DMACS: A MEDIA ACCESS PROTOCOL FOR SINGLE-HOP WAVELENGTH DIVISION MULTIPLEXED LIGHTWAVE NETWORKS

by

Michael C. Montgomery

Thesis submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

APPROVED:

Scott F.

Scott F. Midkiff, Chairman

Marc Abrams

Marc Abrams

Ira Jacobs

July, 1994

Blacksburg, Virginia

DMACS: A MEDIA ACCESS PROTOCOL FOR SINGLE-HOP WAVELENGTH DIVISION MULTIPLEXED LIGHTWAVE NETWORKS

by

Michael C. Montgomery

Committee Chairman: Scott F. Midkiff

Electrical Engineering

(ABSTRACT)

This thesis proposes a new media access protocol for local area and metropolitan area all-optical networks employing wavelength division multiplexing (WDM). Through WDM, multiple channels are created on a single fiber, and an aggregate network bandwidth far greater than the peak electronic processing speed can be realized. The new protocol, Dynamic Media Access Control Scheme (DMACS), is based on the Dynamic Interleaved Slotted Aloha (DISA) protocol. It improves on DISA by adding a common control channel that provides reservations for constant bit-rate traffic, acknowledgments, and global flow control. DMACS supports connection setup and tear down, different traffic classes, flow control, and packet resequencing in an attempt to integrate features of the transport layer directly into the media access control layer. The performance of the DMACS protocol has been evaluated through analytical methods and simulation. It was found to be superior to the DISA protocol and to provide good performance that is relatively insensitive to the number of stations and the traffic conditions in the network.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Scott F. Midkiff, for his helpful advice and guidance throughout my tenure as a graduate student. I would also like to thank Dr. Ira Jacobs and Dr. Marc Abrams for serving on my committee.

Most of all, I am grateful to my wife Heather for her patience, prayers, and enduring support.

This material is based upon work supported under a National Science Foundation Graduate Research Fellowship and a Du Pont Graduate Fellowship in Electrical Engineering.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

All-optical networks promise to revolutionize the world of data communications. In an all-optical network, the entire path between end nodes is optical; conversions between electrical and optical form occur only at the endpoints of a connection [1]. It is unclear when all-optical networks will become commercially available, but experimental networks have already shown that they can become a reality [2, 3, 4]. The low-loss region of a single-mode fiber (1.2 to 1.6 μm) provides a bandwidth of approximately 30 THz [5, 6]. Efficiently utilizing this bandwidth is the key to successfully implementing future high-speed networks.

The use of fiber in communication systems has progressed through three generations [1, 7]:

1. fiber not used,
2. fiber used as a replacement for copper, and
3. fiber used for its unique properties.

Examples of first generation networks include IEEE 802.3 (Ethernet) and IEEE 802.5 (token ring) local area networks. Second generation technology includes point-to-point intercity fiber trunks and the Fiber Distributed Data Interface (FDDI). All-optical networks represent third-generation technology. In this generation, the end user will have access to a bandwidth of 1 Gb/s or higher, and the aggregate network bandwidth will be measured in terabits per second. The electronic bottleneck due to the network interface will be overcome through wavelength division multiplexing (WDM), a technique that provides multiple channels on a single fiber, each operating at a maximum data rate set by electronic technology [8, 9].

Many applications exist today that can take advantage of a gigabit network: video conferencing, scientific visualization, computer imaging, and distributed processing, to name a few [7, 10, 11]. Perhaps more importantly, according to Amdahl's rule of thumb, desktop workstations will soon need a gigabit of input/output (I/O) since they are expected to execute one billion instructions per second by the late 1990s [10]. A large portion of this I/O bandwidth is expected to come from the network. As speeds grow higher, network traffic will become more heterogeneous [12], and new applications are bound to develop to take advantage of the increased speed.

A fundamental change is brought about by transmission rates of 1 Gb/s or higher; the network becomes *latency limited* instead of *capacity limited* [13]. The ratio of propagation delay to packet transmission time increases to point that there can be a large number of packets "in the pipe" [12]. As an illustration, consider the following example:

> Suppose there is a 10 km fiber link between two stations in a network. Furthermore, assume that the channel bit rate is 1 Gb/s and the packet length is 1000 bits. The transmission time of a single packet is 1 μs. Since the speed of light in fiber is about 2×10^8 m/s [14], the propagation delay is 50 μs. Therefore, 50 packets can be in transit at one time!

As a result, carrier sensing and token passing protocols suddenly become impractical [1]. Network architects and users must now face the finite speed of light as a limiting factor, and protocol design has to take the latency of the network into account.

The focus of this research is protocol design for a third generation lightwave network. Specifically, a new media access control (MAC) protocol is developed for a single-hop WDM local area or metropolitan area network (LAN or MAN). A MAC protocol coordinates access to a communication network by multiple stations. A single-hop network, explained further in Chapter 2, allows direct communication between every station in the network. The goals of a MAC protocol include the following [15]:

- fair sharing of the network bandwidth,
- high throughput and low, bounded delay,
- support for priorities and different traffic classes,
- simplicity, robustness, and ease of implementation.

The above goals are challenging, but the advent of WDM has made them easier to reach. Several MAC protocols have already been proposed for single-hop WDM networks [6, 16, 17].

The new protocol, Dynamic Media Access Control Scheme (DMACS), is based on the Dynamic Interleaved Slotted Aloha (DISA) protocol [18]. The primary feature of DMACS not found in previous proposals is a relatively low bit-rate common control channel that is not used for coordinating the reception of data packets. DMACS modifies the DISA flow control feature to include global state information through the control channel. In addition, support for different traffic classes is included as found in only one of the previous MAC protocols [19]. Both prior reservation, suitable for long, steady sessions, and walk-in service, suitable for short, bursty traffic, are provided [12]. It has been suggested that high-speed protocols should combine layers of the traditional open systems interconnection (OSI) protocol stack [11]. Due to the absence of the need for OSI network layer functions in a broadcast network, it is desirable to integrate features of the transport layer directly into the MAC layer. DMACS directly supports connection setup and tear down, different traffic classes, flow control, and packet resequencing, functions normally performed in the transport layer. The performance of the DMACS protocol has been evaluated through analytical methods and computer simulation. It was found to be superior to the DISA protocol and to have a maximum stable throughput of approximately 0.6 packets/(slot × channel) for a realistic traffic mix. In addition, the packet arrival rate was found to have a significantly greater effect on the packet latency than the number of stations, the traffic distribution, or the class distribution.

