than the expected average of 2.67 links per call on the D = 4 mesh for both the 10% and 20% multicast workloads (2.61 and 2.60 links per call, respectively). The algorithm also routes better than the expected average on the D = 6 mesh for 10% multicast workloads (3.52 links per call vs. 4.00 links per call), and about equal to average for the 20% multicast workload at 4.04 links per call. The D = 8 mesh used more links than average of 5.33 links per call to complete a call for the 10% multicast workload (5.77 links per call), and the situation worsened with 20% multicast traffic (6.31 links per call required). As mesh sizes increased, link usage per call diverged further from the expected averages, reflecting greater congestion in the network and more suboptimality in the routes chosen.

Link usage usually increased with the percentage of multicast traffic as well. However, for a D = 10 N-Unicast model that required a calculated 6.67 links per call, the 10% multicast workload required 8.52 links per call while the 20% multicast workload required 7.59 links per call. Although this case seemed contradictory to the trend of heavier multicast workloads requiring increasingly more links per call, the difference can be attributed to two outlier data points (see Figure 5.2) that skew the regression for the 10% multicast workload. The y intercept for this case is quite far below the x-axis, while most of the other regression results are close to or above the x-axis. The R^2 coefficient is also lower than that for most of the other regressions.

Table 5.4: Regression Coefficients for Link Usage in the N-Unicast model (mean multicast group size = 4).

Mesh Size	Percent MC Traffic	Formula Slope	Linear Regression		
			Slope	y intercept	R^2
4	10	2.67	2.61	0.57	0.996
	20		2.60	0.71	0.998
6	10	4.00	3.52	3.84	0.947
	20		4.04	1.11	0.980
8	10	5.33	5.77	-0.02	0.993
	20		6.31	-2.64	0.982
10	10	6.67	8.52	-5.55	0.963
	20		7.59	0.44	0.999

A plot of the links used per call for the 4 x 4 N-Unicast model (with 10% multicast traffic and a mean multicast group size of 4) broken down according to group size appears in Figure 5.21. The mean links per call are shown with 95% confidence intervals along with the formula calculated (expected average) values and optimal link usage values. Since the curves are very similar to each other, Figure 5.22 plots the same data normalized with respect to the optimal routing curve to emphasize the differences in value.

Figure 5.22 shows that the N-Unicast routing model is consistently within 5% of the optimal routing distance for all but the largest group sizes. The N-Unicast model becomes increasingly less optimal with increasing recipient group size. Still, the upper bounds of the observed values all fell within 15% of optimal, with the formula derived values being within confidence limits for multicast group sizes of 2, 4, 5, and 11 to 15. For all other multicast group sizes, the formula values were between the observed and optimal values. Also, the formula derived link usage values were very close to optimal for group sizes of up to 7 recipients, but began to diverge for larger group sizes. While the D = 6 mesh size (not shown) exhibits similar tendencies, the formula generated values for the D = 8 and D = 10 meshes stay very close to the optimal values. Increasing the percentage of multicast traffic from 10% to 20% had little effect on these results. It would appear that the N-Unicast formula was most realistic for small meshes, with optimistic predictions for larger mesh sizes, since the formula ignores link congestion in its calculations.

Conference Call Link Usage

As with the N-Unicast model, the relationship of average link usage per call to multicast group size was linear (refer to Table 5.5). Due to the source-to-hub and hub-to-recipients routing method of the conference call model, only the multicast portion of the formula-derived curve slopes are presented. This is because in large meshes with regional hubs, the average distance from source to hub is significantly less than the average distance from hub to recipients, resulting in a knee in the link usage curve.

Overall, the regression derived slopes of the observed link usage plots were lower than those of the N-Unicast model. The only exception to this trend was for the D = 6 mesh with 10% multicast traffic, which can be explained by the N-Unicast case having a poorer fit (R^2 = 0.947) and a higher intercept on the y-axis (y = 3.84) than the Conference Call case (R^2 =0.994 and y = 1.66).

The formula also predicts lower link usage for the Conference Call model than for the N-Unicast model (compare Tables 5.4 and 5.5). In the Conference Call model, there is less of a divergence from the formula in links per call used for the large mesh sizes than in the N-Unicast model. In addition, the differences in the slopes of the observed links per call plots are much smaller than in the N-Unicast model. For example, for the D = 4 mesh, the formula slope value was 2.13 links

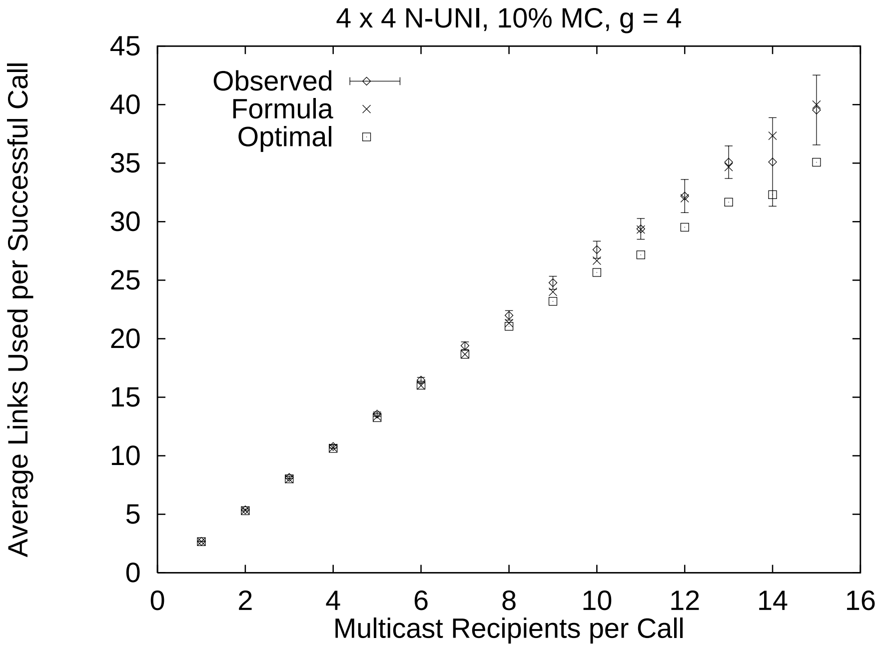

Figure 5.21: Average links per call: observed values with 95% confidence intervals, formula values, and optimal values for the N-Unicast model, D = 4, 10% multicast traffic, g = 4.

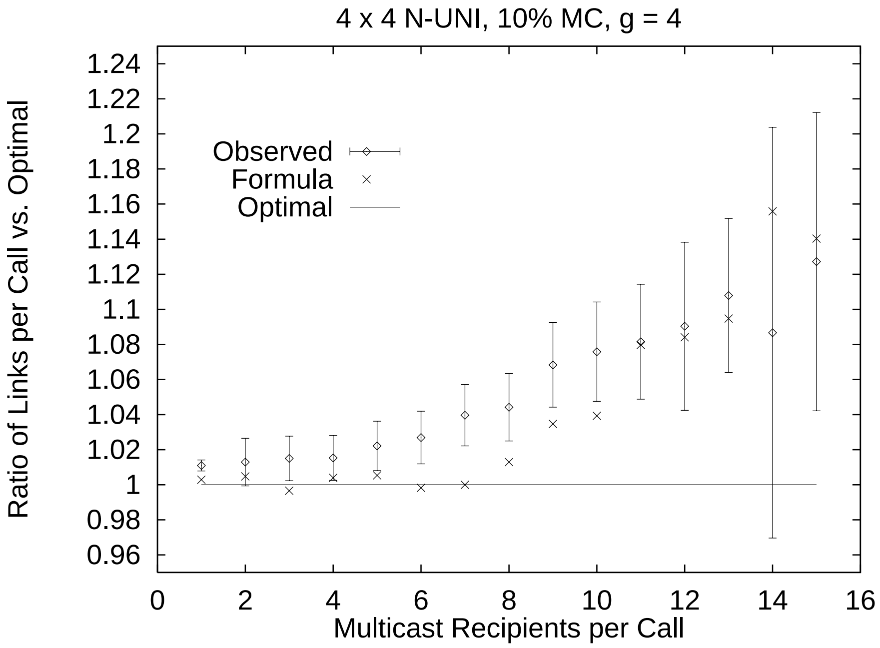

Figure 5.22: Average links per call: observed values with 95% confidence intervals and formula values normalized against optimal link usage for the N-Unicast model, D = 4, 10% multicast traffic, g = 4.

per call, while for the 10% and 20% multicast traffic workloads, the observed slopes were 2.25 and 2.27 links per call, respectively. For the D = 10 mesh the formula slope value was 6.26 links per call, while the observed slopes were 6.99 links per call for the 10% multicast workload and 7.12 links per call for the 20% multicast workload. In all cases, the observed links per call plots had higher slope values than the formula slopes, since the formula considers only node distance, not link congestion.

Table 5.5: Regression Coefficients for Link Usage in the Conference Call Model (mean multicast group size = 4).

Mesh Size	Percent MC Traffic	Formula Slope	Linear Regression		
			Slope	y intercept	R^2
4	10	2.13	2.25	1.35	0.999
	20		2.27	1.58	0.996
6	10	3.43	3.75	1.66	0.994
	20		3.81	1.73	0.991
8	10	4.57	5.09	1.14	0.982
	20		5.12	1.56	0.976
10	10	6.26	6.99	2.56	0.991
	20		7.12	1.87	0.986

The plot of observed links per call, formula derived links per call, and optimal links per call values for the Conference Call model appear in Figure 5.23. The same data normalized to the optimal values appears in Figure 5.24. Again, the case of a 4 x 4 mesh with 10% multicast traffic and a mean multicast group size of 4 was studied.

For all multicast group sizes except 10, 12, and 15 recipients, the formula derived values fell well within the confidence intervals of the observed links per call values. This would attest to the accuracy of the formula for the 4 x 4 mesh. However, with larger mesh sizes (not shown) the formula values usually were well below the confidence intervals of the observed values and were very close to optimal link usage. As with the N-Unicast model, the observed values were within 15% of optimal, but the divergence from optimality did not have the same pattern as the

N-Unicast model. In fact, there was a significant difference in the optimality of link usage between the unicast and multicast calls. Figure 5.24 shows that the Conference Call model routes unicast calls within 2% of the optimal link usage values, while multicast calls were all more than 6% above optimal link usage. This differs significantly from the N-Unicast model which routed to within 5% of optimal link usage for calls of less than 9 recipients. This observation is consistent with the observations pertaining to Figures 5.17 through 5.20 where the Conference Call model required more links to complete a multicast call than the N-unicast model for small group sizes. As with the N-Unicast model, going to 20% multicast traffic did not have a significant impact on these results.

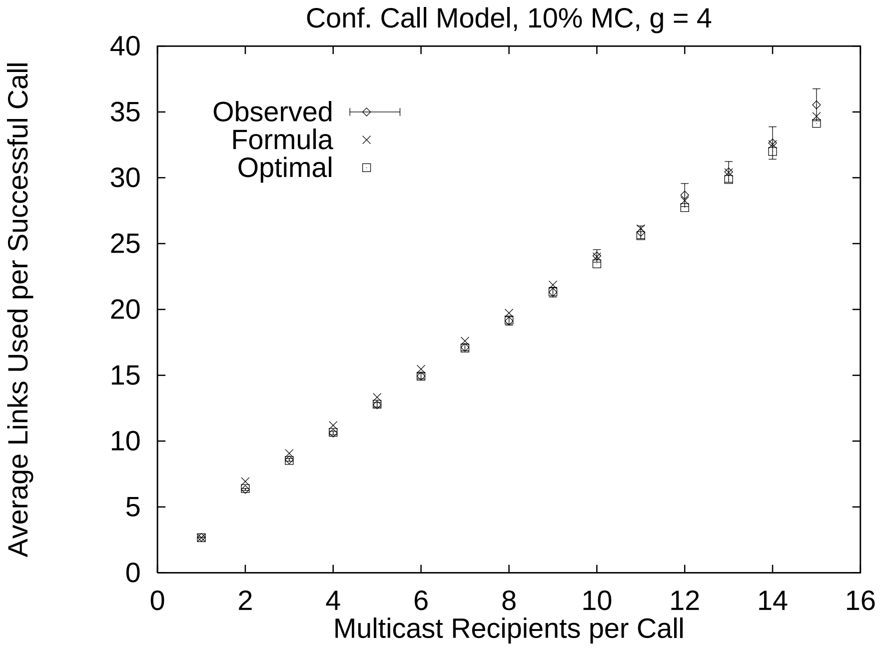

Figure 5.23: Average links per call: observed values with 95% confidence intervals, formula values, and optimal values for the Conference Call model, D = 4, 10% multicast traffic, g = 4.

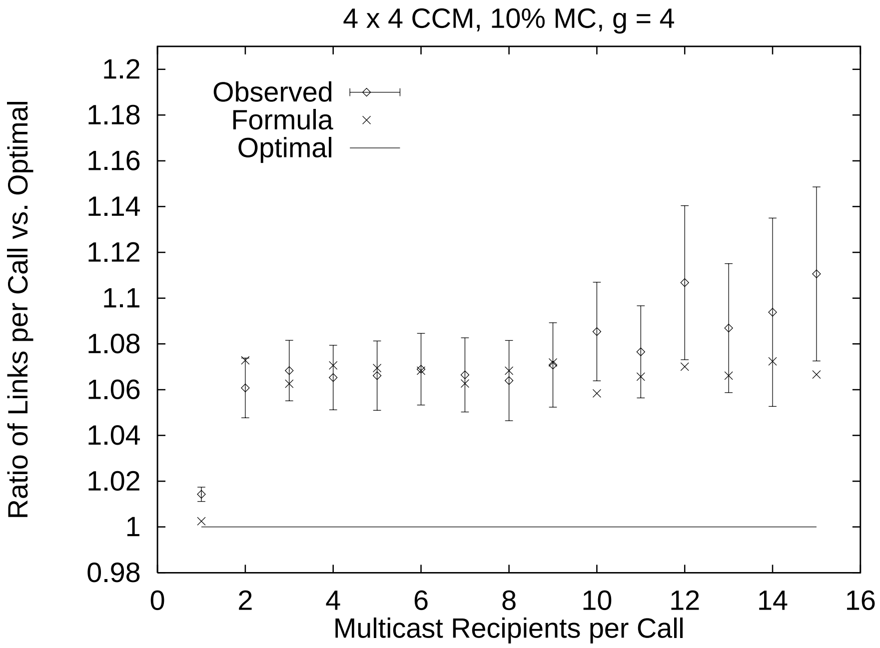

Figure 5.24: Average links per call: observed values with 95% confidence intervals and formula values normalized against optimal link usage for the Conference Call model, D = 4, 10% multicast traffic, g = 4.

CBT Link Usage

As expected, the relationship of average links used per call to recipient group size for the CBT model is linear (refer to Table 5.6). As discussed previously, the formula derived values for link usage were not available for this simulation model. While the slopes for the D = 4 mesh were comparable to the Conference Call model, they were smaller than those of the Conference Call and N-Unicast model for all other mesh sizes. The difference in slopes between the CBT model and the other two routing algorithms increased with mesh size.

Table 5.6: Regression Coefficients for Link Usage in the CBT model (mean multicast group size = 4).

Mesh Size	Percent MC Traffic	Linear Regression		
		Slope	y intercept	R^2
4	10	2.26	-0.33	0.996
	20	2.27	-0.17	0.996
6	10	3.17	0.66	0.997
	20	3.19	0.92	0.996
8	10	4.30	0.93	0.985
	20	4.19	1.53	0.997
10	10	5.97	-1.54	0.971
	20	5.20	2.57	0.994

The observed and optimal link usage plots for the D = 4 CBT model with 10% multicast traffic and a mean multicast group size of 4 appear in Figure 5.25 while the values normalized to the optimal appear in Figure 5.26. For this model, the observed links per call fell within 14% of optimal, and were generally closer to optimal than those of the Conference Call model, but not as close to optimal as the N-Unicast model for group sizes less than 7. Increasing the mesh size or the multicast traffic level to 20% had little effect on the results apart from broadening the confidence levels of the largest groups.

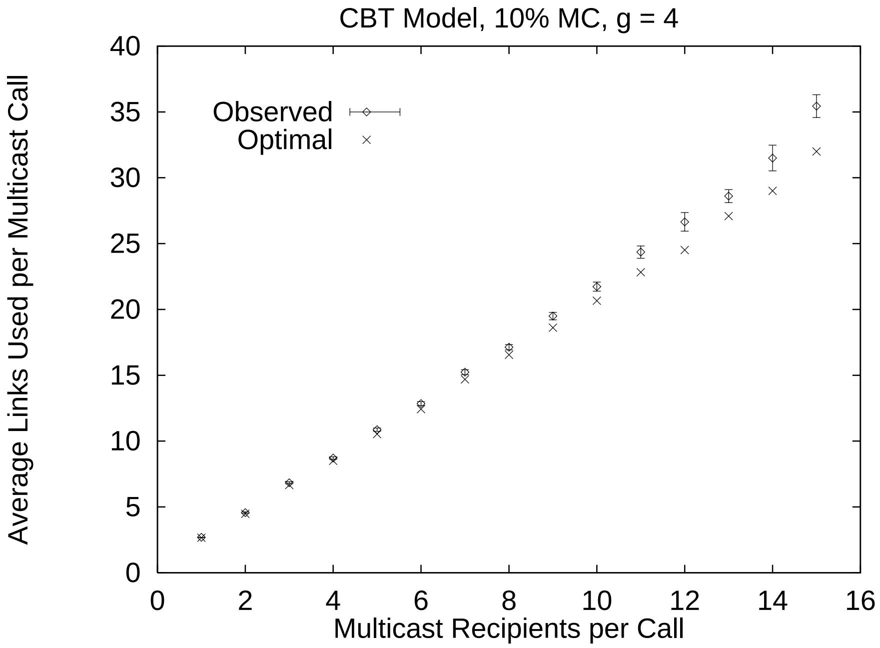

Figure 5.25: Average links per call: observed values with 95% confidence intervals and optimal values for the Conference Call model, D = 4, 10% multicast traffic, g = 4.