More background information on WDM and previously proposed protocols is given in Chapter 2. Chapter 3 describes the DMACS protocol in detail. The analytical results pertaining to the protocol are presented in Chapter 4. Chapter 5 discusses the simulation model developed for both DMACS and DISA. The results and analysis of the simulation experiments are given in Chapter 6. Finally, Chapter 7 provides a summary of the experimental results, specific contributions of DMACS, and topics for future research.

Chapter 2

Background

Wavelength division multiplexing (WDM) partitions the optical bandwidth into a set of multiple channels, each operating at a unique wavelength. The wavelengths form "an orthogonal set of carriers which can be separated, routed, and switched without interfering with each other, as long as the total light intensity is kept sufficiently low" [5]. This technique is called frequency division multiplexing (FDM) when the channel spacing is on the order of the bit rate. Dense WDM, the technique referred to here, is used for systems where the spacing is on the order of 1 nm. With conventional WDM, which is directed at increasing the capacity of point-to-point systems, the wavelengths are separated by tens or even hundreds of nanometers.

The advantage of WDM is the concurrency among multiple transmissions that can be achieved while each channel is operating at the peak electronic processing speed on the order of a few gigabits per second. A network with an aggregate bandwidth approaching a terabit per second can be realized. Concurrency can also be provided through time division multiplexing (TDM) or code division multiplexing (CDM). TDM requires many slots per bit time, and CDM requires hundreds or thousands of "chips" (waveform samples) per bit time. The number of slots or chips required depends on the number of stations in the network. The disadvantage of TDM and CDM is the need to have stations synchronize to within one time slot for TDM or one chip time for CDM [1, 6]. With WDM, all end-user equipment only needs to operate at the bit rate of a single channel. In the context of a multiuser network, the above techniques are often referred to as wavelength division multiple access (WDMA), time division multiple access (TDMA), and code division multiple access (CDMA). A final dimension for achieving concurrency is space. Currently, space division switches are limited by the lack of optical buffers and optical control [7, 20]. Purely optical space division switches do not seem to be viable in the near future. Space division can also be accomplished by using separate fibers and array sources and detectors at a single wavelength. For a large network, this approach is costly and impractical. At this time, WDM is the most promising approach for building third generation all-optical networks.

The remainder of this chapter presents background on WDM networks and protocols. Technological issues critical to the development of WDM networks are discussed in Section 2.1. Section 2.2 defines the two major classifications for WDM networks, *broadcast-and-select* and *wavelength routing*. As discussed in Section 2.3, these categories can be further divided into *single-hop* networks and *multihop* networks. Possible topologies for broadcast-and-select networks are presented in Section 2.4. Finally, as motivation for the development of the DMACS protocol, previously proposed MAC protocols for single-hop WDM networks are reviewed in Section 2.5.

2.1 Critical Technologies

Wavelength agility, the ability to switch between channels at different wavelengths, is accomplished through tunable transmitters and/or receivers. The devices can have either continuous or discrete tuning; selection of discrete wavelengths is required for the systems of interest in this work. Building a tunable transmitter suitable for WDM requires a narrow linewidth tunable laser. Ideally, tuning would take a few nanoseconds or less, and the range would be wide enough to support several hundred or a thousand channels (50 nm or more). The fastest tuning times have been achieved with distributed feedback (DFB) and distributed Bragg reflector (DBR) semiconductor lasers which are tuned by adjusting the injection currents. An experiment with a two-section DFB laser demonstrated a tuning range of 0.32 nm with a tuning time of less than 5 ns [5]. A three-section DBR laser has been reported with a tuning speed of 1.8 ns over a wavelength range of 12.5 nm [5]. Another possibility is an external-cavity semiconductor laser with an acousto-optic tunable filter within the cavity. A tuning range of 83 nm with a tuning time of a few microseconds has been demonstrated; however, the tuning time is limited by the acoustic propagation velocity [5]. Clearly, either the range or the tuning speed of these devices needs to increase. In addition, stability at the selected wavelength is another important issue.

Tunable filters can be used to implement tunable direct detection receivers. Passive filters include Fabry-Perot and Mach-Zehnder filters. Fabry-Perot filters can achieve very fine frequency resolution, but their tuning speed, at least hundreds of microseconds, is limited by mechanical inertia [1]. A Mach-Zehnder filter has been demonstrated with 100 resolvable wavelengths and millisecond tuning times, but its tuning speed is limited by thermal inertia [5, 7]. Active electro-optic filters and laser-diode amplifiers (as tunable filters) can achieve tuning times on the order of nanoseconds but have limited range [5]. It is also possible to use a tunable laser as the local oscillator in a tunable coherent receiver [21]. The polarization of the local oscillator must match that of the incident light. Polarization control is a significant problem that must be faced with coherent receivers. Coherent techniques achieve greater receiver sensitivity than direct detection at the expense of increased cost and complexity.

Recent progress in erbium-doped broadband optical amplifiers have reduced power budget worries due to splitting and excess losses in passive optical couplers. These amplifiers provide 20 to 30 dB of gain across thousands of gigahertz of bandwidth without the need for optical-to-electrical and electrical-to-optical conversions [1, 6]. Such an amplifier could be placed immediately preceding a direct detection receiver to increase its sensitivity close to that of a coherent heterodyne receiver when using amplitude shift-keying (ASK) as the modulation format [7]. Optical amplifiers also make it possible to build an all-optical wide area network (WAN).

The technology for a circuit-switched WDM network is available today. Progress needs to be made before fast packet switching can be implemented in an all-optical network with a large number of channels, but recent research is promising.

2.2 Broadcast-and-Select Versus Wavelength Routing

WDM lightwave networks can be classified as either broadcast-and-select networks or wavelength routing networks [5, 16]. In a broadcast-and-select network, a station's transmission is broadcast to all other stations. The receiver at the destination extracts the desired signal from the entire group of signals transmitted. In wavelength routing networks, the wavelength of a signal is used to route it through the network. The routing can be either fixed or dynamic. Dynamic wavelength routing is needed to support packet switching.

Problems with broadcast-and-select networks include lack of wavelength reuse, splitting loss, and scalability to WANs [16]. Wavelength routing offers wavelength reuse at the expense of costly hardware and complex control. Dynamic wavelength routing is essentially a hybrid wavelength-space division approach that suffers from the same lack of optical buffers and optical control as a pure space-division switch. The broadcast-and-select approach seems to be more suitable for LANs or MANs, while wavelength routing is suitable for WANs. It is envisioned that a point-to-point wavelength routing network can be used to interconnect many broadcast-and-select subnets [16].

2.3 Single-Hop Versus Multihop

Both broadcast-and-select and wavelength routing networks can be further classified as single-hop or multihop [16, 6, 22]. A single-hop network allows direct communication between any two stations. In a multihop network, a transmission may have to travel through intermediate stations before reaching its destination. For a broadcast-and-select network, single-hop communication requires tunable transmitters and/or receivers. A multihop broadcast-and-select network can be built with a small number of fixed transmitters and receivers at each station. With dedicated channels, each transmitter is at a different wavelength, and a single receiver at another station is tuned to that transmitter. Stations can transmit directly only to receivers tuned to one of their transmit wavelengths. Thus, for multihop networks, a logical topology is laid over the physical broadcast topology. With dedicated channels, no MAC protocol is needed to enable channel sharing. However, shared channel approaches have also been proposed [22]. Multihop networks can be built with existing technology, but electronic switching is used at intermediate stations. Logically, the network is no longer all-optical as it is in the single-hop case.

2.4 Physical Network Topologies

Two fundamental characteristics of fiber optic networks that differentiate them from copper networks are [9]:

- limited signal power, and
- abundant transmission bandwidth.

Fiber networks are "power-limited," not "bandwidth-limited" [9], while copper networks are the opposite. Power limits are a primary consideration when choosing the physical topology for an all-optical LAN or MAN.

Physical topologies that have been considered for broadcast-and-select local area WDM networks include a star, a unidirectional bus, and a tree. These topologies are shown in Figure 2.1. Traditionally, an $N \times N$ star coupler has been constructed from $\log_2 N$ stages of passive 2×2 couplers (4×4 couplers can be used [7]). The taps in a bus and the combiners and splitters in a tree are constructed from 2×2 couplers as well. Note that the physical network is totally passive except for the possible use of optical amplifiers.

The star topology is superior to a bus for the following reasons [9, 16, 7].

- Assume a station transmits with power P and there are N stations in the network. With a bus, *splitting losses* reduce the power of a transmitted signal to P/N after passing through the "talk side," assuming splitting ratios are adjusted to make the effective power coupled to the network the same for all stations. This signal must then be distributed among the N receivers on the "listen side," so it is further attenuated by $1/N$ and a signal with power P/N^2 is received by each station. With a star topology, the power of a transmitted signal is evenly distributed among the N output fibers and each station receives a signal with power P/N.
- When expressed in decibels, the total *excess loss* in a bus due to non-ideal couplers grows linearly with the number of stations. This loss grows logarithmically with the number of stations in a star.
- Even with broadband optical amplifiers, a star can support more stations than a bus in a multichannel environment because of a significant reduction in amplification bandwidth when many amplifiers are cascaded.

Therefore, it is clear that a star is more power efficient than a bus.

The tree topology has not received as much attention as the star or bus. It has been shown that a tree is more cost effective than a star for a MAN [23]. Without amplification,