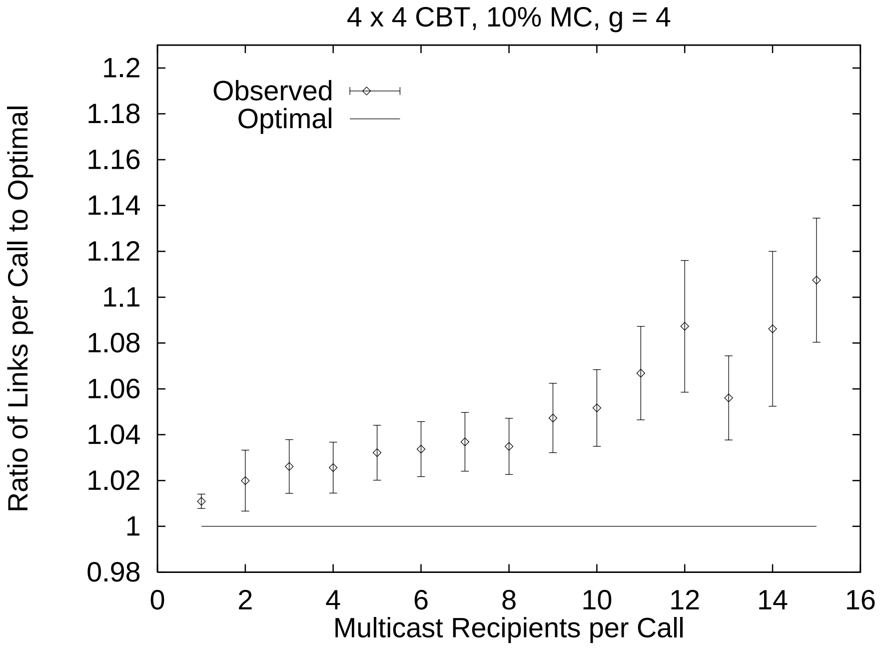

Figure 5.26: Average links per call: observed values with 95% confidence intervals normalized against optimal link usage for the Conference Call model, D = 4, 10% multicast traffic, g = 4.

5.5 Effect of Multicast Group Size

In Section 4.5, two levels of mean multicast group size were chosen for the simulation experiments: a mean group size of 4 to represent the expected distributions of early multicast groups, and a mean group size of 8 to represent larger multicast groups expected in the future. The latter mean group size setting also has the benefit of providing a larger number of data samples for large recipient group sizes. This section discusses the effect of mean multicast group size on call blockage and link usage. All figures and tables for this section are contained in Appendix A and Appendix B, respectively, since they are very similar in appearance to the figures and tables presented in Sections 5.1 to 5.3.

5.5.1 N-Unicast Model

The general trends described in Section 5.1 for the N-Unicast model continued to be observed when the mean multicast group size was raised from 4 to 8. Figure A.1 illustrates a breakdown of the call blocking probability as a function of group size for the 10% multicast traffic workloads, while Figure A.3 does likewise for the case of 20% multicast traffic.

Increasing the mean multicast group size increased call blockage for both the 10% and 20% multicast traffic workloads. The effect was most pronounced in small mesh sizes, and less so in large meshes. For example, for a 4 x 4 mesh with a 10% multicast workload, increasing the mean multicast group size from 4 to 8 resulted in the average call blocking rate increasing from 1.0% to 2.3%. For a 10 x 10 mesh with the same fraction of multicast traffic, the incidence of call failure increased from 4.8% to 6.8%. Similar but smaller increases occurred with the 20% multicast traffic model. In the case of a 4 x 4 mesh, increasing the multicast group size increased call blocking from 2.9% to 5.7%. With a 10 x 10 mesh, the increase was from 11.4% to 14.3%. This trend of call blockage increasing with mean multicast group size made intuitive sense: the more recipients there are in the network, the greater the chance of call blocking.

The plots of link usage with respect to group size for a mean multicast group size of 8 are shown in Figures A.2 and A.4 for the 10% and 20% multicast traffic workloads, respectively. Table B.1 summarizes the results of the regression analysis for the N-Unicast routing model with a mean multicast group size of 8 members. Unlike the situation with call blocking, the increase in mean multicast group size had only a very small effect on average link usage. For the case of 10% multicast traffic, increasing the mean multicast group size increased the average links used per call from 3.5 to 3.7 for a 4 x 4 mesh, while the links used per call for a 10 x 10 mesh remained unchanged at 9.1. Similarly, for workloads with 20% multicast traffic, increasing the mean multicast group size raised the link usage from 4.4 links per call to 4.7 per call for the 4 x 4 mesh, while in the 10 x 10 mesh link usage decreased slightly from 10.8 links per call to 10.7 per call. The ability of the larger sized mesh to easily absorb additional multicast traffic is most likely due to the increased number of alternate paths available to route the multicast calls.

5.5.2 Conference Call Model

The behaviour associated with increasing the mean multicast group size from 4 to 8 in the Conference Call model paralleled the case of the N-Unicast model in the previous subsection. Increasing the mean multicast group size from 4 to 8 resulted in increased call blockage, and the relative increase was greatest for the small mesh sizes. With the 10% multicast traffic workloads, increasing the mean multicast group size from 4 to 8 resulted in a rise in the percentage of failed calls from 1.1% to 2.0% for the 4 x 4 mesh, while for the 10 x 10 mesh, the call blocking probability grew from 5.4% to 7.2%. For the 20% multicast traffic workloads, the corresponding increase for the 4 x 4 mesh was from 4.2% to 4.5%. For a 10 x 10 mesh, however, increasing the mean multicast group size from 4 to 8 resulted in a small decrease in the call blocking probability from 10.2% to 10.1%. A breakdown of call blocking probability as a function of group size is shown in Figure A.5 for 10% multicast workloads and in Figure A.7 for the 20% multicast traffic workloads.

As was the case with the N-Unicast model, the average number of links used per call was insensitive to the mean multicast group size. For a 4 x 4 mesh with the 10% multicast traffic workload, the average links used per call for a mean multicast group size of 4 was 3.5 links per call, but for a workload with a mean multicast group size of 8, only 3.7 links were needed per call on average. For a 10 x 10 mesh with 10% multicast calls in the workload, increasing the mean multicast group size from 4 to 8 resulted in a decrease in link usage from 8.9 to 8.8 links per call. With 20% multicast traffic workloads, increasing the mean multicast group size from 4 to 8 resulted in an increase in link usage from 4.2 to 4.5 links per call in the 4 x 4 mesh, and a decrease from 10.2 to 10.1 links per call in the 10 x 10 mesh. A breakdown of link usage with respect to group size for 10% multicast traffic is shown in Figure A.6 and for 20% multicast traffic in Figure A.8. Table B.2 summarizes the results of the regression analysis for the Conference Call model with a mean multicast group size of 8 members.

5.5.3 Core Based Tree Model

The Core Based Tree model mimicked the behaviour of the other two shared link models when the mean multicast group size was raised from 4 to 8. Call blockage increased with both 10% and 20% multicast workloads with the largest relative increase occurring in the small mesh sizes. With 10% multicast traffic, the percent failed calls in the 4 x 4 mesh increased from 0.5% to 1.3% when the mean multicast group size was raised from 4 to 8. In a 10 x 10 mesh with the same amount of multicast traffic, the call blocking probability increased from 5.0% to 6.6%. For 20% multicast traffic, increasing the mean multicast group size increased the percentage of failed calls in the 4 x 4 mesh from 1.9% to 4.0%, and from 10.4% to 13.3% in the 10 x 10 mesh. For a more detailed breakdown of call blocking probability as a function of group size, refer to Figure A.9 for 10% multicast traffic workloads and to Figure A.11 for the 20% multicast traffic workloads.

As with the N-Unicast and Conference Call models, the average links used per call was relatively insensitive to changes in mean multicast group size. For the 10% multicast traffic workloads, increasing the mean multicast group size from 4 to 8 increased the average links used per call from 3.4 to 3.6 for a 4 x 4 mesh, while link usage decreased from 8.0 to 7.9 links per call in a 10 x 10 mesh. With 20% multicast traffic, increasing the group size caused the average link usage to increase from 4.1 links per call to 4.5 for the 4 x 4 mesh, and with the 10 x 10 mesh to decrease from 8.8 links per call to 8.6 on average. A breakdown of link usage with respect to group size for 10% and 20% multicast traffic is shown in Figures A.10 and A.12, respectively. Table B.3 summarizes the results of the regression analysis for the CBT model with a mean multicast group size of 8 members.

5.6 Hot Spot Analysis

This section compares the non-shared link routing models from the viewpoint of average number of calls on the network links (average link usage) for workloads with a mean multicast group size of 4. Corresponding link usage figures for workloads with a mean multicast group size of 8 appear in Appendix C. The goal was to identify the links which are the most heavily used (network hot spots) and see how the location and intensity of hot spots relates to the performance results seen in Sections 5.1 to 5.3.

5.6.1 N-Unicast Link Usage Mesh Plots

The link usage plots of the N-Unicast model showed symmetrical link usage patterns with the most heavily used links in the center of the mesh. The pattern of average number of calls per link in the mesh displayed some consistent trends. Figures 5.27 and 5.28 depict the link loads for 10% and 20% multicast traffic, respectively, for the N-Unicast model with a 4x4 mesh and a mean multicast group size of 4. For these and all other mesh diagrams, line thickness roughly corresponds to link usage. In the examples shown and for all sizes of mesh for the N-Unicast model, a “hot cross” phenomenon exists where the centermost two rows and two columns and their interconnecting links are consistently the most heavily used. The average link usage is heaviest in the 4 central links. Link usage gradually declines towards the edge of the mesh, and links nearest the corners of the mesh are the least used. For all routing model simulations, the maximum number of calls observed on any link was equal to the maximum capacity of that link, indicating that call failures due to blocking were possible anywhere in the mesh. Increasing the mean multicast group size to 8 in the N-Unicast model increased the average link usage only slightly, as shown in Figures C.1 and C.2 for 10% and 20% multicast traffic, respectively.

5.6.2 Conference Call Model Link Usage Mesh Plots

The patterns of average number of calls per link in the mesh were much more interesting in the Conference Call model than those of the N-Unicast model. Figures 5.29 and 5.30 depict the link loads for 10% and 20% multicast traffic, respectively, for the Conference Call model with a 4 x 4 mesh and a mean multicast group size of 4. In the examples shown, the busiest links were the ones connected to the conference hub node (denoted by an "X"). The overall pattern is much like that of the "hot cross" pattern of the N-Unicast model, except that average link usage is skewed towards the hub node. For a given workload, the average number of calls on the busiest links of the Conference Call models were significantly higher than on the busiest links of the N-Unicast model. This would explain why the call failure rates tend to be higher for the Conference Call model than for the N-Unicast model in many cases.

The presence of multiple regional hubs in larger size meshes using the Conference Call model affects the location of hot spots. An example of the average link usage map for a 6 x 6 mesh appears in Figure 5.31. In this case, there are two conference hubs. Average link usage is heaviest in a pattern about the two hubs, while link usage is lowest at the corners of the mesh. For 8 x 8 and 10 x 10 meshes, the pattern becomes less interesting, since there are four symmetrically placed hubs in these two mesh cases, and the pattern tends to smooth out.

As with the N-Unicast model, increasing the mean multicast group size increased the average link usage only by a small amount. Figures C.3 and C.4 depict the link loads for 10% and 20% multicast traffic, respectively, for the N-Unicast model for a 4x4 mesh with mean multicast group size of 8.

5.6.3 CBT Link Usage Mesh Plots

The link usage patterns of the CBT model resembled those of the N-Unicast model for all mesh sizes. This was not surprising, since the node responsible for dispatching multicast calls could be located anywhere in the mesh for both these models (but with the CBT model, cores located at or near the corners of the mesh would be rare). Figures 5.32 and 5.33 show the link usage diagrams for the CBT model with 10% and 20% multicast traffic, respectively, for a 4 x 4 mesh with a mean multicast group size of 4. While the central nodes for both the 10% and 20% multicast workloads had higher average link usage than the N-Unicast model, the nodes outside the center of the mesh had lower average link usage, and the savings increased with distance from the mesh center. This trend became particularly noticeable with large meshes, which could explain why the CBT model has the lowest link usage, particularly in large meshes. Larger meshes have a larger number of non-central links which give the large meshes using CBT routing greater link savings over the other two routing models.

As with the other two routing models, increasing the mean multicast group size increased the average link usage only by a small amount. Figures C.5 and C.6 depict the link loads for 10% and 20% multicast traffic, respectively, for the CBT routing model for a 4 x 4 mesh with mean multicast group size of 8.

5.6.4 Random CBT Link Usage Mesh Plots

The link usage diagrams of the previous three routing models served to suggest another possible routing heuristic: random placement of the core for the CBT model. This approach would be attractive for load balancing, since it should reduce

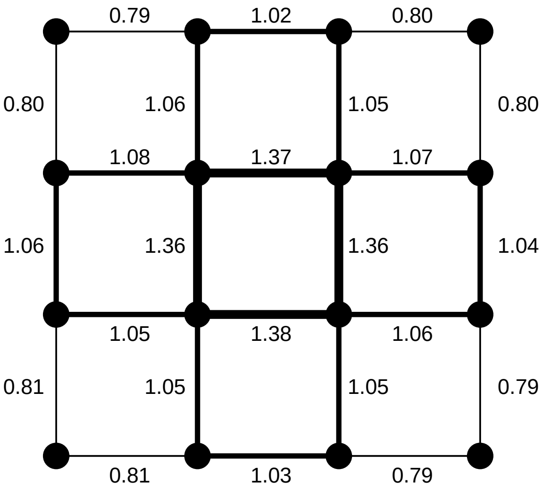

Figure 5.27: Avg. link usage, N-UNI model, 10% MC traffic, D = 4, g = 4.

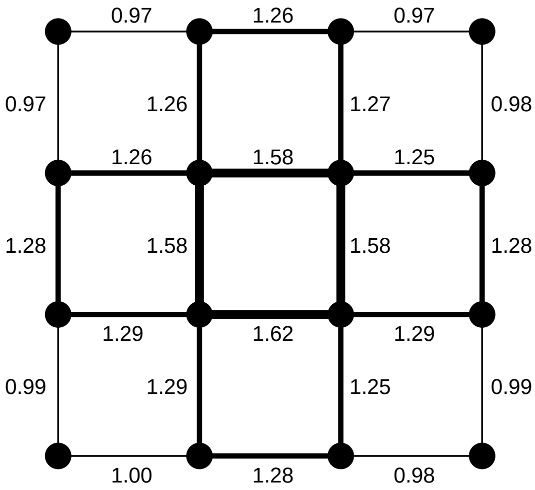

Figure 5.28: Avg. link usage, N-UNI model, 20% MC traffic, D = 4, g = 4.

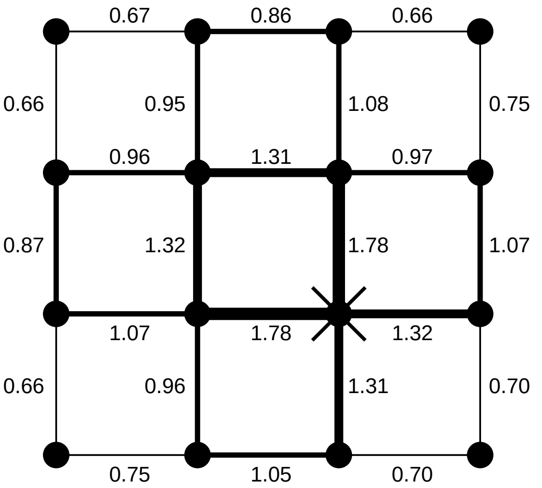

Figure 5.29: Avg. link usage, CC model, 10% MC traffic, D = 4, g = 4.

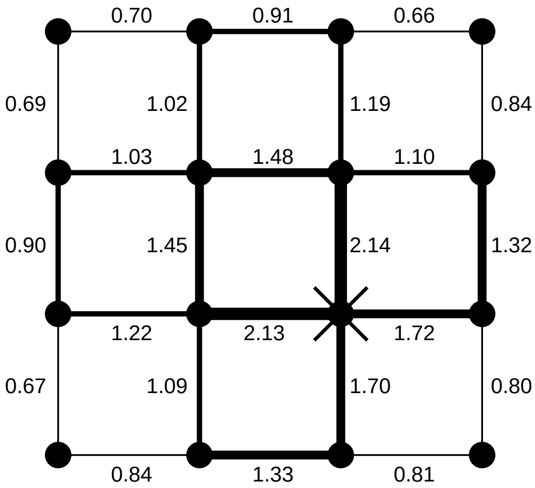

Figure 5.30: Avg. link usage, CC model, 20% MC traffic, D = 4, g = 4.

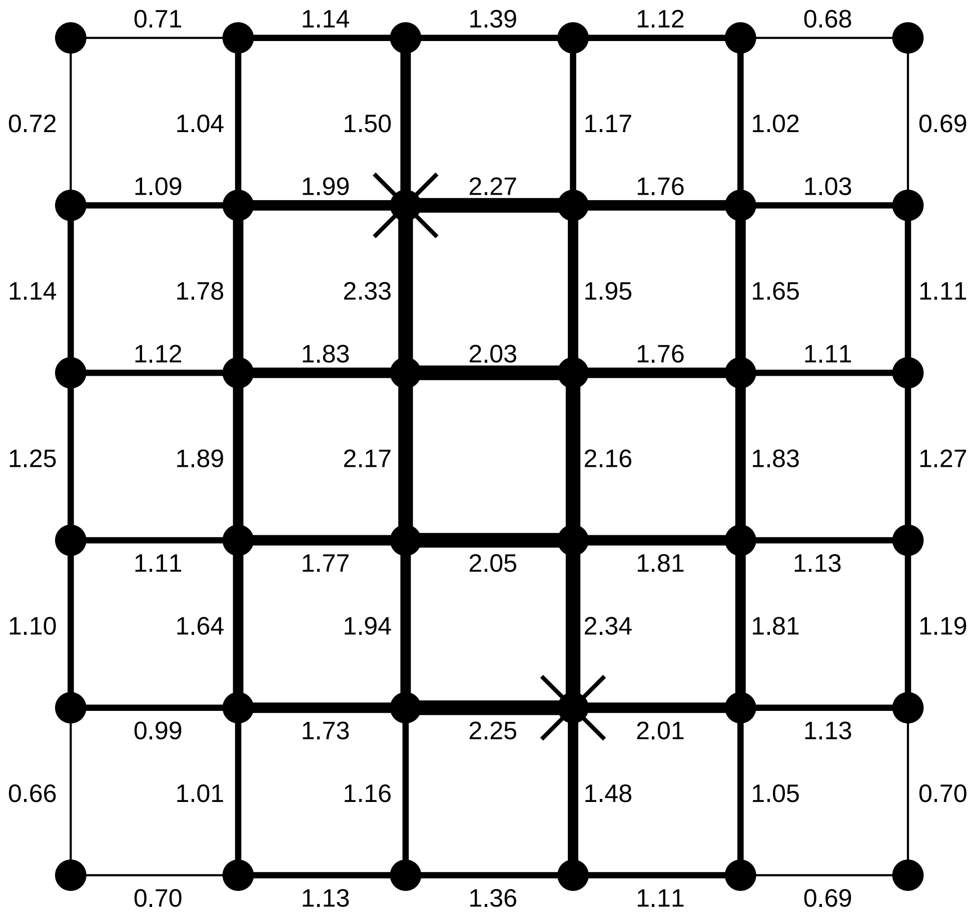

Figure 5.31: Avg. link usage, CC model, 10% MC traffic, D = 6, g = 4.