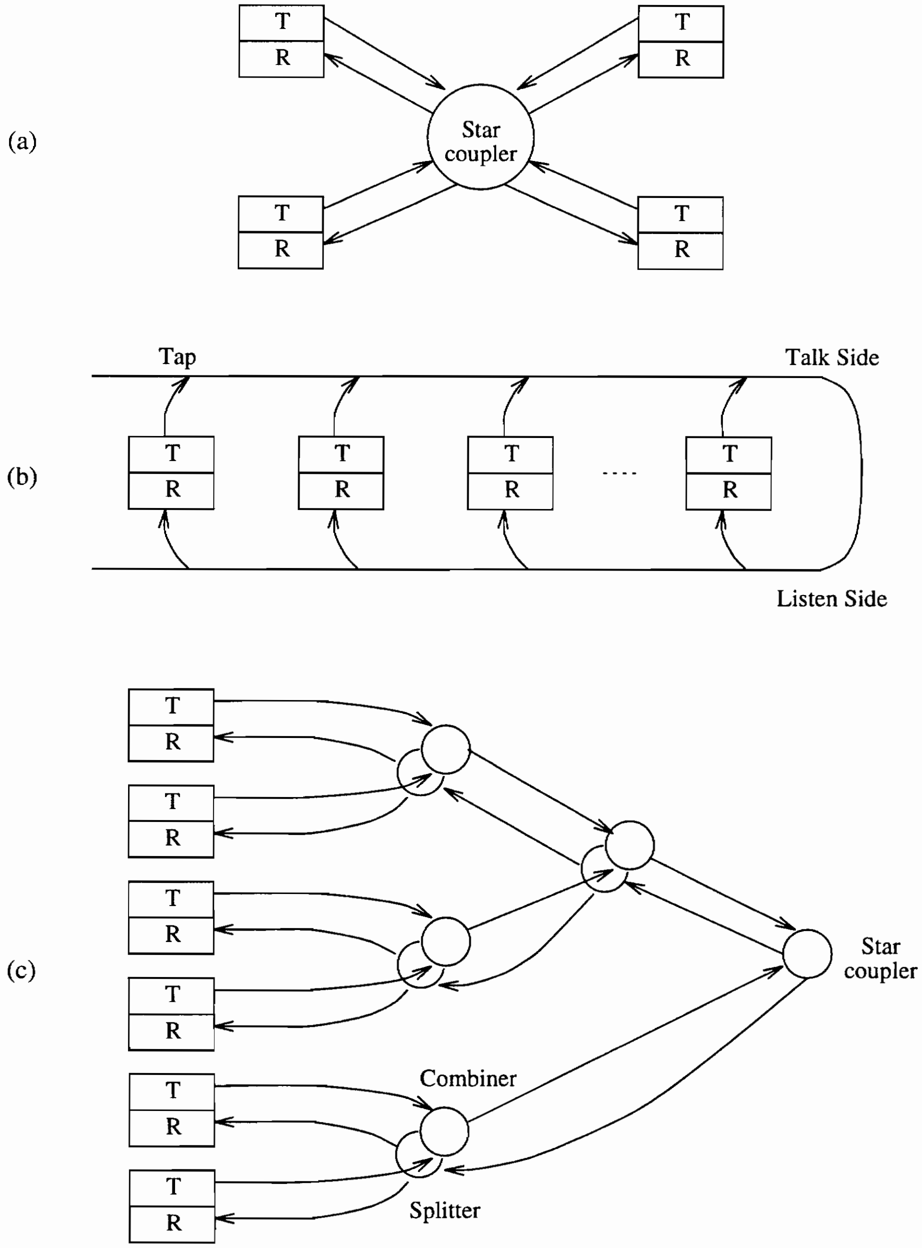


Figure 2.1: Possible physical topologies for a local area WDM network: (a) star, (b) unidirectional bus, and (c) tree.

a star can support more stations than a tree, 1024 with a star instead of 32 with a tree for typical power budget calculations [24]. However, with amplification, a tree or hybrid tree/star topology becomes more attractive since amplifiers can be installed at the roots of subtrees [24]. A pure star requires one amplifier per port. A star does have the advantage of easier wiring. In addition, a faulty cable in a star isolates a single station whereas in a tree it might isolate many stations. Optimization of the physical topology is still an open area of research for all-optical LANs and MANs. The star topology is chosen for this study because of the advantages over a bus and a tree discussed above. The DMACS protocol does work on any single-hop broadcast-and-select WDM network, independent of the physical topology.

2.5 Review of MAC Protocols for Single-Hop WDM Networks

Numerous MAC protocols for single-hop broadcast-and-select WDM networks have been proposed. Only protocols that work on any broadcast topology are considered here. Protocols that apply only to a unidirectional bus have been proposed, but they are not discussed.

If stations in the network are equipped with tunable transmitters (TTs) and fixed receivers (FRs), channel collisions are possible as in a conventional single-channel broadcast network. With fixed transmitters (FTs) at unique wavelengths and tunable receivers (TRs), *destination conflicts* can occur when two or more stations transmit on different wavelengths to the same destination at the same time. In this case, the destination can only receive one of the transmissions. If stations have TTs (or FTs with shared wavelengths) and TRs, both collisions and destination conflicts are possible.

The proposed protocols can be divided into two categories [6]: (1) pretransmission coordination required, or (2) no pretransmission coordination required. Pretransmission coordination is generally required when TRs are used. The receiver must be informed of the wavelength of each incoming transmission. All of the protocols based on pretransmission coordination employ one or more control channels, possibly embedded on the data channels, for accomplishing this purpose. Note that this coordination is not required if the transmission schedule is fixed or pre-determined in a distributed manner at each station. With FRs, no pretransmission coordination is required since each receiver is at a fixed wavelength for reception of incoming traffic. A review of the proposed protocols based on pretransmission coordination is given in the next section. In Section 2.5.2, the protocols based on no pretransmission coordination are discussed.

2.5.1 Protocols Based on Pretransmission Coordination

The protocols requiring pretransmission coordination are summarized in Table 2.1 using the following metrics.

- **Equipment**: This lists the components required at each station. Note that C is defined to be the number of channels.
- **Channels**: The first number of the pair is the number of control channels required by the protocol. The second number is the number of data channels. N is defined to be the number of stations in the network.
- **Processing**: This gives a relative measure of the processing requirements. Monitoring of a single control channel with packet headers for all the network traffic can become an electronic processing bottleneck. Distributed algorithms executed to avoid collisions and/or destination conflicts further increase the processing requirements.
- **Tell and go**: This feature allows a station to inform the destination it is transmitting a packet and then transmit it without waiting [16, 19]. In reservation protocols, a station must wait for at least one round-trip delay before transmitting.

Table 2.1: Comparison of MAC Protocols Based on Pretransmission Coordination

Protocol	Equipment	Channels	Processing	Tell and go	Through-put	Comments[a]
Aloha/Aloha [25]	TT,TR	$1, \geq 1$	High	Yes	Low	No Sync.
S-Aloha/Aloha						Sync.
Aloha/CSMA						No Sync.
CSMA/Aloha						
CSMA/N-Server						
CLAR [26]	FT,TT,FR,TR	$1, \geq 1$	High	No	Low	CA,DCA, Var.
Improved S-Al/Al[27]	TT,TR	$1, \geq 1$	High	No	Low	
S-Aloha/N-Server						
S-Al/Polite Access [28]	TT,TR	$1, \geq 1$	High	No	Low	CA
S-Al/Sync. N-Server						
MC Demand Assignment [29]	C FTs,C FRs, Mux/DeMux	$1, \geq 1$	High	No	Low	CA,DCA
Improved Aloha/S-CSMA [30]	TT,TR	$1, \geq 1$	High	No	Low	
MC S-Al/S-Al [31]	TT,TR	$\geq 1, \geq 1$	Medium	Yes	Low	Emb. CC
MC Impr. S-Al/S-Al				No		
S-Aloha/S-Aloha [32]	TT,TR	$1, \geq 1$	High	Yes	Low	
Improved S-Al/S-Al				No		Some CA
Reservation Aloha				No		CA
S-Al/Reservation Protocol [33]	FT,TT,FR,TR	$1, \geq 1$	Very High	No	Medium	CA,DCA
RCA [34, 35]	TT,TR	$1, \geq 1$	Very High	No	Medium	CA,DCA
MultiS-Net [36]	TT,TR	$1, \geq 1$	Very High	No	Medium	CA,DCA
DT-WDMA [37]	2 FTs,FR,TR	$1, N$	High	Yes	Medium	CA
Conflict-free DT-WDMA [38]	2 FTs,FR,TR	$1, N$	Very High	No	High	CA,DCA
TDMA-C [39, 40]	TT,FR,TR	$1, \geq 1$	Very High	No	Medium	CA,DCA, Var.
DAS [41, 42]	2 FTs,FR,TR	$1, N$	Very High	No	High	CA,DCA,
HTDM			High	Some		Sync.Rand.
Quadro [43, 44]	2 FTs,FR,TR, Delay lines	$1, N$	High	Yes	High	CA, Some DCA
POPSMAC [45, 46]	FT,TT,FR,TR	N, N	Low	Yes	Low	CA,Var., No Sync.
N-DT-WDMA [19]	FT,TT,FR,TR	N, N	Medium	Yes	High	CA

---

[a] Key to Comments: No Sync. = no network-wide synchronization is required; CA = collision avoidance protocol; DCA = destination conflict avoidance protocol; Var. = variable-sized data packets are supported; Emb. CC = control channels embedded on data channels; Sync. Rand. = synchronized random number generators with identical seeds are required.

- **Throughput**: This is a relative measure of the maximum achievable throughput. Collisions and/or destination conflicts limit the throughput as the offered load increases. In general, dynamic reservation schemes that avoid collisions and destination conflicts can achieve the highest throughput.

When there are multiple protocols in a single reference, values in a column that are the same are not repeated.

Except where noted in the comments, all protocols require network-wide synchronization and fixed-size packets. Also, protocols that avoid channel collisions and destination conflicts are marked. All of the pretransmission coordination-based protocols allow random access to each channel except for HTDM, a hybrid protocol, which employs a combination of fixed and random access.

In [25], five protocols are proposed. They are denoted as X / Y, where X is the control channel protocol and Y is the data channel protocol. For example, in the Aloha/Aloha protocol, a busy station randomly picks a data channel number and transmits a control packet containing the source, destination, and channel information on the control channel. The data packet is then immediately transmitted on the selected data channel. If there are no collisions on the control and data channels and the receiver is idle, the packet can be successfully received. Note that to efficiently utilize the data channels, a control packet has to be much smaller than a data packet. Protocols in [26, 27, 28, 30, 31, 32, 33, 34, 35, 36] attempt to improve upon this scheme. A commonly proposed improvement is to only transmit a data packet if the control packet is successful. This eliminates the tell and go feature. In [32, 33, 34], there are C minislots on the control channel per slot on the data channels, where C is the number of data channels. A successful control packet in the i^{th} minislot is followed by transmission of the data packet on wavelength λ_i which eliminates data channel collisions.

All of the above protocols use carrier sense multiple access (CSMA) or Aloha for either the control channel, the data channels, or both. It is well known that CSMA is not an effective protocol when the ratio of propagation delay to transmission time is greater than a few tenths [47]. In a gigabit LAN or MAN, this ratio is much greater than one. The Aloha protocol is not sensitive to propagation delay, but the throughput is limited ($1/e$ for slotted Aloha) [47].

Dynamic Time-Wavelength Division Multiaccess (DT-WDMA) [37] uses fixed TDMA on the control channel for broadcasting packet headers. Each station has a unique fixed wavelength for transmission of data packets, so collisions are avoided. When busy, a station transmits a control packet in its assigned time slot on the control channel and follows through by transmitting a data packet on its data channel. All stations monitor the control channel with a fixed receiver. Destination conflicts can occur, causing execution of a global distributed algorithm to determine which packet to receive. Protocols in [38, 39, 40, 41, 42] also use TDMA on the control channel, but they avoid destination conflicts through reservations. Quadro [43, 44], another similar protocol, uses delay lines to buffer packets arriving simultaneously on different wavelengths. In all of these schemes, the control channel becomes a processing bottleneck as N, the number of stations, grows large. In addition, the timing requirements for synchronization are stringent since there are N control minislots per data slot.

The Passive Optical Packet-Switched Medium Access Control (POPSMAC) protocol [45, 46] and N-DT-WDMA [19] allocate a control channel as well as a data channel to each station, requiring $2N$ channels for N stations. Each station has a FR at a unique wavelength to monitor its control channel and a FT at a unique wavelength for transmitting its data packets. The added control channels significantly reduce the processing requirements since each station only needs to monitor control packets intended for itself. POPSMAC is an extension of the Aloha/Aloha protocol [25] that is designed for a MAN or WAN. Note that it does not require network-wide synchronization. N-DT-WDMA is the first proposed protocol to support different traffic classes in an attempt to integrate the MAC layer with the transport layer. The three supported classes are connection-oriented with guaranteed bandwidth, connection-oriented without guaranteed bandwidth, and datagram traffic. DMACS supports the same classes but does not require pretransmission coordination.

2.5.2 Protocols Based on No Pretransmission Coordination

The protocols that do not require pretransmission coordination are summarized in Table 2.2. The tell and go column is not shown because it is not necessary for a station to tell the destination that it is transmitting. In the equipment column, k is used to denote an arbitrary number of transmitters or receivers. Protocols with only a single number in the channel column do not utilize any control channels. In general, these protocols do not need a control channel since pretransmission coordination is not required. As noted in the second column, each of the protocols falls into one of the following classes.