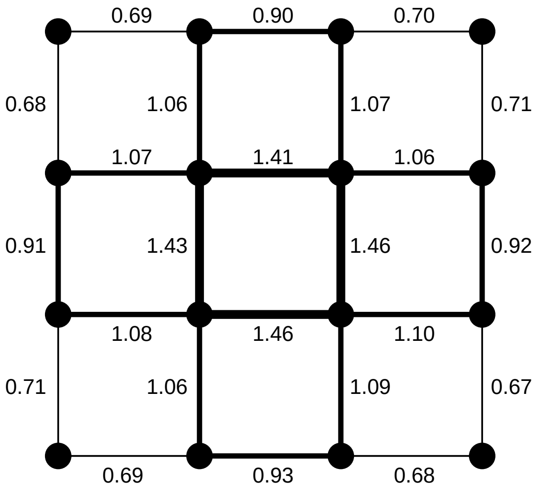

Figure 5.32: Avg. link usage, CBT model, 10% MC traffic, D = 4, g = 4.

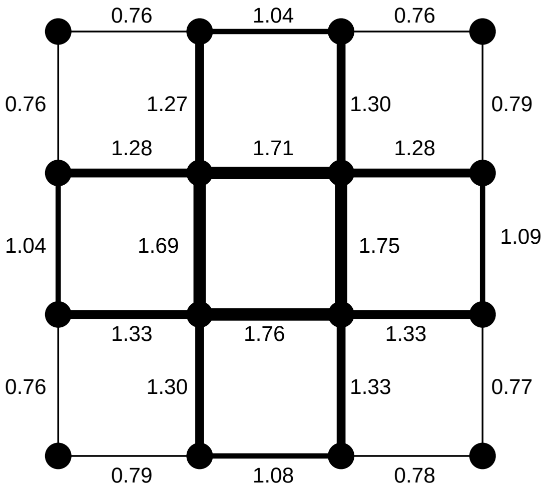

Figure 5.33: Avg. link usage, CBT model, 20% MC traffic, D = 4, g = 4.

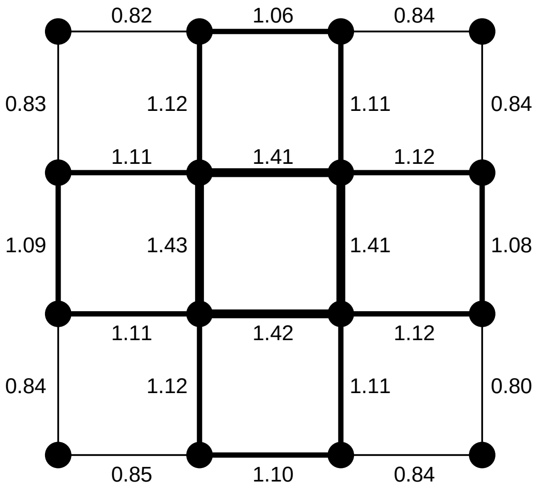

Figure 5.34: Avg. link usage, RCBT model, 10% MC traffic, D = 4, g = 4.

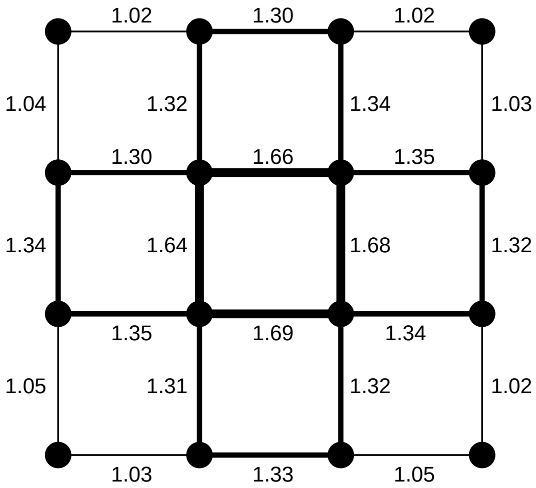

Figure 5.35: Avg. link usage, RCBT model, 20% MC traffic, D = 4, g = 4.

hot spots.

The link usage of the CBT routing model with random core placement was evaluated for a mean multicast group size of 4 to compare link usage with the three routing models previously discussed. The mesh plots for this model appear in Figure 5.34 for 10% multicast traffic and in Figure 5.35 for a 20% multicast traffic workload. Both mesh plots had average link usage profiles similar to the N-Unicast and CBT mesh link plots. However, the average link usage profile appeared to combine the high values of the busiest links of the CBT model with the high values of the corner links of the N-Unicast model. In other words, random core placement distributes slightly more call traffic to the edges and corners of the mesh than CBT does. With an average call blocking probability of 1.4% for 10% multicast traffic and 4.3% for 20% multicast traffic, this model had a higher rate of call failure than the other three routing models. This was not surprising, given the higher average link usage throughout the mesh, particularly along edges and at corner nodes. Thus random core placement does not appear to be a useful heuristic for multicast routing.

5.7 Summary

A summary table containing the average percentage of failed calls and average links used per call for the three non-shared link routing models appears in Table 5.7. The results are for a mean multicast group size of 4 recipients. A similar summary table for all test runs where the mean multicast group size was 8 recipients appears in Appendix D.

The experimental results revealed some trends which applied to all three non-shared link routing models:

- Large multicast groups had a much higher call blocking probability than small groups when all other factors were held constant.

- Call blocking increased with mesh size for the workloads used.

Table 5.7: Summary of Average Percent Failed Calls (PFC) and Average Links Used per Successful Call for the non-shared link routing models with 10% and 20% multicast traffic (mean multicast group size = 4).

Routing Algorithm		Mesh Size							
		D = 4		D = 6		D = 8		D = 10	
		Avg. PFC	Avg. Links	Avg. PFC	Avg. Links	Avg. PFC	Avg. Links	Avg. PFC	Avg. Links
N-Unicast	10%	1.0	3.5	1.5	5.4	2.6	7.2	4.8	9.1
	20%	2.9	4.4	4.6	6.5	7.3	8.7	11.4	10.8
Conf. Call	10%	1.1	3.5	2.0	5.3	3.1	6.9	5.4	8.9
	20%	4.0	4.2	6.2	6.2	8.2	8.0	11.8	10.2
CBT	10%	0.5	3.4	1.6	5.0	2.9	6.5	5.0	8.0
	20%	1.9	4.1	4.8	5.7	7.4	7.3	10.4	8.8

- Increasing the fraction of multicast traffic from 10% to 20% increased the probability of call blocking.
- The relationship between average number of links used per successful call and group size is strongly linear.
- All three routing models had average link per call usages within 15% of optimal for 4 x 4 meshes.
- Large meshes used more network links than small meshes for a given recipient group size.
- Increasing the fraction of multicast traffic caused the average number of links used per call to increase.
- Increasing the mean multicast group size in the workload increased call blockage.
- Increasing the mean multicast group size had only a small effect on average links used per call.

In addition, the results also revealed some relationships between the non-shared link routing models:

- The Conference Call model had higher average call blocking probability than the N-Unicast model.

- The Conference Call model had lower link usage than the N-Unicast model, with the difference being greatest on large mesh sizes.

- The average percentage of failed calls of the CBT model in relation to the N-Unicast and Conference Call models depended on mesh size and recipient group size.

- The CBT model consistently had the lowest average link usage of the three non-shared link routing models.

- N-Unicast and CBT models tended to have "hot cross" link usage patterns centered in the network mesh.

- The Conference Call model had particularly heavy link usage in the links connecting the hub nodes. This explains the higher average call blocking probability inherent with this routing model.

The results of the simulations indicated that the CBT routing model was the most promising non-shared link routing model. This routing model had the lowest average link usage and usually had the lowest average call blocking probability. The effect of link sharing on the performance of this algorithm, as well as on the Conference Call model, is explored in the next chapter.

CHAPTER 6

EXPERIMENTAL RESULTS: LINK SHARING

This chapter presents the simulation results for shared link multicast routing models. The two routing models investigated in this chapter are shared link versions of the Conference Call model and the Core Based Tree routing model. The effect of the link sharing heuristic will be easily discernible since it is the only modification made to the two routing models mentioned. By its definition, the N-Unicast algorithm does not permit link sharing. Thus it is not included in the experiments of this chapter.

The layout of this chapter is as follows. Section 6.1 presents the results for the shared link Conference Call model, while Section 6.2 discusses the shared link version of the Core Based Tree model. Section 6.3 focuses on the comparison of routing models. Section 6.4 evaluates the effect of multicast group size on these two multicast routing models, while Section 6.5 compares the two routing models from the standpoint of network hot spot analysis. Finally, Section 6.6 provides a summary of the findings of this chapter.

6.1 Conference Call Model with Link Sharing

The shared link Conference Call model uses statically located conference hubs to route multicast calls in the same way as the non-shared link version. The only difference between the two versions of this routing algorithm is that the shared link version attempts to route multicast calls using common link channels whenever possible. That is, if two or more recipients lie in the same general direction from the Conference Call hub, then a common path (requiring only one unit of channel capacity) is used to route toward those recipients. Routes diverge from this common path only as necessary to reach the individual recipients. The expectation of the link sharing heuristic was that link usage would be reduced and call blocking would decrease. The simulation results discussed in this section for the Conference Call model with link sharing are for a mean multicast group size of 4 recipients (g = 4).

For the 10% multicast traffic workload, the Conference Call model attained an average call blocking probability of 0.0%, on a 4 x 4 mesh. The average call blocking probabilities for the other mesh sizes were 0.2% for a 6 x 6 mesh, 0.8% for an 8 x 8 mesh, and 3.5% for a 10 x 10 grid. A breakdown of call blocking probability as a function of group size is shown in Figure 6.1.

The shared link model allows large multicast groups to be connected, since calls with more than 25 recipients are being successfully routed, as seen in Figure 6.1. For comparison, the largest successful group in the non-shared link Conference Call model with the same workload had only 18 recipients (Figure 5.6). This behaviour was reasonable, since link sharing clearly increases the maximum size of groups that can be routed (*i.e.,* the constraint $n \leq 4m$ no longer applies).

Link sharing also reduces call blocking. The average values of call blocking probability in Figure 6.1 are much lower than their counterparts in Figure 5.6 for the non-shared link version of this model, particularly for meshes of size 8 x 8 and smaller. However, the 10 x 10 mesh size did not realize as significant a reduction. It was also noted that the call blocking probability for unicast calls was 3.1%, compared with 2.4% for the non-shared link model. It appears that once a certain mesh size is reached (10 x 10 in this case), multicast calls are less likely to be blocked at the expense of increased unicast call failures. The call blocking probability still increased with mesh size, but for multicast calls the call blocking probability was always lower than that of the non-shared link Conference Call algorithm for a given workload and recipient group size.

As with the non-shared link routing models, the relationship between average link usage per call and multicast group size (refer to Figure 6.2) is linear. Average link usage per call increases with group size, and for a given multicast group size, the link usage increases with mesh size. For example, with the link sharing Conference Call model and a 10% multicast traffic workload, the 4 x 4 mesh used an average of 3.6 links per call, and the 6 x 6 mesh used 5.6 links per call. The 8 x 8 mesh required 7.6 links per call, and the 10 x 10 mesh needed 10.1 links per call on average. These values were larger than the values for the non-shared link Conference Call routing model with the same workload, with the difference becoming larger with increasing mesh size (particularly for the 10 x 10 mesh). The increase in link usage was surprising, since it was expected that the shared link model would reduce link usage in multicast calls. The explanation is that the increased number of multicast calls in the network load so many links to capacity that longer alternate routes are required to route new multicast calls.

With 20% multicast traffic, the Conference Call model with link sharing achieved an average call blocking probability of 0.2% on a 4 x 4 mesh, 1.0% for a 6 x 6 mesh, 3.4% for an 8 x 8 mesh, and 9.3% for a 10 x 10 mesh. As expected, these values were larger than for 10% multicast traffic. A breakdown of call blocking probability as a function of group size is shown in Figure 6.3. Again, call blocking probability increased with mesh size when all other factors were held constant, and it increased with recipient group size for a given workload. For all recipient group sizes, the shared link version of the Conference Call model had lower call blocking probabilities than the non-shared link version for the same offered workload to the simulation model.

The increase in the fraction of multicast traffic made the phenomenon of high unicast call blockage become more apparent. For the 20% multicast workload, 7.3% of unicast calls were blocked with the shared link Conference Call model, compared with only 3.6% call blockage for the same model without link sharing.

The increase in link usage for the shared link Conference Call model was also more apparent with the 20% multicast traffic workloads than it was with 10% multicast traffic. With 20% multicast traffic, the shared link Conference Call model averaged 4.5 links per call for a 4 x 4 mesh and 7.3 links for a 6 x 6 mesh (see Figure 6.4). The 8 x 8 sized version of this algorithm used 9.9 links per call on average, and 13.2 links per call for a 10 x 10 mesh. Again, the difference in link usage increased with group size. As with the case of 10% multicast workload, this link usage behaviour was unexpected. The relationship between average links used per call and multicast group size continued to remain linear. The link usage regression statistics for this model will be presented and discussed in detail in Section 6.3.

6.2 Core Based Tree Model with Link Sharing

The shared link Core Based Tree (CBT) model differed from its non-shared link counterpart only in that the former attempted to share common link channels in a multicast call when possible. As with the shared link Conference Call model, it was expected that the link sharing heuristic would reduce the call blocking probability and decrease link usage for the CBT routing model. The results discussed in this section are for the CBT model with link sharing and a mean multicast group size of 4 recipients.

For the 10% multicast traffic workload, the shared link Core Based Tree model attained an average call blocking probability of 0.0% on a 4 x 4 mesh. The average call blocking probability for the other mesh sizes were 0.1% for a 6 x 6 mesh, 0.8% for a mesh of size 8 x 8, and 2.9% for a 10 x 10 grid. These average values were lower than those for the shared link Conference Call model. As with all the other models, the call blocking probability increased with mesh size, and for a given workload, call blocking probability increased with the number of recipients in a group. A breakdown of call blocking probability as a function of group size appears in Figure 6.5. As with the shared link Conference Call model, larger group sizes were possible in the shared link Core Based Tree model compared with the non-shared link version.

In the shared link CBT model, unicast calls appeared to be blocked more frequently than in the non-shared link CBT model, but this effect was not as pronounced as in the shared link Conference Call model. The call blocking probability for unicast calls in the shared link CBT model was 2.3%, while in the non-shared link CBT model unicast calls were blocked only 1.2% of the time. However, the call blocking probability for multicast calls in the shared link CBT model were almost always less than those of the non-shared link CBT model. The only exceptions were the infrequent occurrences of very large group sizes, which had 100% call failure in both models.

Once again, the relationship between average link usage per call and multicast group size was linear (refer to Figure 6.6). The average link usage per call increased with group size, and for a given multicast group size, link usage increased with mesh size. With the shared link CBT model and a 10% multicast traffic workload, the 4 x 4 mesh used an average of 3.4 links per call, and the 6 x 6 mesh averaged 5.3 links per call. The 8 x 8 mesh required 7.2 links per call, and the 10 x 10 mesh needed 9.3 links per call on average. These values were equal to or larger than the values for the non-shared link CBT routing model with the same workload, with the difference becoming larger with increasing mesh size. Again, the increase in link usage was surprising, since it was expected that the shared link model would reduce link usage in multicast calls. However, the average link usage values were slightly lower than the corresponding values for the shared link Conference Call model, as expected. For example, with a 4 x 4 mesh and 10% multicast traffic, the shared link CBT model averaged 3.4 links per call compared with 3.6 links per call for the shared link Conference Call model. For a 10 x 10 mesh, the shared link CBT model required 9.3 links per call on average, compared with 10.1 links per call for the shared link Conference Call model.

For the 20% multicast traffic workload, the shared link CBT model attained an average call blocking probability of 0.0% on a 4 x 4 mesh. The average call blocking probability for the other mesh sizes were 0.7% for a 6 x 6 mesh, 2.9% for an 8 x 8 mesh, and 7.8% for a 10 x 10 mesh. A breakdown of call blocking probability as a function of group size is shown in Figure 6.7. As expected, these values were equal to or larger than those for the 10% multicast load. The relationship between call blocking probability and mesh size remained unchanged. In addition, the trend of call blocking being lower in the shared link CBT model than in the non-shared link CBT model reappeared in the 20% multicast traffic workload. The increase in call blocking with unicast calls remained. In the shared link CBT model, 5.4% of the unicast calls were blocked, while only 1.4% of the unicast calls were blocked in the non-shared link CBT model. Again, the shared link unicast call blockage probability was lower than that of the shared link Conference Call model.

The average links used per call for the shared link CBT model with 20% multicast traffic was 4.2 for a 4 x 4 mesh, 6.6 for a 6 x 6 mesh, 9.1 for an 8 x 8 mesh, and 11.8 for a 10 x 10 mesh (refer to Figure 6.8). With 20% multicast traffic workloads, the shared link CBT model continued to have lower average link usage than the shared link Conference Call model. The savings became more apparent with increasing mesh size. With a 4 x 4 mesh and 20% multicast traffic, the shared link CBT model averaged 4.2 links per call compared with 4.5 links per call for the shared link Conference Call model. For a 10 x 10 mesh, the shared link CBT model required 11.8 links per call on average, compared with 9.3 links per call for the shared link Conference Call model. The graph of average link usage per call as a function of group size for 20% multicast traffic appears in Figure 5.13. The linear regression statistics for this model's link usage are presented and discussed in Section 6.3.

6.3 Comparison of Models

The call blocking and link usage performance of the two shared link routing models were compared using confidence intervals to facilitate a more rigorous evaluation of the models. The details of these analyses are presented in this section.

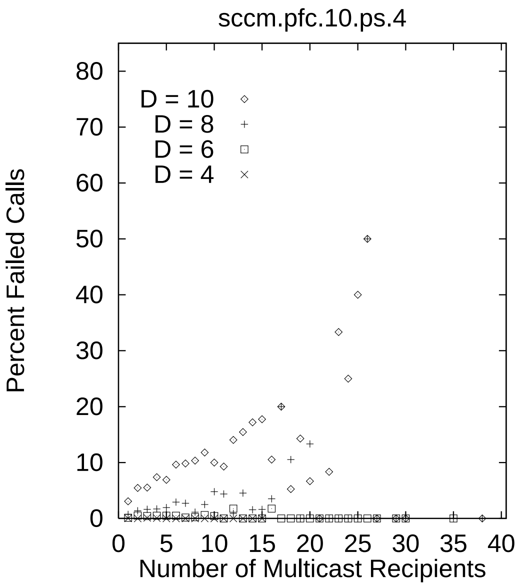

Figure 6.1: Percent failed calls, CC model, 10% MC traffic, g = 4.

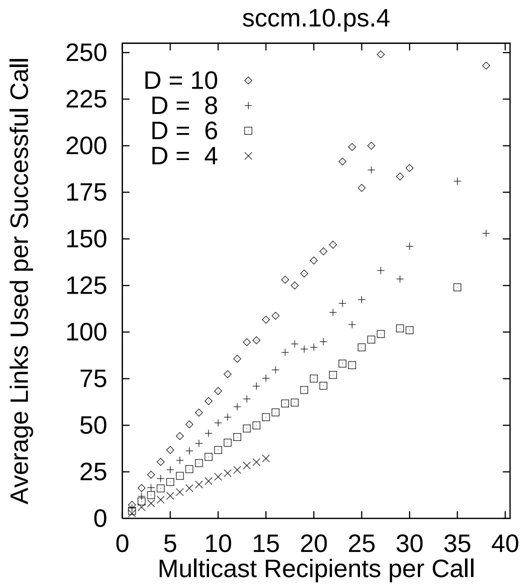

Figure 6.2: Avg. links per call, CC model, 10% MC traffic, g = 4.

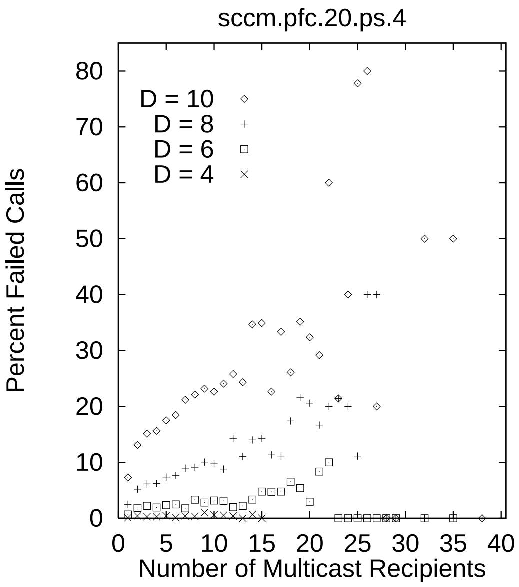

Figure 6.3: Percent failed calls, CC model, 20% MC traffic, g = 4.

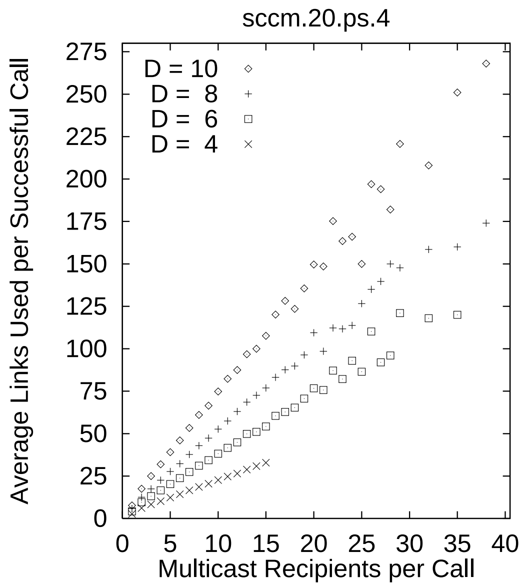

Figure 6.4: Avg. links per call, CC model, 20% MC traffic, g = 4.

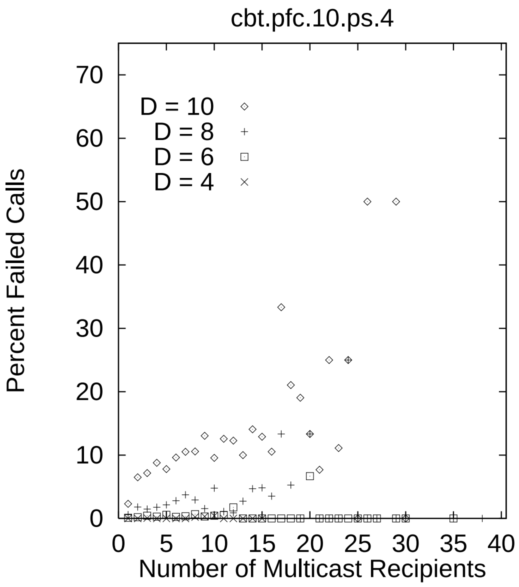

Figure 6.5: Percent failed calls, CBT model, 10% MC traffic, g = 4.

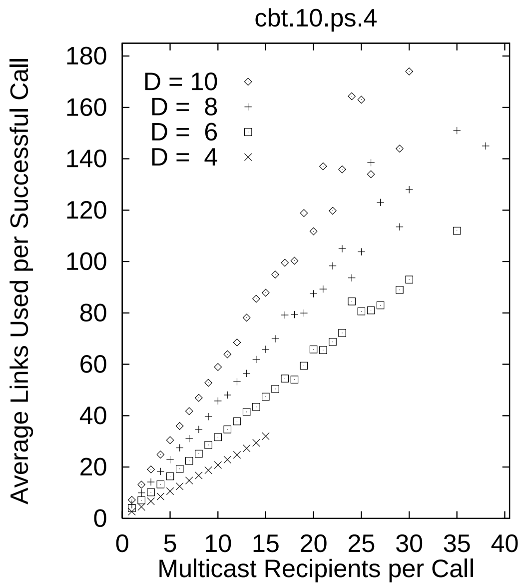

Figure 6.6: Avg. links per call, CBT model, 10% MC traffic, g = 4.

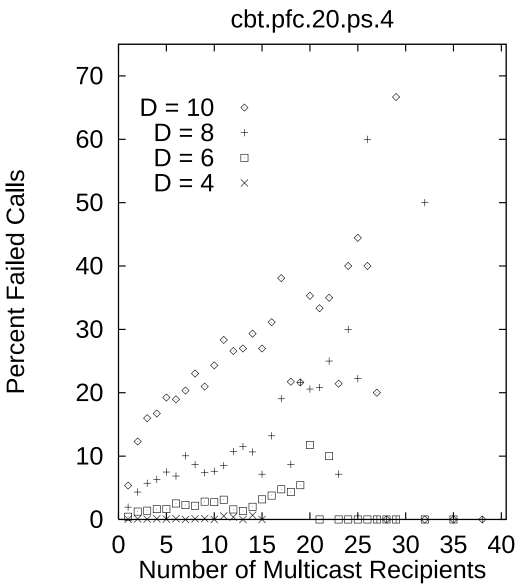

Figure 6.7: Percent failed calls, CBT model, 20% MC traffic, g = 4.

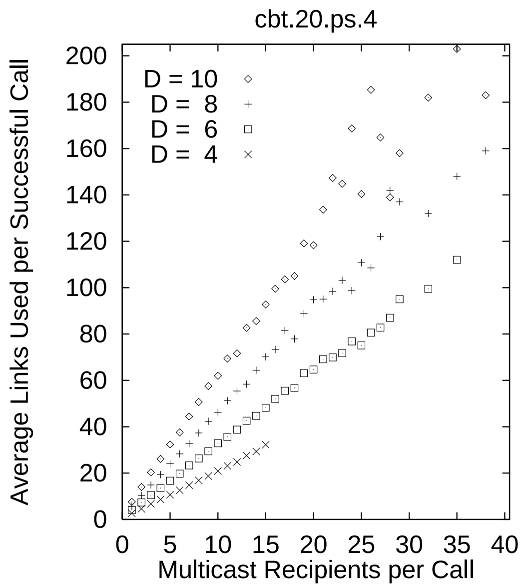

Figure 6.8: Avg. links per call, CBT model, 20% MC traffic, g = 4.

6.3.1 Call Blocking

As in Sections 5.1 and 5.4.1, a batch means analysis technique was employed to determine confidence intervals for call blocking results for each routing algorithm. For each shared link routing algorithm, the mean and standard deviation of call blocking probability were tabulated for unicast calls, multicast calls, and all calls. The batch means analyses were performed with 10 batches of 10,000 calls each, as in Sections 5.1 and 5.4.1. The comparison of shared link routing models according to unicast, multicast, and combined calls is covered in this subsection followed by more specific analysis of the individual routing algorithms.

The 95% confidence interval plots of unicast call blocking probability for all 4 mesh sizes appear in Figure 6.9. The workload consisted of 10% multicast traffic with a mean multicast group size of 4. The shared link CBT model conclusively had lower call blocking probability for unicast calls than the shared link Conference Call model for the D = 10 mesh size, but the results were inconclusive for other meshes due to confidence interval overlap. However, the mean values indicate that the shared link CBT model tends to have a lower call blocking probability for unicast calls than the shared link Conference Call model. The distinct confidence intervals for the D = 10 mesh size were consistent with the observation in Section 6.2 that the shared link CBT model blocked unicast calls less frequently than the shared link Conference Call model for large mesh sizes.

The 95% confidence interval comparison plots of call blocking probability for multicast calls appear in Figure 6.10. The confidence intervals for both routing models overlap for all mesh sizes. The only observation that can be made is that multicast call blocking performance between the shared link CBT and the shared link Conference Call model is very similar.

Since only 10% of the calls in these comparisons were multicast, it was not surprising to see that the 95% confidence level comparison plot for overall call blocking probability in Figure 6.11 resembled the unicast call blocking plot. As with the unicast plot, the D = 4 and D = 10 mesh sizes had non-overlapping confidence intervals with the shared link CBT model having a lower call blocking probability than the shared link Conference Call model. Once again, the mean call blocking probability values described a trend of the shared link CBT model having the lowest call blocking probability overall. The lower unicast call blocking probability of the shared link CBT model was chiefly responsible for the difference in overall call blocking performance between the two models.

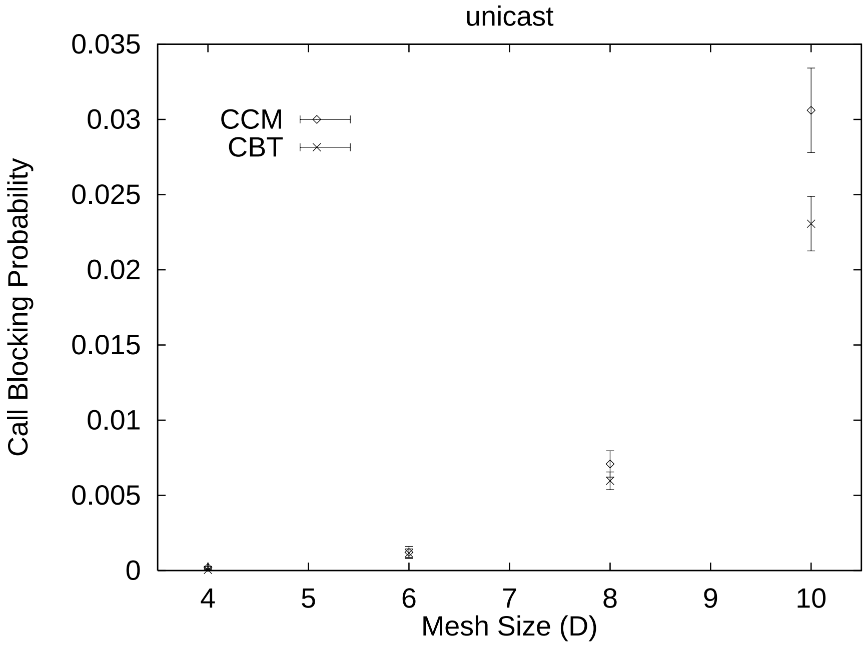

Figure 6.9: Unicast call blocking probability: observed values with 95% confidence intervals for the shared link Conference Call and CBT models (all mesh sizes with 10% multicast traffic, g = 4).

Conference Call Model

The call blocking results for all three categories of calls for the shared link Conference Call routing algorithm are summarized in Table 6.1. Unlike the case of the non-shared link routing models, the standard deviation in call blocking probability was not usually an order of magnitude smaller than the mean call blocking probability. Instead, the mean and standard deviation in call blocking probability was approximately the same order of magnitude for the small mesh sizes. As mesh size increased, the standard deviation in call blocking probability diminished until it

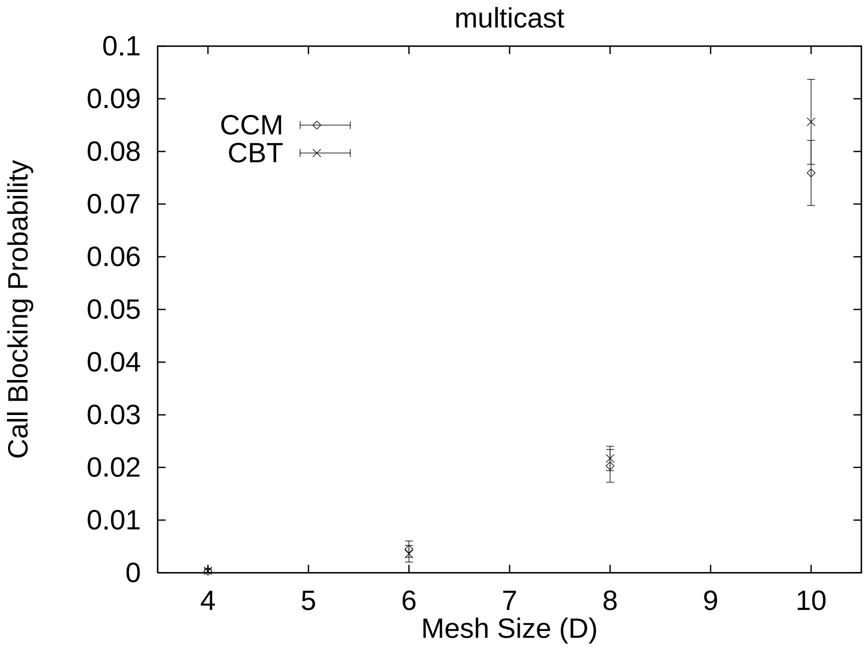

Figure 6.10: Multicast call blocking probability: observed values with 95% confidence intervals for the shared link Conference Call and CBT models (all mesh sizes with 10% multicast traffic, g = 4).

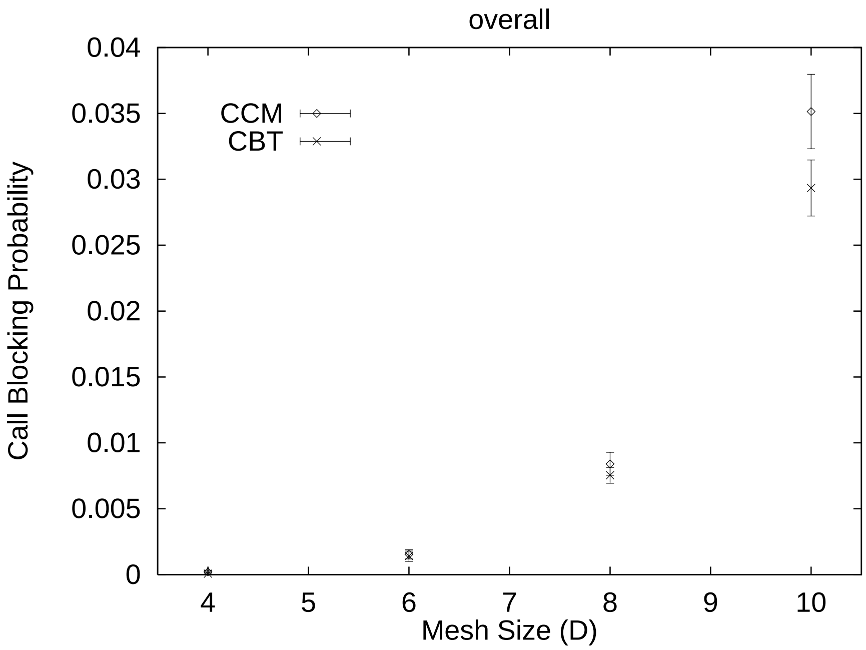

Figure 6.11: Overall call blocking probability: observed values with 95% confidence intervals for the shared link Conference Call and CBT models (all mesh sizes with 10% multicast traffic, g = 4).

was about an order of magnitude smaller than the mean call blocking probability for D = 10. However, this appears to have little effect on the confidence bounds, since for small mesh sizes the call blocking probability is so small (0.02% for D = 4 and 0.15% for D = 6) that the 95% confidence intervals would also be very small.

Table 6.1: Mean and Standard Deviation of the Call Blocking Probability for the Shared Link Conference Call routing model with 10% multicast traffic (mean multicast group size = 4).

Mesh Size	Call Blocking Probability for the Conf. Call Model					
	Unicast Calls		Multicast Calls		All Calls	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
4	0.0002	0.0001	0.0003	0.0005	0.0002	0.0001
6	0.0012	0.0006	0.0045	0.0025	0.0015	0.0006
8	0.0071	0.0014	0.0203	0.0050	0.0084	0.0014
10	0.0306	0.0045	0.0759	0.0100	0.0351	0.0046

CBT Model

A summary of the call blocking performance of the shared link CBT model appears in Table 6.2. As with the shared link Conference Call model, the standard deviation in call blocking probability was about the same order of magnitude as the mean call blocking probability for small mesh sizes. As mesh size increased, the standard deviation in call blocking probability decreased until it was approximately an order of magnitude smaller than its corresponding mean value in the largest mesh sizes.