- **Token Passing**: A token allowing transmission is passed in round-robin order.
- **Fixed**: The transmission schedule is fixed and does not change dynamically.
- **Random**: Random access to each channel is allowed.
- **Hybrid**: A combination of fixed and random access is used.

By nature, these protocols all avoid destination conflicts. In addition, all of them avoid collisions except for Slotted Aloha (S-Aloha), Interleaved Slotted Aloha (I-SA), and I-SA*. Dynamic Interleaved Slotted Aloha (DISA) and DMACS are dynamic and do allow collisions at low offered loads. The difference between I-SA* and Interleaved TDMA* (I-TDMA*)

Table 2.2: Comparison of MAC Protocols Based on No Pretransmission Coordination

Protocol	Protocol class	Equipment	Channels	Processing	Throughput	Comments[a]
MC Token Passing [48]	token passing	C FTs,C FRs, Mux/DeMux	1, ≥ 2	Low	Very Low	No Sync.
MC Piggyback Token Passing [49]	token passing	C FTs,C FRs, Mux/DeMux	≥ 2	Low	Very Low	No Sync.
Source/Dest. Allocation [50]	fixed	TT,TR	≥ 1	Low	High	
TWSA [51]	fixed	k TTs,k TRs	≥ 1	Low	High	
Many-to-One [52] One-to-Many One-to-One Many-to-Many	fixed	TT,FR or FT,TR or TT,TR	N	Low	High	
S-Aloha [53] Random TDMA	random	TT,k FRs	≥ 1	Low High	Low High	 Sync. Rand.
I-SA [54, 55, 56, 40] I-TDMA	random fixed	TT,FR	≥ 1	Low	Low	
I-SA* [57, 55, 40] I-TDMA*	random fixed	TT,FR	≥ 1	Low	Low High	
PAC [58, 59]	random	TT,TR,PAC, Control star	N	Low	Medium	No Sync.
DISA [18]	hybrid	TT,FR	≥ 1	Medium	High	
DMACS	hybrid	FT,TT,2 FRs	1, N	Medium	High	

---

[a] Key to Comments: No Sync. = no network-wide synchronization is required; Sync. Rand. = synchronized random number generators with identical seeds are required.

and their counterparts, I-SA and I-TDMA, is that they have a separate queue for each channel's packets instead of a single queue for all packets. In addition, I-SA* uses a different acknowledgment scheme than I-SA.

Since the ratio of propagation delay to transmission time is high, the token passing protocols [48, 49] do not allow much concurrency and thus perform poorly in a high-speed multichannel LAN/MAN. The fixed protocols [50, 51, 52, 54, 57, 55, 40] do not accommodate bursty traffic efficiently. With these protocols, a method to change the slot assignment pattern dynamically should be provided. The random protocols [53, 54, 55, 56, 40, 57, 58, 59] handle bursty traffic well but have limited throughput. The Protection-Against-Collision (PAC) Optical Packet Network is a hardware solution that employs carrier sensing [58, 59]. Only one "control" star is needed in the network. Each station, however, does need its own PAC circuit. This scheme is attractive for a photonic switch or multiprocessor interconnect, but not for a LAN or MAN.

DISA [18] is a hybrid protocol which functions as a random protocol at low loads and a fixed protocol (TDMA) at high loads. DISA uses a local flow control threshold based on the number of collisions plus arrivals in the past cycle to switch between the random access and fixed modes of operation. DMACS, which is described in detail in the next chapter, is an extension of DISA that employs a common control channel not used for pretransmission coordination. Reservations are allowed at any level of offered traffic. It has been noted that a hybrid between reservation-based and random access schemes "could provide the ideal solution" [42].

Chapter 3

Protocol Description

This chapter explains the operation of the DMACS protocol. It begins with an overview of the network architecture for which the protocol is intended. The actual definition of the protocol is presented in Sections 3.2 through 3.7. Where applicable, implementation-dependent choices are denoted. Section 3.8 gives an example of the data packet transmission algorithm. Finally, four possible extensions to the protocol are outlined in Section 3.9.

3.1 Network Architecture

The network architecture under consideration is a single-hop broadcast-and-select WDM LAN/MAN consisting of N stations connected via a $N \times N$ passive star. Throughout this chapter, N is defined to be the number of stations, and C is the number of channels or wavelengths for transmission of data. Each station is equipped with a fixed transmitter (FT), a tunable transmitter (TT), and two fixed receivers (FRs). The FT and one FR are tuned to a common wavelength used for a control channel. Assuming the number of available wavelengths is greater than N, the other FR is tuned to a unique wavelength used for a data channel. This means that $C = N$, and $N + 1$ wavelengths are needed. The TT is capable of tuning over the entire range of wavelengths. This is a CC-FT-TT-FR^2 system according to the classification scheme in [6]. In Section 3.9, it is shown that the protocol can be implemented with only a TT and a tunable receiver (TR) per station and with fewer than $N + 1$ wavelengths.

Time is slotted on data packet boundaries with a cycle being composed of N slots. Addition and removal of stations does affect the cycle length because the value of N changes. As with most protocols proposed so far for all-optical networks, network-wide cycle and slot synchronization is assumed. The existence of a common control channel allows broadcasting of a synchronization signal periodically [18]. Furthermore, synchronization in DMACS is required on data packet boundaries instead of control packet boundaries making timing requirements less stringent.

3.2 Definitions

Relevant definitions for DMACS, including those from the DISA protocol description [18], are given below.

- **TDMA destination**: For a given slot, this is the TDMA-allocated destination channel for the given source station [18].
- **circuit destination**: For a given slot, the circuit destination channel, if it exists, supersedes the TDMA destination.
- **random destinations**: These are all destinations other than the TDMA destination in a given slot, possibly including the circuit destination [18].
- **grand cycle**: A cycle of $k \times N$ data slots. A normal cycle is composed of N slots, so a grand cycle is k normal cycles.
- **collision threshold**: This threshold is the number of collisions per grand cycle at which a station stops attempting to transmit to random destinations on a particular channel. During each grand cycle, each station broadcasts a collision threshold bit

indicating whether a certain number of collisions occurred during the previous grand cycle on that station's home channel.