6.3.2 Link Usage

As with the non-shared link routing models, comparisons of link usage per call were performed between shared link routing models. However, only comparisons between observed simulation results could be made. This was because the shared link routing heuristic did not easily lend itself to analysis of average links per call or determination of optimal link usage. Thus these two comparisons were not possible

Table 6.2: Mean and Standard Deviation of the Call Blocking Probability for the Shared Link CBT routing model with 10% multicast traffic (mean multicast group size = 4).

Mesh Size	Call Blocking Probability for the CBT Model					
	Unicast Calls		Multicast Calls		All Calls	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
4	0.0000	0.0001	0.0004	0.0007	0.0001	0.0001
6	0.0012	0.0005	0.0036	0.0025	0.0014	0.0006
8	0.0060	0.0010	0.0217	0.0037	0.0075	0.0010
10	0.0231	0.0029	0.0856	0.0130	0.0293	0.0034

for shared link routing models. As with the link usage comparisons between the non-shared link routing models, care had to be taken in comparing link usage between the shared link models due to differences in call blocking characteristics. The comparisons were considered valid for the shared link models for the same reasons as the non-shared link models (as outlined in Section 5.4.2).

Link Usage Comparison Between Routing Models

The 95% confidence intervals for the 4 x 4, 6 x 6, 8 x 8, and 10 x 10 meshes are shown in Figures 6.12, 6.13, 6.14, and 6.15, respectively, for the shared link Conference Call and CBT models. The workloads used had 10% multicast traffic and a mean multicast group size of 4. The mean value of links used per call for the shared link CBT model was consistently lower than the corresponding mean value of links used per call for the shared link Conference Call model. The confidence intervals remained relatively small up to group sizes of approximately 15 for all mesh sizes. For all mesh sizes, the confidence intervals between the routing algorithms did not overlap for group sizes between 2 and 13 recipients. This confirms that the shared link CBT requires the fewest average number of links to complete a multicast call. Similar plots for workloads with 20% multicast traffic (not shown) displayed similar behaviour.

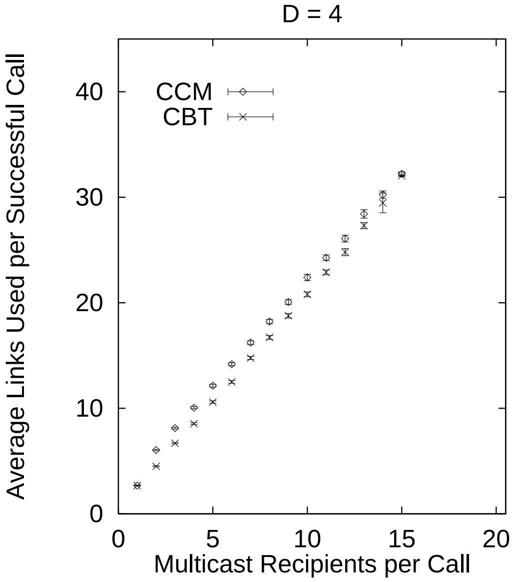

Figure 6.12: Avg. links per call 95% confidence intervals for D = 4, 10% MC traffic, g = 4.

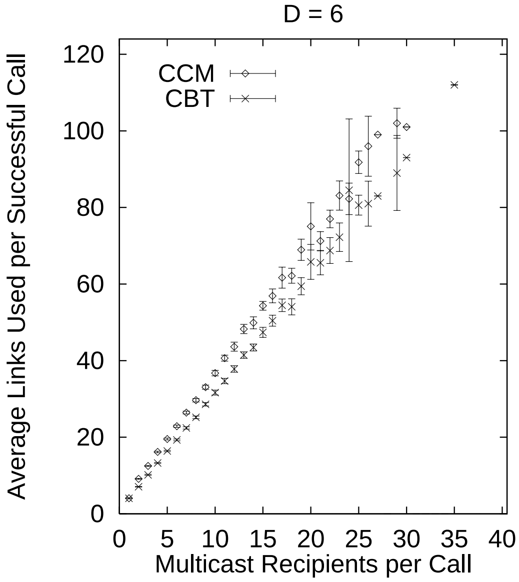

Figure 6.13: Avg. links per call 95% confidence intervals for D = 6, 10% MC traffic, g = 4.

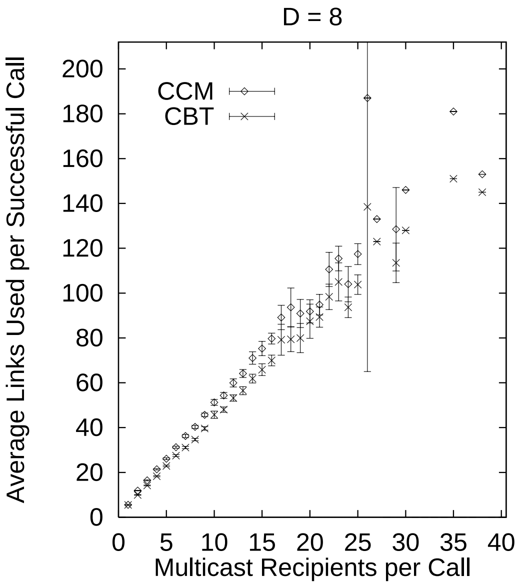

Figure 6.14: Avg. links per call 95% confidence intervals for D = 8, 10% MC traffic, g = 4.

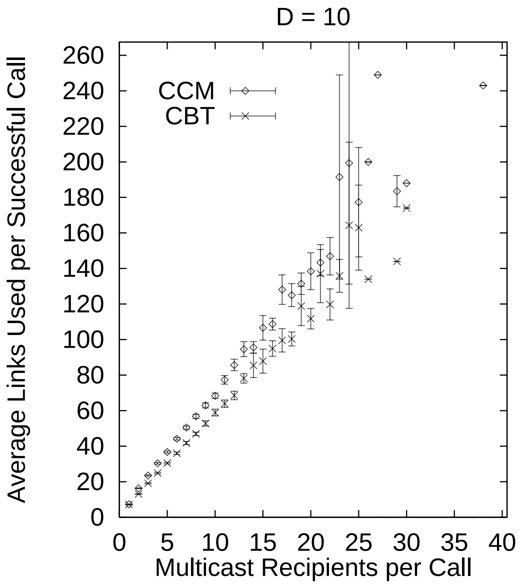

Figure 6.15: Avg. links per call 95% confidence intervals for D = 10, 10% MC traffic, g = 4.

Conference Call Link Usage

The plots of average links used per multicast call versus the number of multicast recipients per call were close to linear for both the 10% and 20% multicast traffic cases on all four mesh sizes for a mean multicast group size of 4 (see Figures 6.2 and 6.4).

Table 6.3 summarizes the regression analysis results for the shared link Conference Call model for workloads with a mean multicast group size of 4 members. Since all the R^2 regression coefficients were larger than 0.9, the link usages were deemed to fit very well with a linear model.

The slopes of the link usage curves for the shared link Conference Call model were slightly lower than the corresponding slopes for the non-shared link Conference Call model. The slope reductions (using workloads with 10% multicast traffic) were 8.9% for the 4 x 4 mesh, 7.5% for a 6 x 6 mesh, 7.1% on a 6 x 6 mesh, and -0.4% for a 10 x 10 mesh. The comparisons for the 20% multicast traffic workloads resulted in link usage savings of 7.9%, 5.0%, 7.0%, and 3.5% for D = 4, D = 6, D = 8, and D = 10, respectively. It appears that in small meshes, the link sharing heuristic successfully reduces the number of links used per call. In larger meshes, the opportunity to share link channels in a multicast call is reduced because the multicast recipients tend be more dispersed and do not share as many common links. As with the non-shared link Conference Call model, increasing the fraction of multicast traffic did not have a significant impact on the results.

CBT Link Usage

As expected, the relationship of average links used per call to recipient group size for the CBT model is linear (refer to Table 6.4). The R^2 regression coefficient values were greater than 0.9 for all mesh sizes and levels of multicast traffic.

The slopes of the link usage plots for the shared link CBT model from being more than 8% less than to nearly 6% more than the non-shared link CBT model. For workloads with 10% multicast traffic, the slope reductions due to the link sharing

Table 6.3: Regression Coefficients for Link Usage in the shared link Conference Call Model (mean multicast group size = 4).

Mesh Size	Percent MC Traffic	Linear Regression		
		Slope	y intercept	R^2
4	10	2.05	1.68	0.999
	20	2.09	1.70	0.999
6	10	3.47	1.96	0.996
	20	3.62	2.01	0.980
8	10	4.73	3.75	0.926
	20	4.76	5.01	0.990
10	10	7.02	2.59	0.947
	20	6.87	5.19	0.985

heuristic were 5.8% for the 4 x 4 mesh, 1.0% for the 6 x 6 mesh, 3.3% for the 8 x 8 mesh, and 4.4% for the 10 x 10 mesh. The results for workloads with 20% multicast traffic were 8.4%, 2.5%, -2.3%, and -5.8% for D = 4, D = 6, D = 8, and D = 10, respectively. Accurate comparisons are difficult to make since the y-intercept values for the regression analyses vary greatly, particularly for the large meshes. At best, it appears that the link sharing heuristic reduces link usage, at least for small meshes.

With the exception of the case of 10% multicast traffic for a 4 x 4 mesh, all slope values for the shared link CBT model were lower than their counterparts for the shared link Conference Call model, particularly for mesh sizes of D = 6 and larger. The slope values for the D = 4 mesh were nearly identical for both shared link routing algorithms.

6.4 Effect of Multicast Group Size

As in Section 5.4, a comparison of the effect of mean multicast group size on the performance metrics on the shared link models was conducted. This section discusses the effects of increasing the mean multicast group sizes of the workloads from 4 recipients to 8 recipients. All figures and tables referred to in this section

Table 6.4: Regression Coefficients for Link Usage in the shared link CBT Model (mean multicast group size = 4).

Mesh Size	Percent MC Traffic	Linear Regression		
		Slope	y intercept	R^2
4	10	2.07	0.27	0.999
	20	2.08	0.34	0.999
6	10	3.14	0.46	0.995
	20	3.11	1.47	0.998
8	10	4.16	3.17	0.974
	20	4.29	3.65	0.986
10	10	5.71	2.58	0.966
	20	5.52	8.37	0.954

are contained in Appendix E and Appendix F, respectively, due to their similarity to Figures 6.1 through 6.8.

6.4.1 Conference Call Model

The behaviour associated with increasing the mean multicast group size from 4 to 8 in the shared link Conference Call model differed slightly from the case of the non-shared link version of this model. Increasing the mean multicast group size from 4 to 8 usually resulted in increased call blockage, but the relative increase was greatest for the large mesh sizes instead of for the small mesh sizes (the case in the non-shared link Conference Call model). With the 10% multicast traffic workloads, increasing the mean multicast group size from 4 to 8 resulted in no measurable change in the percentage of failed calls (both 0.0%) for a 4 x 4 mesh, while for the 10 x 10 mesh, the call blocking probability increased from 3.5% to 6.3%. For the 20% multicast traffic workloads, the corresponding increase for the 4 x 4 mesh was from 0.2% to 0.3%. For a 10 x 10 mesh, increasing the mean multicast group size from 4 to 8 resulted in an increase in the call blocking probability from 9.3% to 14.5%. A breakdown of call blocking probability as a function of group size is shown in Figure E.1 for 10% multicast workloads and in Figure E.3 for the 20% multicast traffic workloads.

Unlike the non-shared link Conference Call model, the average number of links used per call was affected by the mean multicast group size in the shared link Conference Call model. The differences increased with increasing group size. For a 4 x 4 mesh with the 10% multicast traffic workload, the average links used per call for a mean multicast group size of 4 was 3.6 links per call, but for a workload with a mean multicast group size of 8, 3.9 links were needed per call on average. For a 10 x 10 mesh with 10% multicast calls in the workload, increasing the mean multicast group size from 4 to 8 resulted in an increase in link usage from 10.1 to 12.4 links per call. With 20% multicast traffic workloads, increasing the mean multicast group size from 4 to 8 resulted in an increase in link usage from 4.5 to 5.2 links per call in the 4 x 4 mesh, and an increase from 13.2 to 17.1 links per call in the 10 x 10 mesh. A breakdown of link usage with respect to group size for 10% multicast traffic is shown in Figure E.2 and for 20% multicast traffic in Figure E.4. Table F.1 summarizes the results of the regression analysis for the Conference Call model with a mean multicast group size of 8 members.

6.4.2 Core Based Tree Model

The shared link Core Based Tree model followed the behaviour of the shared link Conference Call model when the mean multicast group size was raised from 4 to 8. Call blockage increased with both 10% and 20% multicast workloads with the largest increase occurring in the large mesh sizes. With 10% multicast traffic, the percent failed calls in the 4 x 4 mesh remained unchanged at 0.0% when the mean multicast group size was raised from 4 to 8. In a 10 x 10 mesh with the same amount of multicast traffic, the call blocking probability increased from 2.9% to 5.4%. For 20% multicast traffic, increasing the mean multicast group size increased the percentage of failed calls in the 4 x 4 mesh from 0.0% to 0.2%, and from 11.8 % to 15.1% in the 10 x 10 mesh. For a more detailed breakdown of call blocking probability as a function of group size, refer to Figure E.5 for 10% multicast traffic workloads and to Figure E.7 for the 20% multicast traffic workloads.

As with the shared link Conference Call model, the average links used per call increased with mean multicast group size, with the differences becoming larger with increasing mesh size. For the 10% multicast traffic workloads, increasing the mean multicast group size from 4 to 8 increased the average links used per call from 3.4 to 3.8 for a 4 x 4 mesh, and link usage increased from 9.3 to 11.3 links per call in a 10 x 10 mesh. With 20% multicast traffic, increasing the group size caused the average link usage to increase from 4.2 links per call to 4.9 for the 4 x 4 mesh, and with the 10 x 10 mesh from 11.8 links per call to 15.1 on average. A breakdown of link usage with respect to group size for 10% and 20% multicast traffic is shown in Figures E.6 and E.8, respectively. Table F.2 summarizes the results of the regression analysis for the CBT model with a mean multicast group size of 8 members.

6.5 Hot Spot Analysis

This section compares the shared link models on the basis of average number of calls per network links (average link usage) for workloads with a mean multicast group size of 4. Corresponding link usage figures for workloads with a mean multicast group size of 8 appear in Appendix G. As with Section 5.5 for the non-shared link models, the goal was to relate average link usage (*i.e.,* hot spots) to the performance results for the shared link models described in previous sections.

6.5.1 Conference Call Model Link Usage Mesh Plots

The patterns of average number of calls per link in the mesh for the shared link Conference Call model resembled those of the non-shared link model. Figures 6.16 and 6.17 depict the link loads for 10% and 20% multicast traffic, respectively, for the Conference Call model with a 4 x 4 mesh and a mean multicast group size of 4. The busiest links were the ones adjacent to the conference hub node (denoted by an “X”). However, the shared link model had significantly lower average link usage than the non-shared link version in the busiest links (compare Figures 6.16 and 5.29).

Increasing the mean multicast group size increased the average link usage only by a small amount. Figures G.1 and G.2 depict the link loads for 10% and 20% multicast traffic, respectively, for the CC routing model for a 4 x 4 mesh with mean multicast group size of 8.

6.5.2 CBT Link Usage Mesh Plots

The link usage patterns of the shared link CBT model resembled those of the non-shared link CBT model for all mesh sizes. Figures 6.18 and 6.19 show the link usage diagrams for the CBT model with 10% and 20% multicast traffic, respectively, for a 4 x 4 mesh with a mean multicast group size of 4. The busiest links in the CBT model had a lower average link usage than the busiest links in the Conference Call model.

As with the shared link Conference Call model, increasing the mean multicast group size increased the average link usage only slightly. Figures G.3 and G.4 illustrate the link loads for 10% and 20% multicast traffic, respectively, for the CBT routing model for a 4 x 4 mesh with mean multicast group size of 8.

6.6 Summary of Results

A table containing a summary of the average percentage of failed calls and average links used per call for both shared link routing models, where the mean multicast group size was 4 recipients, appears in Table 6.5. A similar summary table for all runs where the mean multicast group size was 8 recipients appears in Appendix H.

The findings that were observed for all routing models in the non-shared link simulations were also observed for the shared link routing models. In addition to those observations (listed in Section 5.6), the experimental results for the shared link models revealed the following:

- The shared link models resulted in fewer call failures, as expected.

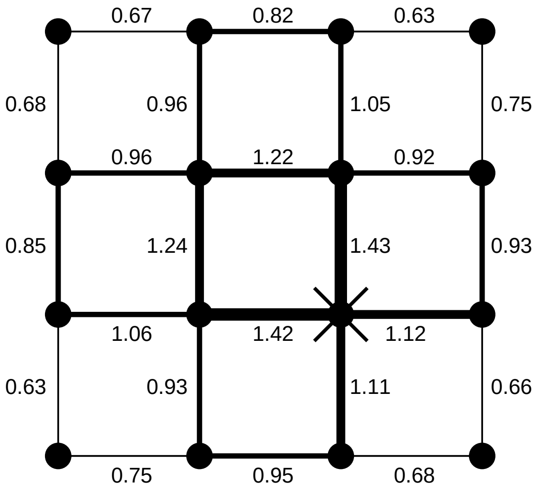

Figure 6.16: Avg. link usage, CC model, 10% MC traffic, D = 4, g = 4.

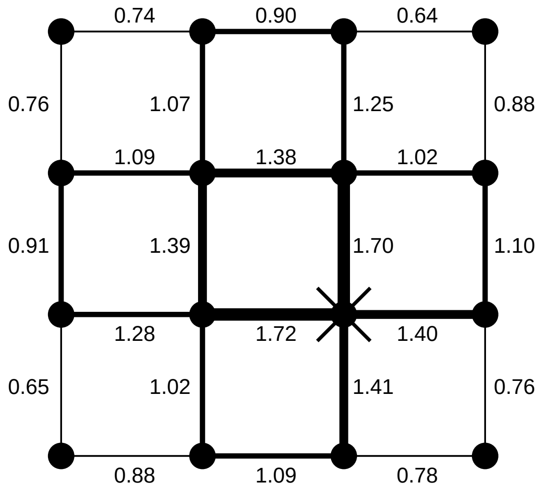

Figure 6.17: Avg. link usage, CC model, 20% MC traffic, D = 4, g = 4.

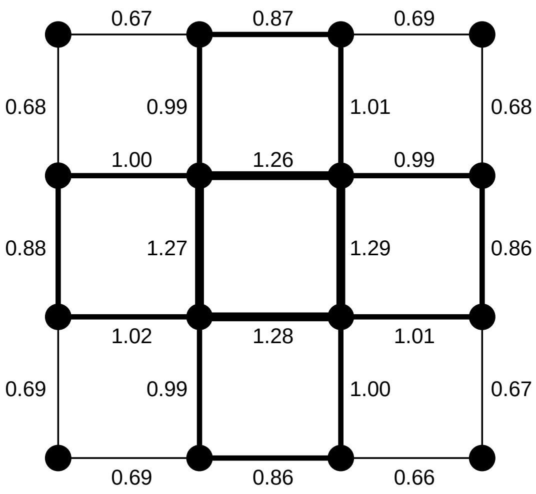

Figure 6.18: Avg. link usage, CBT model, 10% MC traffic, D = 4, g = 4.

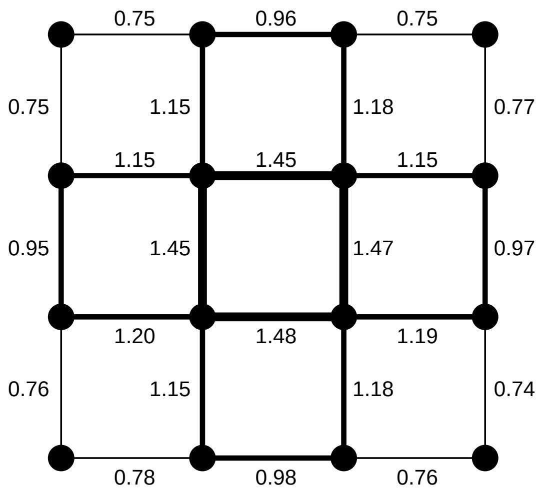

Figure 6.19: Avg. link usage, CBT model, 20% MC traffic, D = 4, g = 4.

Table 6.5: Summary of Average Percent Failed Calls (PFC) and Average Links Used per Successful Call for the shared link routing models with 10% and 20% multicast traffic (mean multicast group size = 4).

Routing Algorithm		Mesh Size							
		D = 4		D = 6		D = 8		D = 10	
		Avg. PFC	Avg. Links	Avg. PFC	Avg. Links	Avg. PFC	Avg. Links	Avg. PFC	Avg. Links
Conf. Call	10%	0.0	3.6	0.2	5.6	0.8	7.6	3.5	10.1
Link Shared	20%	0.2	4.5	1.0	7.3	3.4	9.9	9.3	13.2
CBT	10%	0.0	3.4	0.1	5.3	0.8	7.2	2.9	9.3
Link Shared	20%	0.0	4.2	0.7	6.6	2.9	9.1	7.8	11.8

- The shared link models allowed larger multicast group sizes than their non-shared link counterparts, but this was at the expense of more links being required to route these calls.

- The shared link models were not as biased against large recipient groups or mesh sizes as the non-shared link models.

- The link sharing heuristic achieved modest reductions in the number of links used to route a call for small meshes.

- The shared link models blocked unicast calls more frequently than did the non-shared link models for mesh sizes larger than 6 x 6. That is, the reduced blocking of multicast calls in the shared link models came at the expense of unicast calls in large meshes.

- The best overall multicast routing model in the shared link category is the Core Based Tree model. Its dynamic core placement allowed it to have the lowest average link usage per call, and this model had the lowest overall call blocking (primarily due to lower unicast call blocking).

CHAPTER 7

SUMMARY AND CONCLUSIONS

This chapter presents the main results of this research and the conclusions. Section 7.1 restates the objectives of this work, while Section 7.2 summarizes the experimental method used. Section 7.3 summarizes the results of the research, and Section 7.4 presents the conclusions. Finally, Section 7.5 outlines some areas for future research.

7.1 Objectives

The objective of this research was to evaluate the performance of multicast routing algorithms for an ATM network environment. In particular, the goal was to determine the performance differences between multicast routing algorithms, and to determine under what workload conditions these performance differences became significant. Investigation of the scalability of these algorithms was also a research objective, as was determining what heuristics were useful in improving the efficiency of multicast routing algorithms.

7.2 Experimental Method

Call level simulation was used to evaluate multicast routing algorithms on simple mesh based networks. A set of synthetic workload files were created to simulate an offered load of calls to the network, with a separate set of workloads for each size of mesh. Each workload was run on all of the routing algorithms to evaluate the differences in performance between them.

Two groups of routing algorithms were investigated. The first group was the non-shared link routing algorithms which routed multicast calls from point to multipoint without regard for sharing common link channels in completing a call. The second group, the shared link routing algorithms, attempted to share common link channels in a multicast call whenever possible. The former group included the N-Unicast, Conference Call, and Core Based Tree models. The latter group consisted of shared link versions of the Conference Call and Core Based Tree models. Direct comparison between these two model groups was possible with the Conference Call and Core Based Tree models.

The simulation experiments used a full factorial experimental design. The four experimental factors in the experiments were routing algorithm, mesh size, fraction of multicast traffic in the workload, and mean multicast group size of the workload.

7.3 Summary of Results

Several general relationships between workload parameters and their impact on call blocking and link usage were established for the factors and levels tested:

- There was a bias against large multicast groups and large mesh sizes. Large multicast groups had a greater probability of call blocking than smaller groups when all other factors were held constant. For the workloads used in the experiments, it was found that call blocking increased with network mesh size since workload was scaled with mesh size. Also, increasing the fraction of multicast traffic in the workloads offered to the network models increased call blocking probability.

- The relationship between average links used per call and the multicast group size was consistently a linear relationship for all multicast routing models.

The slope varied with the mesh size and routing algorithm.

- In the non-shared link models, increasing the mean multicast group size had little effect on the average links used per call, but increased call blocking. In the shared link models, increasing the mean multicast group size had little effect on call blocking, but caused link usage to increase noticeably, particularly in large mesh sizes with heavy multicast workloads. This was because the reduced call blocking probability of the shared link models increased the number and size of calls in the network, which resulted in longer average paths being required to connect incoming calls.

- Network hot spot analysis for non-shared link models analysis revealed that in general, links adjacent to the corner nodes of a network have the lowest average calls per link. In the Conference Call model, links adjacent to the conference hubs are the busiest, while in the N-Unicast and Core Based Tree models the busiest links tend to be centered in the mesh in a "hot cross" pattern. The Conference Call model had the "hottest" hot spots among the three non-shared link models for a given workload.

- Shared link routing models scaled better (*i.e.,* had lower call blocking probabilities) with respect to recipient group size and mesh size than non-shared link routing models.

- The shared link models resulted in fewer call failures than the non-shared link routing models, which was the expected behaviour. For large mesh sizes the shared link models had approximately double the unicast call blocking probability of their non-shared link counterparts. However, the call blocking probability of the multicast calls was so much lower for the shared link models that they had a lower overall call blocking probability than the non-shared link routing models.

- The hot spot patterns in shared link models closely resembled the patterns in their non-shared link counterparts. However, the busiest links were significantly less busy in the shared link models.

In the non-shared link category, the CBT model showed the most promise because it had the lowest average link usage and it usually had the lowest average call blocking probability. The Conference Call model had higher call blocking probability than the N-Unicast model, but used fewer links per call on average, particularly with large mesh sizes. The N-Unicast model's advantage over the Conference Call model was that the N-Unicast did not suffer from the hot spot problems that the Conference Call model had. The busiest links in the N-Unicast model were less busy than the busiest links in the Conference Call model. The N-Unicast model had no advantage over the CBT model, apart from simplicity.

The best overall multicast routing model in both the non-shared link and shared link category is the Core Based Tree model. Its dynamic core placement allowed it to have the lowest average link usage per call, and this model usually had the lowest call blocking.

7.4 Conclusions

Three main conclusions can be drawn from this research:

First, serious consideration must be given to the design of multicast routing algorithms for future networks. The experiments reported in this thesis have shown that simplistic approaches to multicast call routing, such as N-Unicast or the Conference Call model, do not scale well with network size, with multicast load, or with mean multicast group size. More sophisticated routing models, such as Core Based Trees, offer much more potential for scalability in multicast routing. In both the shared link and non-shared link categories, the Core Based Tree model had the lowest average link usage per call, and it usually had the lowest call blocking probability.

Second, among the routing heuristics considered for multicast traffic, the link sharing heuristic is the most important. Experiments have shown that dynamic core placement (or even random core placement) in the CBT model can somewhat alleviate the hot spots inherent in the Conference Call and N-Unicast models, but it is only with the addition of the link sharing feature that call blocking is dramatically reduced. Link sharing reduces (but does not totally eliminate) the bias against large multicast groups, and permits the routing of larger multicast calls than is possible in the non-shared link routing models.

Third, some consideration must be given to service provisioning when multicast traffic is introduced in a network, since the presence of multicast calls can affect the performance seen by unicast calls. For example, multicast calls require more network links than unicast calls, and link requirements increase linearly with the group size. A multicast call pricing schedule that is directly proportional to recipient group size may be one way for a network to operate profitably. Indeed, in several experiments, the overall reduction in call blocking observed for shared link models came from reduced multicast call blocking, at the expense of increased unicast call blocking. For the network manager, balancing this tradeoff becomes an important issue. Under some circumstances it may be better to block a large multicast call to allow several smaller multicast or unicast calls, and vice versa.

7.5 Future Work

While this thesis provides a good starting point for the performance evaluation of multicast routing algorithms, there are still several promising areas for future research. These areas include the design of the network model, the simulation workloads, and dynamic multicast groups.

7.5.1 Network Modelling

If further research is to more closely simulate realistic ATM networks, then more complex network topologies need to be simulated. This would include networks based on irregular topologies, such as those formed by linking together cities in a geographical area. Hierarchical networks of ATM switches also need to be considered.

Deciding on a "typical" topology to model is the most difficult part of this problem. Clearly, a realistic ATM network model must include geographical distances, heterogeneous link speeds, and arbitrary forms of network interconnection. The difficulty of choosing a representative network topology was the main reason why this thesis research considered only mesh based networks. While this modelling assumption made the simulation easier to implement, and the simulation results easier to understand, extensions to support more general network topologies are obviously required.

7.5.2 Workload Modelling

The simulation results in this thesis make several simplifying assumptions about the multicast traffic workload. For example, the model assumes that call arrivals are Poisson, call durations are exponentially distributed, and multicast group sizes follow a geometric distribution. Furthermore, the simulations in this thesis assume that all calls require exactly one unit of link capacity (homogeneous traffic), which is unrealistic.

More accurate workload models for multicast traffic are required. Accurate workload synthesis, however, would require characterization from a real multicast ATM network with numerous switches, and a more uniform implementation of ATM multicast than is currently the case in practice. In the interim, workload characterization from the Internet MBONE may suffice.

7.5.3 Dynamic Multicast Groups

The experiments conducted in this research assume static multicast groups. That is, the members of the multicast group are known ahead of time, before the call is routed, and members remain part of the multicast group for the complete duration of the call. In practice, this is unrealistic, since members may be dropped and reconnected due to network failures, or because recipients may voluntarily join or leave the group during a multicast call.

Dynamic group membership adds even further complexity to the already difficult multicast routing problem, since "good" multicast routes may become arbitrarily poor routes when group membership changes. Future work in ATM multicast (whether theoretical, simulation, or experimental) should be able to allow for these types of connections and be able to quantify their effects.

REFERENCES

[1] H. Ahmadi. A Survey of Modern High-Performance Switching Techniques. *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 7, pp. 1091-1103, September 1989.

[2] T. Ballardie, P. Francis, and J. Crowcroft. Core Based Trees (CBT). *Proceedings of ACM SIGCOMM '93*, pp. 85-94, September 1993.

[3] R. Ballart and Y. Ching. SONET: Now It's the Standard Optical Network. *IEEE Communications Magazine*, vol. 27, no. 3, pp. 8-15, March 1989.

[4] Bellcore. *Telecommunications Transmission Engineering, Volume 3: Networks and Services.* Bellcore Technical Publications, 1990.

[5] S. Casner. Frequently Asked Questions (FAQ) on the Multicast Backbone (MBONE). Available by anonymous FTP from venera.isi.edu in directory mbone/faq.txt, May 1993.

[6] S. Casner and S. Deering. First IETF Internet Audiocast. *ACM Computer Communications Review*, vol. 22, no. 3, pp. 92-97, July 1992.

[7] C. Catlett. In Search of Gigabit Applications. *IEEE Communications Magazine*, vol. 30, no. 4, pp. 42-51, April 1992.

[8] W. Currie. *LANs Explained: A Guide to Local Area Networks.* Ellis Horwood Limited, Chichester, England, 1988.

[9] L. Cuthbert and J. Sapanel. *ATM: The Broadband Telecommunications Solution.* The Institution of Electrical Engineers, London, 1993.

[10] S. Deering. Multicast Routing in Internetworks and Extended LANS. *Proceedings of ACM SIGCOMM '88*, pp. 55-64, August 1988.

[11] S. Deering. Host Extensions for IP Multicasting. Technical Report, Stanford University, August 1989. RFC 1112.

[12] S. Deering. *Multicast Routing in a Datagram Internetwork*. PhD thesis, Stanford University, 1991.

[13] S. Deering and D. Cheriton. Multicast Routing in Datagram Internetworks and Extended LANS. *ACM Transactions on Computer Systems*, vol. 8, no. 2, pp. 85-110, May 1990.

[14] M. Doar. *Multicast in the Asynchronous Transfer Mode Environment*. PhD thesis, University of Cambridge, 1993.

[15] ATM Forum. *ATM User-Network Interface Specification Version 3.0*. Prentice Hall Incorporated, Englewood Cliffs, NJ, 1993.

[16] A. Frank, L. Wittie, and A. Bernstein. Multicast Communication on Network Computers. *IEEE Software*, vol. 2, no. 3, pp. 49-61, May 1985.

[17] F. Hwang and D. Richards. Steiner Tree Problems. Technical Report, Bell Laboratories, *c.*1987.

[18] R. Jain. *The Art of Computer Systems Performance Analysis*. John Wiley & Sons, Incorporated, New York, 1991.

[19] M. Laubach. Classical IP and ARP Over ATM. Technical Report, Hewlett-Packard Laboratories, January 1994. RFC 1577.

[20] J. Le Boudec. The Asynchronous Transfer Mode: a Tutorial. *Computer Networks and ISDN Systems*, vol. 24, no. 4, pp. 279-309, May 1992.

[21] D. Lynch and M. Rose. *Internet System Handbook*. Addison-Wesley Publishing Company, Incorporated, Reading, MA, 1993.

[22] R. Melen. Current Architectures for ATM Implementation. *European Transactions on Telecommunications and Related Technologies*, vol. 3, no. 2, pp. 145-155, March-April 1992.

[23] S. Minzer. Broadband ISDN and Asynchronous Transfer Mode (ATM). *IEEE Communications Magazine*, vol. 27, no. 9, pp. 17-24, September 1989.

[24] R. Ruedisueli. Audio Teleconferencing Transmission Systems & Considerations. Technical Report, A.T.&T Co. Long Lines, Washington, D.C., *c.*1977.

[25] C. Symborski. cell-relay-faq. Available by anonymous FTP from cell-relay.indiana.edu in directory /pub/cell-relay, November 1993.

[26] A. Tanenbaum. *Computer Networks, 2nd ed.* Prentice Hall Incorporated, Englewood Cliffs, NJ, 1989.

[27] F. Tobagi. Fast Packet Switch Architectures for Broadband Integrated Services Digital Networks. *Proceedings of the IEEE*, vol. 78, no. 1, pp. 33-167, January 1990.

[28] J. Turner. Optimal Nonblocking Multipoint Virtual Circuit Switching. Technical Report WUCS-93-47, Computer Science Department, Washington University, St. Louis, September 1993.

[29] D. Waitzman, C. Partridge, and S. Deering. Distance Vector Multicast Routing Protocol. Technical Report, BBN, STC, and Stanford University, November 1988. RFC 1075.

[30] D. Wall and S. Owicki. Construction of Centered Shortest-Path Trees in Networks. *Networks*, vol. 13, no. 2, pp. 207-231, Summer 1983.

[31] B. Waxman. Routing of Multipoint Connections. *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 9, December 1988.

[32] C. Williamson and T. Harrison. On Multipoint Routing in Mesh Based Networks. Technical Report DISCUS TR-95-9, paper.95-9.ps (in preparation), Computer Science Department, University of Saskatchewan, Saskatoon, July 1995.

[33] P. Winter. Steiner Problems in Networks: A Survey. *Networks*, vol. 17, no. 2, pp. 129-167, Summer 1987.

APPENDIX A

CALL FAILURE AND LINK USAGE GRAPHS FOR NON-SHARED LINK MODELS WITH g = 8

The graphs of Percent Failed Calls as a function of multicast group size and Average Links Used per Successful Call with respect to multicast group size appear in this appendix. All plots are based on workloads with a mean multicast group size of 8 recipients. These results can be directly compared with the $g = 4$ graphs presented in Chapter 5.

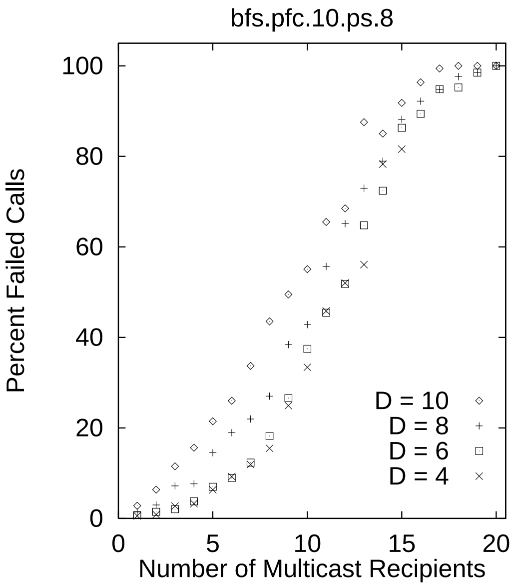

Figure A.1: Percent failed calls, N-UNI model, 10% MC traffic, g = 8.