- **arrival threshold**: This threshold is the number of system-wide arrivals per grand cycle at which a station stops attempting to transmit to random destinations on any channel. During each grand cycle, each station broadcasts the number of arrivals that occurred at that station during the previous grand cycle. The total number of arrivals for all stations is then compared to the threshold value.

Three different classes of traffic are supported as in N-DT-WDMA [19]:

- **Class 1**: Connection-oriented with guaranteed bandwidth.
- **Class 2**: Connection-oriented without guaranteed bandwidth.
- **Class 3**: Connectionless, datagram traffic.

Each station maintains three queues for each of the $N-1$ destination channels. Packets of a particular class with a common destination channel share the same queue. Each of the queues has the following flag:

- **R flag**: The ready flag indicates whether a packet is ready to be transmitted on this channel [18].

Queues associated with class 2 and class 3 traffic (if made reliable in the implementation) also have the following flags:

- **B flag**: The backoff flag indicates whether this queue is in the backoff state due to a collision in a previous transmission on this channel [18].
- **W flag**: The waiting flag indicates whether m packets in this queue are outstanding, where m is the maximum number of packets allowed to be outstanding at once.

0	1 31
VALID	SLOT
REQUEST/RELEASE	CHANNEL

Figure 3.1: The format for one reservation command in a control packet.

3.3 Control Channel Operation

The control channel operates under fixed TDMA. During each cycle of N slots, a station broadcasts a control packet in its assigned slot. Unlike other protocols with a single control channel, a control packet from each station is not broadcast during each data slot, but during each cycle composed of N data slots. This significantly reduces processing requirements.

The control packet consists of reservation commands, acknowledgments, the collision threshold bit, the number of arrivals in the last grand cycle, and a CRC. The number of arrivals only needs to be included for the first control packet broadcast in each grand cycle. A possible format for the reservation command using 8 bytes is given in Figure 3.1. The reservation commands are broadcast on the control channel to request or release slots for circuit connections which carry traffic requiring guaranteed bandwidth. To prevent complete blocking of class 2 and class 3 traffic, no reservations are allowed during the last cycle of a grand cycle. In the implementation under study, new connections are not blocked if the needed bandwidth is not available, although this would be a desirable feature to implement. A limit on slots reserved for one circuit connection could be imposed. Each station has status tables to keep track of reserved (slot, channel) pairs as explained in Section 3.4.

A possible format for the acknowledgment using 12 bytes is given in Figure 3.2. As will be further explained in Section 3.7, acknowledgments are necessary for making class 2 or class 3 traffic reliable. A selective repeat automatic repeat request (ARQ) protocol

<table>
<tr><td>0</td><td>1   31</td></tr>
<tr><td>VALID</td><td>DESTINATION ADDRESS</td></tr>
<tr><td colspan="2">REQUEST NUMBER</td></tr>
<tr><td colspan="2">REQUEST NUMBER 2</td></tr>
</table>

Figure 3.2: The format for one acknowledgment in a control packet.

has been chosen due to the high speed of the network [47, 11]. In this implementation, the lowest two packet numbers that have not been received (RN and RN2) are sent back in each acknowledgment. The destination address is needed because of the broadcast nature of the control channel; all other stations simply ignore the acknowledgment.

In addition to the reservation commands and acknowledgments, there is a collision threshold bit, the number of arrivals in the last grand cycle (31 bits), and a CRC (32 bits). In this implementation, there are 5 reservation commands per packet (40 bytes total) and 10 acknowledgments per packet (120 bytes total). Since the control channel operates under TDMA, a source field is not needed. Therefore, the total length of a control packet is 168 bytes. It is assumed that the data channels operate at 1 Gb/s and that data packets are 500 bytes long. Thus, the control channel operates at $(168/500) \times 1$ Gb/s $= 336$ Mb/s.

3.4 Status Tables

Four status tables are required at each station to support the reservation and flow control scheme of DMACS. In the following description, k is the number of cycles in a grand cycle.

- **Reservation matrix** $S = [s_{ij}]_{kN \times C}$: $s_{ij} = 1$ indicates slot i, channel j is reserved for a circuit connection by another station. Note that slot i is the i^{th} slot in a grand cycle rather than a normal cycle.

- **Request matrix** $R = [r_{ij}]_{kN \times C}$: $r_{ij} = 1$ indicates slot i, channel j has been requested by this station to be reserved for a circuit connection. Note that for a given slot i there can only be one channel j such that $r_{ij} = 1$.

- **Circuit connection matrix** $U = [u_{ij}]_{kN \times C}$: $u_{ij} = 1$ indicates slot i, channel j is reserved by this station for a circuit connection. Note that for a given slot i there can only be one channel j such that $u_{ij} = 1$.

- **Collision threshold array** $T = [t_j]_C$: $t_j = 1$ indicates that the collision threshold has been exceeded for channel j during the previous grand cycle.

These tables are updated upon reception of each control packet on the control channel. In addition, the request matrix R is updated when reservation commands are sent. The number of arrivals at each station which are broadcast during the first cycle of each grand cycle are summed to determine if the arrival threshold has been exceeded during the previous grand cycle. A single flag, A, is set to indicate that the arrival threshold is exceeded.

3.5 Transmission Algorithm

The slot transmission algorithm is similar to DISA [18] with a few modifications due to the support of circuit connections and the use of separate arrival and collision thresholds.

- **Step 1**: Attempt to transmit a packet to a circuit destination if it exists.

- **Step 2**: If step 1 fails, attempt to transmit a packet to the TDMA destination if the channel is not reserved.

- **Step 3**: If step 2 fails and the arrival threshold has not been exceeded, search among the random destinations for a packet to transmit on a channel that is not reserved and where the collision threshold has not been exceeded.