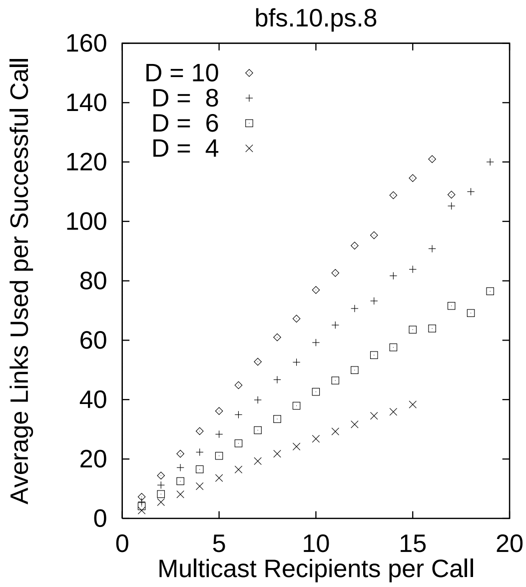

Figure A.2: Avg. links per call, N-UNI model, 10% MC traffic, g = 8.

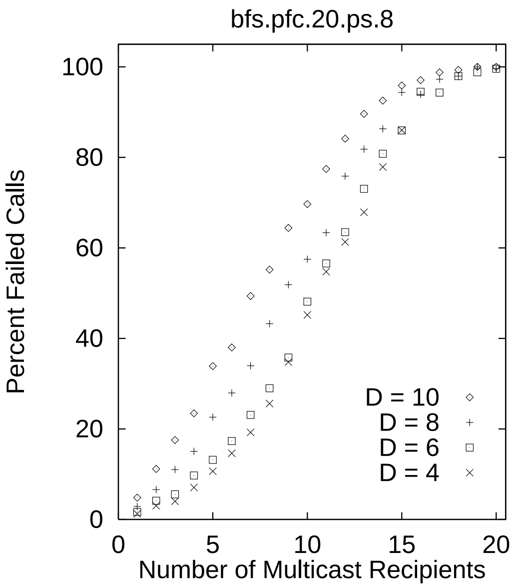

Figure A.3: Percent failed calls, N-UNI model, 20% MC traffic, g = 8.

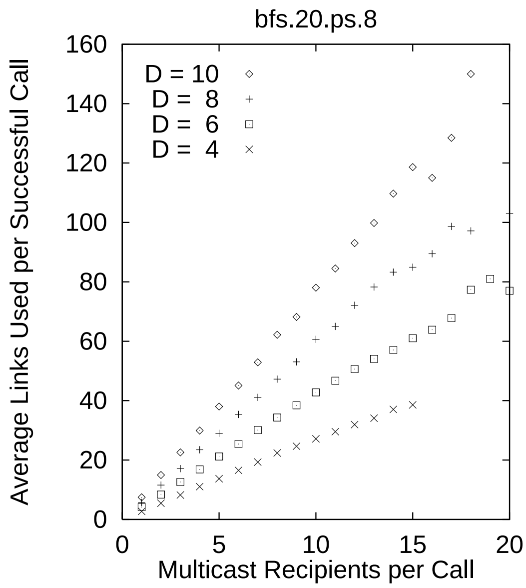

Figure A.4: Avg. links per call, N-UNI model, 20% MC traffic, g = 8.

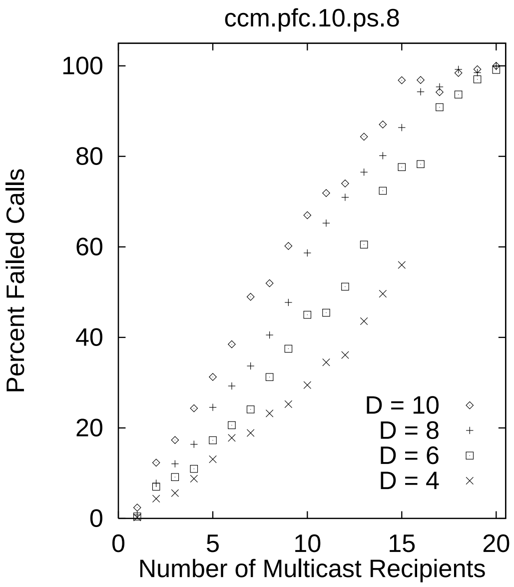

Figure A.5: Percent failed calls, CC model, 10% MC traffic, g = 8.

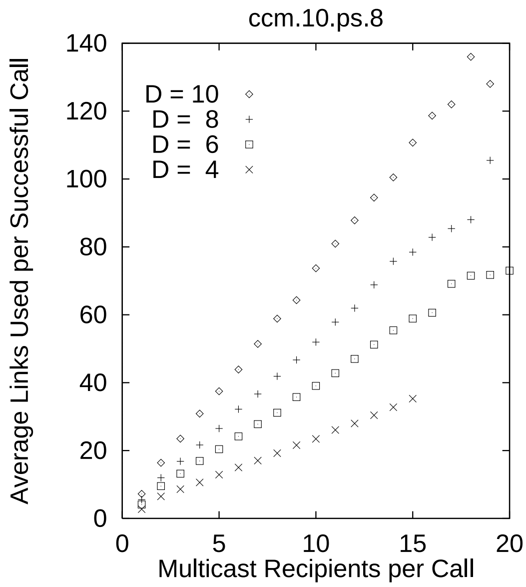

Figure A.6: Avg. links per call, CC model, 10% MC traffic, g = 8.

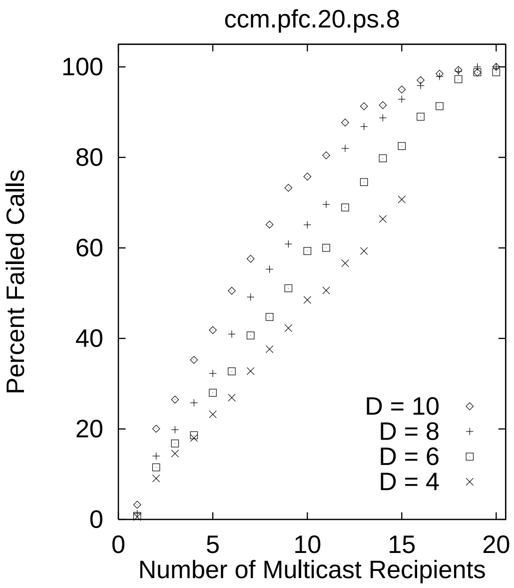

Figure A.7: Percent failed calls, CC model, 20% MC traffic, g = 8.

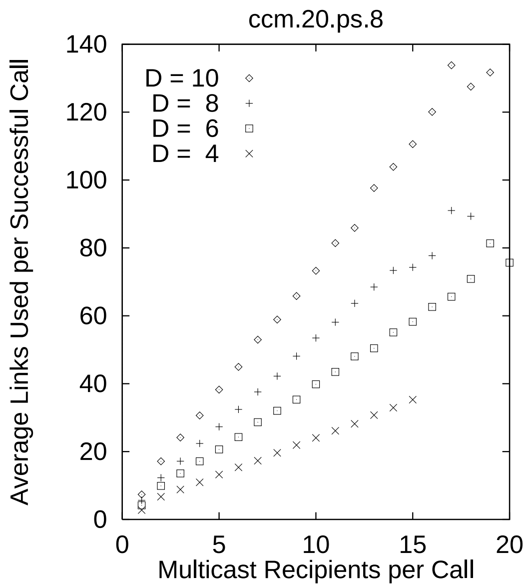

Figure A.8: Avg. links per call, CC model, 20% MC traffic, g = 8.

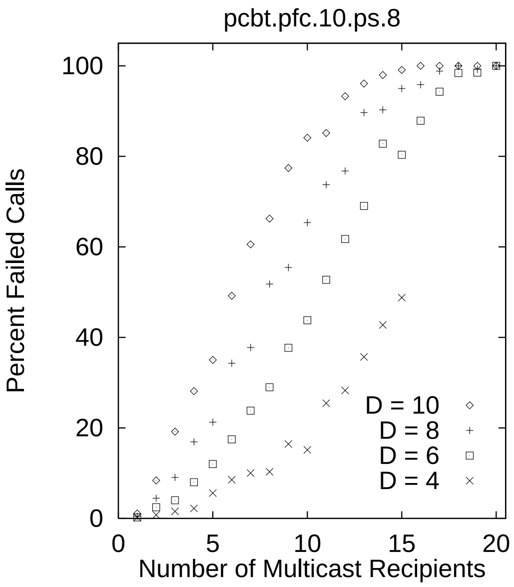

Figure A.9: Percent failed calls, CBT model, 10% MC traffic, g = 8.

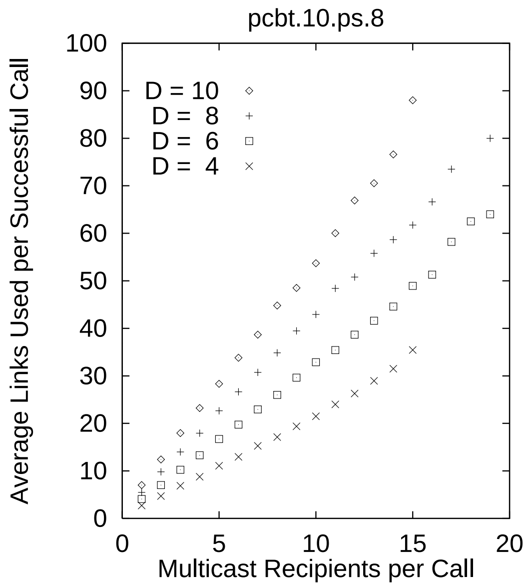

Figure A.10: Avg. links per call, CBT model, 10% MC traffic, g = 8.

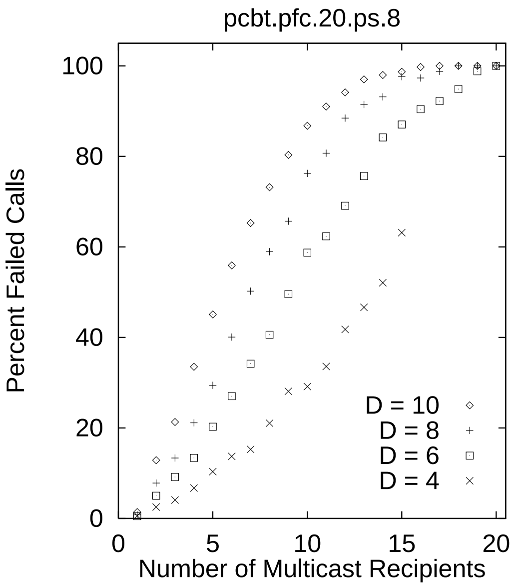

Figure A.11: Percent failed calls, CBT model, 20% MC traffic, g = 8.

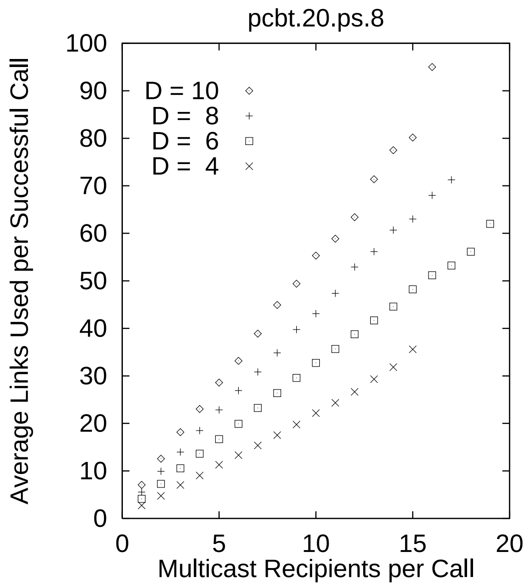

Figure A.12: Avg. links per call, CBT model, 20% MC traffic, g = 8.

APPENDIX B

LINEAR REGRESSION STATISTICS FOR NON-SHARED LINK MODELS WITH g = 8

The tables of regression coefficients for link usage in non-shared link models appears in this appendix. The mean multicast group size used in these tables is 8. Tables B.1, B.2, and B.3 can be directly compared to Tables 5.4, 5.5, and 5.6, respectively, in Chapter 5.

Table B.1: Regression Coefficients for Link Usage in the N-Unicast model (mean multicast group size = 8).

Mesh Size	Percent MC Traffic	Linear Regression Slope	y intercept	R^2
4	10	2.57	0.71	0.998
	20	2.60	0.73	0.998
6	10	4.03	1.05	0.996
	20	4.04	1.15	0.994
8	10	6.09	-2.05	0.994
	20	5.46	2.80	0.989
10	10	7.27	1.32	0.984
	20	7.85	-1.34	0.993

Table B.2: Regression Coefficients for Link Usage in the Conference Call model (mean multicast group size = 8).

Mesh Size	Percent MC Traffic	Linear Regression		
		Slope	y intercept	R^2
4	10	2.23	1.49	0.999
	20	2.23	1.76	0.998
6	10	3.76	1.65	0.996
	20	3.87	1.19	0.994
8	10	5.15	0.94	0.994
	20	4.96	2.52	0.995
10	10	7.12	1.82	0.995
	20	7.20	1.94	0.994

Table B.3: Regression Coefficients for Link Usage in the CBT model (mean multicast group size = 8).

Mesh Size	Percent MC Traffic	Linear Regression		
		Slope	y intercept	R^2
4	10	2.25	-0.23	0.995
	20	2.27	-0.10	0.997
6	10	3.32	-0.19	0.996
	20	3.12	1.21	0.999
8	10	4.12	1.76	0.999
	20	4.14	1.88	0.999
10	10	5.48	0.87	0.996
	20	5.46	0.95	0.993

APPENDIX C

LINK USAGE DIAGRAMS FOR NON-SHARED LINK MODELS WITH g = 8

The average link usage diagrams for selected sizes of non-shared link models are contained in this appendix. All diagrams were based on simulations using workloads with a mean multicast group size of 8 recipients.

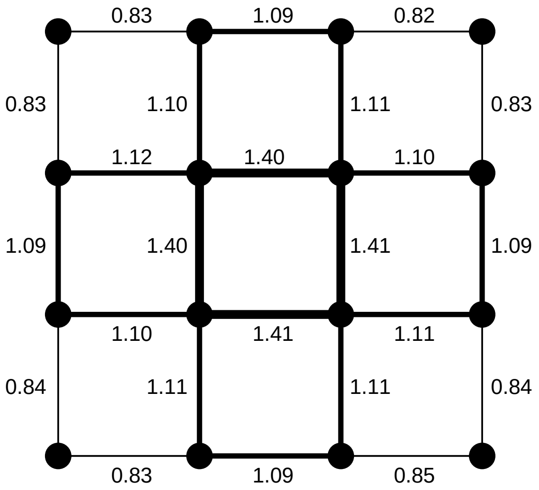

Figure C.1: Avg. link usage, N-UNI model, 10% MC traffic, D = 4, g = 8.

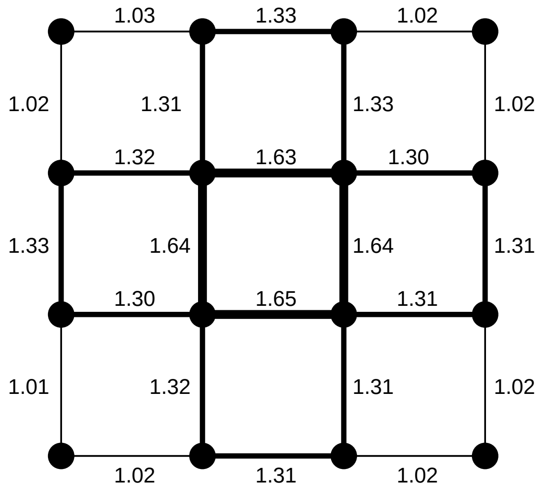

Figure C.2: Avg. link usage, N-UNI model, 20% MC traffic, D = 4, g = 8.

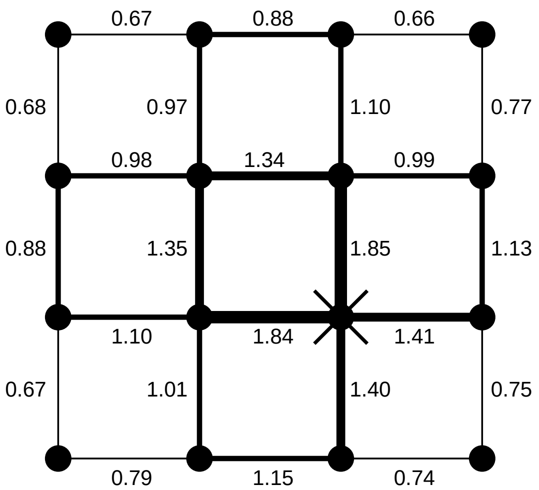

Figure C.3: Avg. link usage, CC model, 10% MC traffic, D = 4, g = 8.

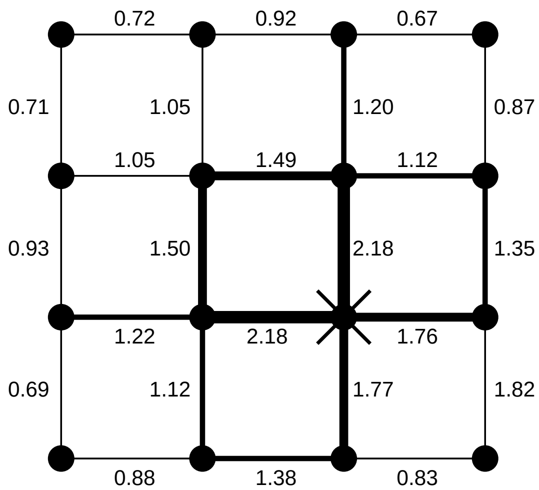

Figure C.4: Avg. link usage, CC model, 20% MC traffic, D = 4, g = 8.

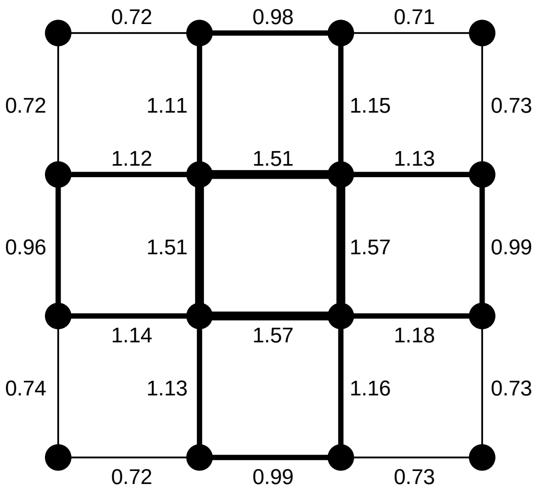

Figure C.5: Avg. link usage, CBT model, 10% MC traffic, D = 4, g = 8.

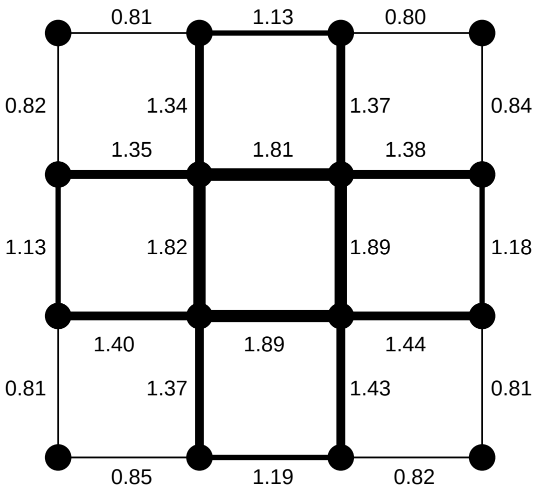

Figure C.6: Avg. link usage, CBT model, 20% MC traffic, D = 4, g = 8.

APPENDIX D

OVERALL PERFORMANCE SUMMARY FOR NON-SHARED LINK MODELS WITH g = 8

The summary table for Average Percent Failed Calls and Average Links Used per Successful Call for the non-shared link simulation models appears in this appendix. The information presented here is based on simulations using workloads with a mean multicast group size of 8 recipients, and corresponds to Table 5.6 in Section 5.5.

Table D.1: Summary of Average Percent Failed Calls (PFC) and Average Links Used per Successful Call for the non-shared link routing models with 10% and 20% multicast traffic (mean multicast group size = 8).

Routing Algorithm		Mesh Size D = 4 Avg. PFC	D = 4 Avg. Links	D = 6 Avg. PFC	D = 6 Avg. Links	D = 8 Avg. PFC	D = 8 Avg. Links	D = 10 Avg. PFC	D = 10 Avg. Links
N-Unicast	10%	2.3	3.7	3.6	5.5	4.9	7.3	6.8	9.1
	20%	5.7	4.7	8.6	6.8	11.1	8.8	14.3	10.7
Conf. Call	10%	2.0	3.7	3.9	5.4	5.0	6.9	7.2	8.8
	20%	6.5	4.5	9.7	6.4	11.5	8.0	14.4	10.1
CBT	10%	1.3	3.6	3.7	5.0	5.0	6.4	6.6	7.9
	20%	4.0	4.5	8.7	5.8	11.0	7.2	13.3	8.6

APPENDIX E

CALL FAILURE AND LINK USAGE GRAPHS FOR SHARED LINK MODELS WITH g = 8

The graphs of Percent Failed Calls with respect to multicast group size and Average Links Used per Successful Call as a function of multicast group size appear in this appendix. All plots are based on workloads with a mean multicast group size of 8 recipients. These results can be directly compared with the $g = 4$ graphs presented in Chapter 6.

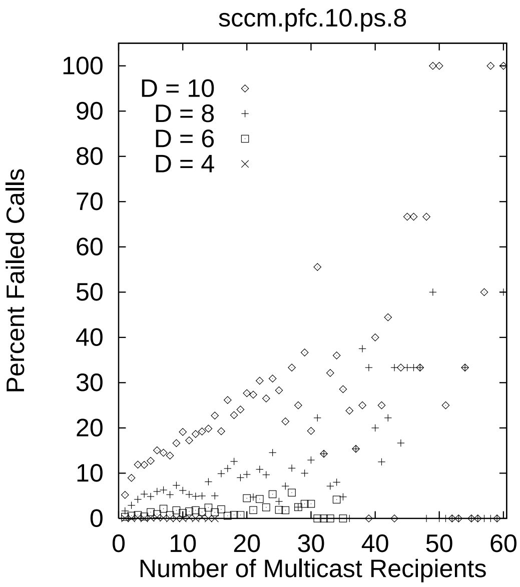

Figure E.1: Percent failed calls, CC model, 10% MC traffic, g = 8.

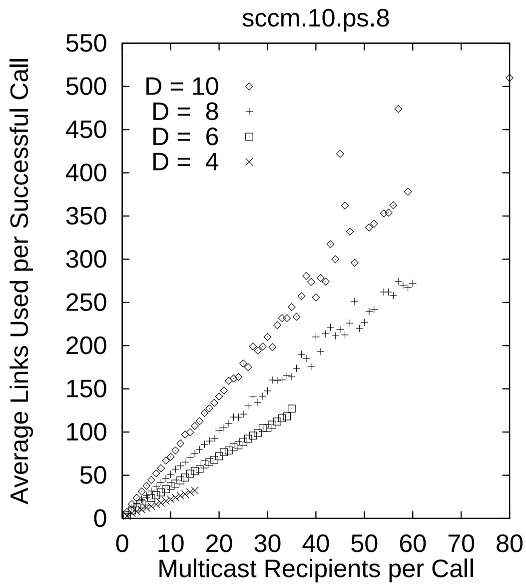

Figure E.2: Avg. links per call, CC model, 10% MC traffic, g = 8.

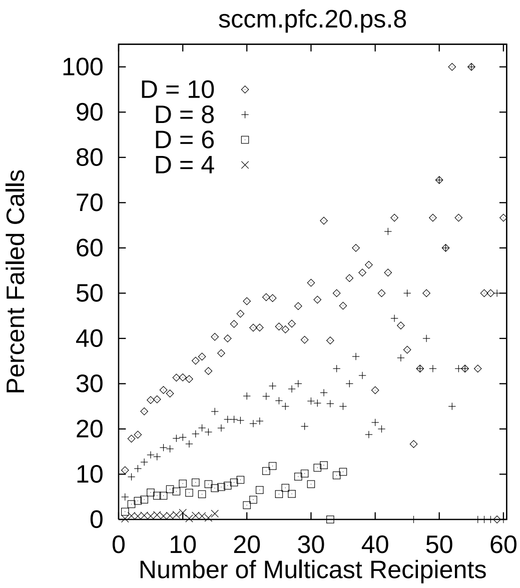

Figure E.3: Percent failed calls, CC model, 20% MC traffic, g = 8.

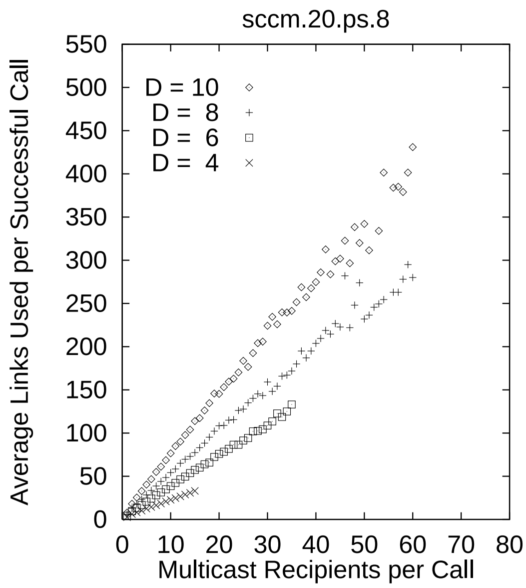

Figure E.4: Avg. links per call, CC model, 20% MC traffic, g = 8.

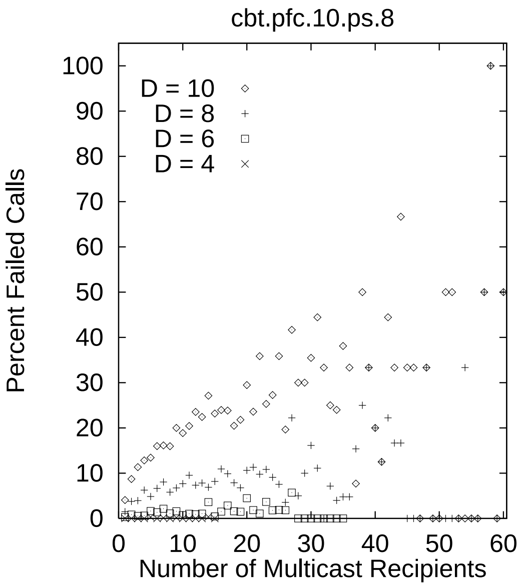

Figure E.5: Percent failed calls, CBT model, 10% MC traffic, g = 8.

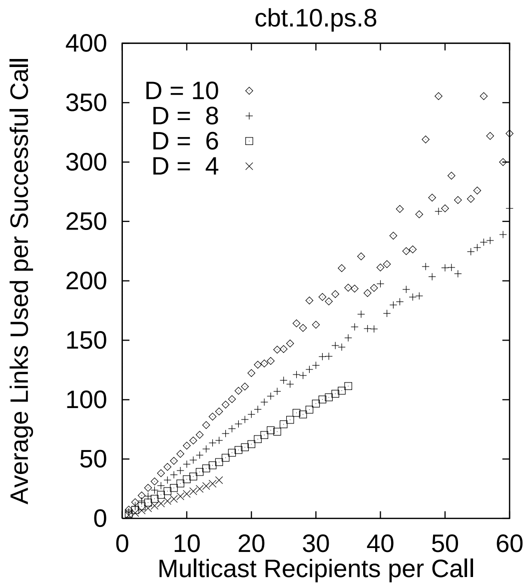

Figure E.6: Avg. links per call, CBT model, 10% MC traffic, g = 8.

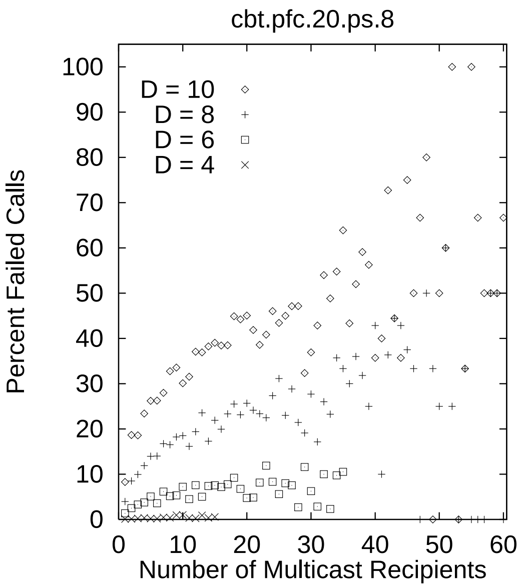

Figure E.7: Percent failed calls, CBT model, 20% MC traffic, g = 8.

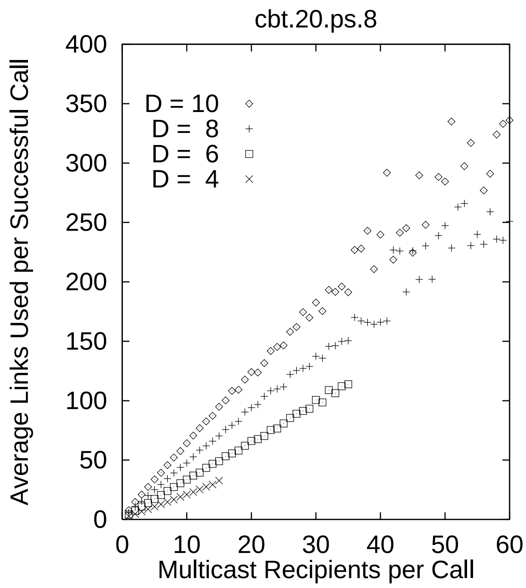

Figure E.8: Avg. links per call, CBT model, 20% MC traffic, g = 8.

APPENDIX F

LINEAR REGRESSION STATISTICS FOR SHARED LINK MODELS WITH g = 8

The tables of regression coefficients for link usage in shared link models appears in this appendix. The mean multicast group size used in these tables is 8. Tables F.1 and F.2 can be directly compared to Tables 6.3 and 6.4, respectively, in Chapter 6.

Table F.1: Regression Coefficients for Link Usage in the Conference Call model (mean multicast group size = 8).

Mesh Size	Percent MC Traffic	Linear Regression		
		Slope	y intercept	R^2
4	10	2.06	1.66	0.999
	20	2.09	1.69	0.999
6	10	3.46	2.48	0.999
	20	3.61	2.33	0.998
8	10	4.64	6.33	0.993
	20	4.78	7.50	0.984
10	10	6.74	6.68	0.965
	20	6.62	11.32	0.990

Table F.2: Regression Coefficients for Link Usage in the CBT model (mean multicast group size = 8).

Mesh Size	Percent MC Traffic	Linear Regression		
		Slope	y intercept	R^2
4	10	2.07	0.31	0.999
	20	2.09	0.30	0.999
6	10	3.16	0.76	0.999
	20	3.23	1.02	0.999
8	10	4.19	4.27	0.985
	20	4.41	4.45	0.974
10	10	5.43	7.58	0.961
	20	5.65	8.69	0.941

APPENDIX G

LINK USAGE DIAGRAMS FOR SHARED LINK MODELS WITH g = 8

The average link usage diagrams for shared link models of size 4 x 4 are contained in this appendix. All diagrams were based on simulations using workloads with a mean multicast group size of 8 recipients.

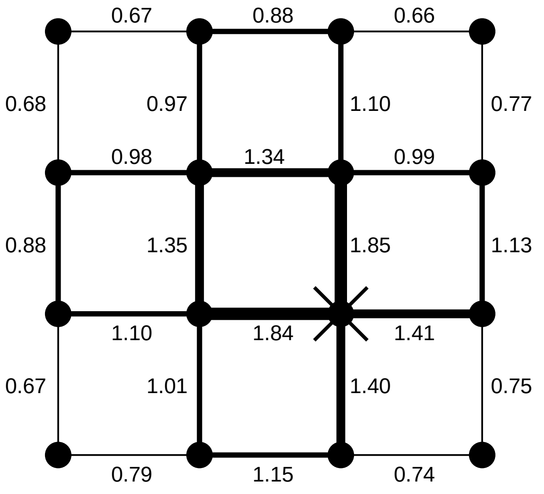

Figure G.1: Avg. link usage, CC model, 10% MC traffic, D = 4, g = 8.

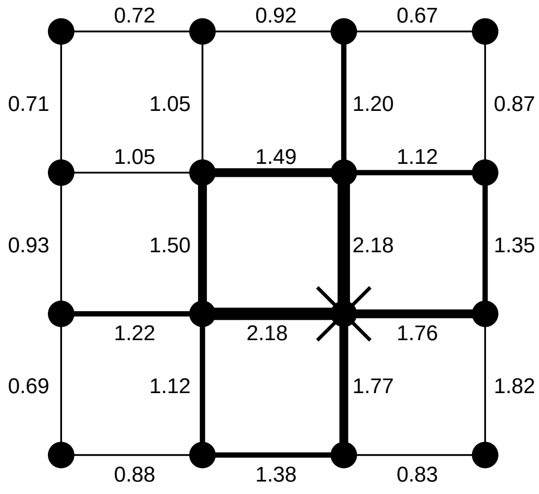

Figure G.2: Avg. link usage, CC model, 20% MC traffic, D = 4, g = 8.

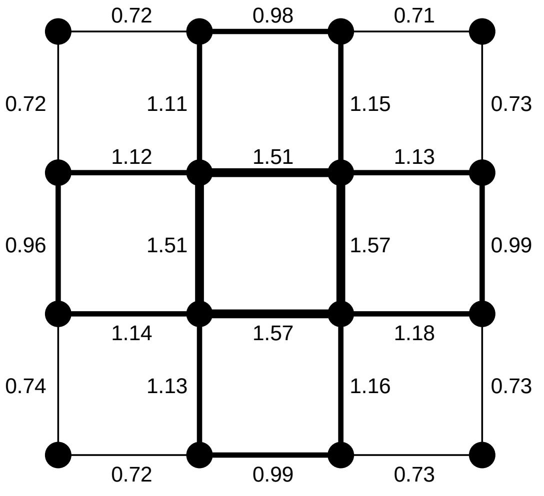

Figure G.3: Avg. link usage, CBT model, 10% MC traffic, D = 4, g = 8.

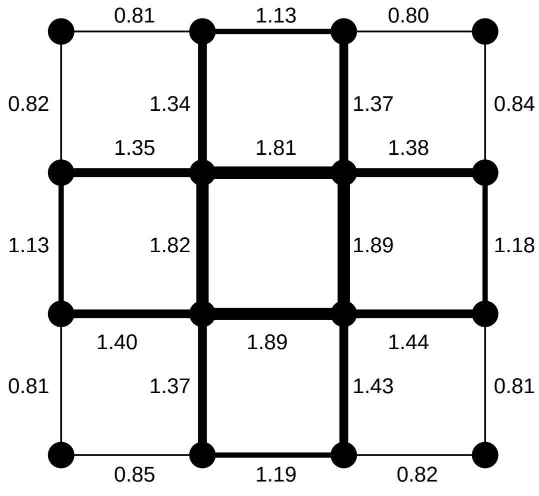

Figure G.4: Avg. link usage, CBT model, 20% MC traffic, D = 4, g = 8.

APPENDIX H

OVERALL PERFORMANCE SUMMARY FOR SHARED LINK MODELS WITH g = 8

The summary table for Average Percent Failed Calls and Average Links Used per Successful Call for the shared link simulation models appears in this appendix. The information presented here is based on simulations using workloads with a mean multicast group size of 8 recipients, and corresponds to Table 6.3 in Section 6.5.

Table H.1: Summary of Average Percent Failed Calls (PFC) and Average Links Used per Successful Call for the shared link routing models with 10% and 20% multicast traffic (mean multicast group size = 8).

Routing Algorithm		Mesh Size							
		D = 4		D = 6		D = 8		D = 10	
		Avg. PFC	Avg. Links	Avg. PFC	Avg. Links	Avg. PFC	Avg. Links	Avg. PFC	Avg. Links
Conf. Call	10%	0.0	3.9	0.5	6.9	2.1	9.5	6.3	12.4
Link Shared	20%	0.3	5.2	2.5	9.9	7.2	13.4	14.5	17.1
CBT	10%	0.0	3.8	0.4	6.4	2.0	8.9	5.4	11.3
Link Shared	20%	0.2	4.9	2.1	9.0	6.4	12.3	12.5	15.1