A detailed description of the algorithm is given in the flowchart in Figure 3.3. Note that transmissions to circuit destinations are collisionless, but transmissions to TDMA destinations are not collisionless unless the arrival threshold has been exceeded or the collision threshold has been exceeded for that channel. If the arrival threshold has been exceeded, the next grand cycle is fixed TDMA for the entire system except for preemption due to circuit connections. If the collision threshold has been exceeded for a particular channel, the next grand cycle is fixed TDMA for that channel except for preemption due to circuit connections.

When transmitting to a TDMA or random destination, priority is given to class 2 traffic over class 3 traffic. The calculation of the TDMA destination and random destinations is shown below as well as in Figure 3.3. Let *id* be the source station number, i the slot number, and *tdma* the TDMA destination channel number. The TDMA destination is given by

$$tdma = (N + i - id) \bmod N \tag{3.1}$$

Let $c_{r,j}$ be the jth random destination checked, where $0 < j < C$. Setting $c_{r,0} = tdma$, the random destinations can be calculated by

$$c_{r,j} = (c_{r,j-1} + 1) \bmod C \tag{3.2}$$

These calculations are the same as in [18] except the first random destination checked is $c_{r,1}$ instead of $c_{r,0}$. If the number of stations N is greater than C, the TDMA destination does not exist when $tdma \geq C$. This means that only C of the N stations have a TDMA destination in each slot.

If a collision occurs for a class 2 packet during transmission, the class 2 queue for that destination enters the backoff state (B flag = 1). The queue remains in this state until a packet is successfully transmitted in a TDMA slot for that destination [18]. The TDMA slot in the last cycle of a grand cycle is guaranteed to not be reserved for a circuit connection.

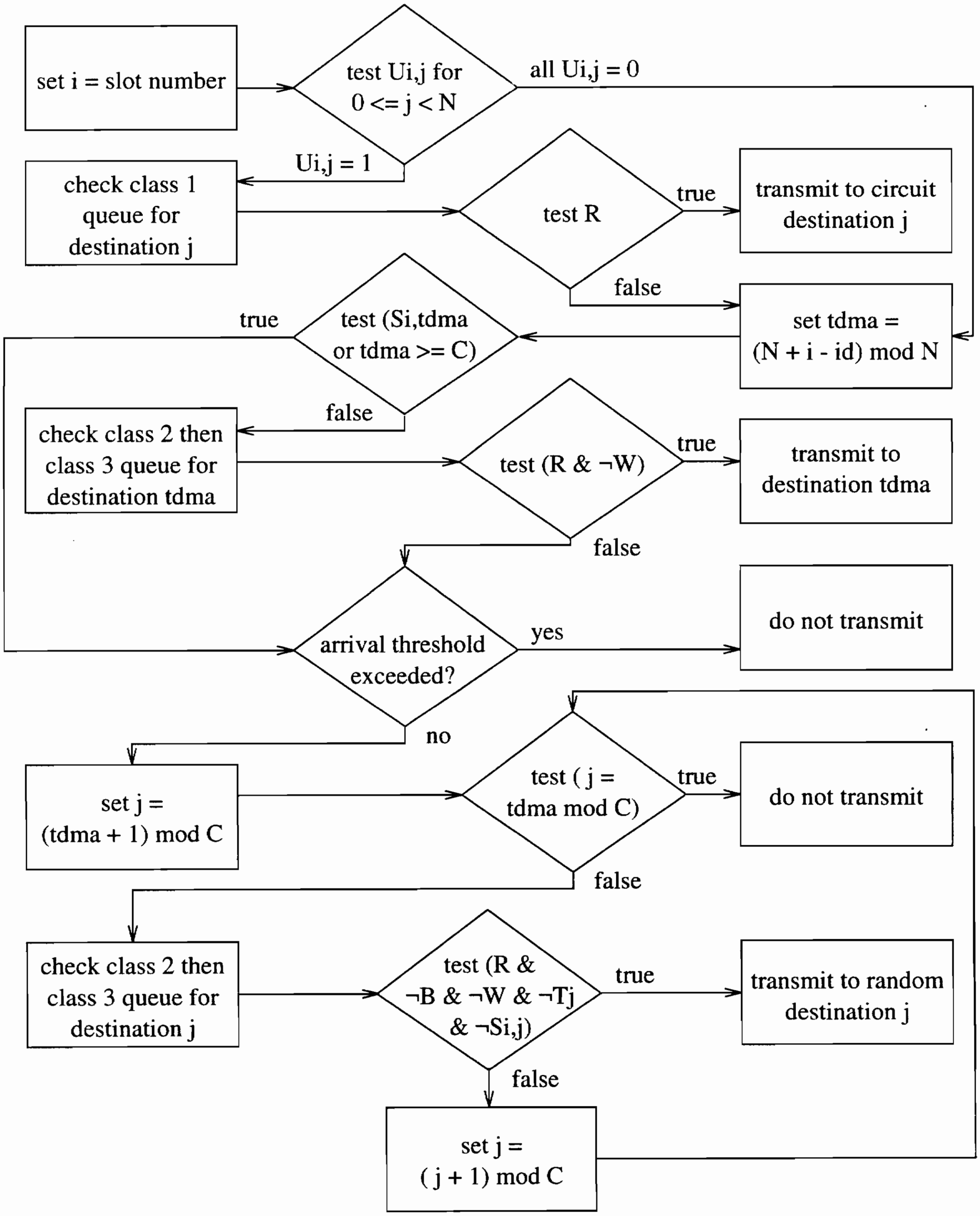


Figure 3.3: Flowchart describing DMACS transmission algorithm.

In this implementation, retransmissions have a higher priority than new packets to speed recovery from collisions.

3.6 Connection Setup and Tear Down

A class 1 connection is made by broadcasting a reservation request on the control channel for an unreserved slot. When the request comes back to the requesting station, collisionless transmission can begin in that slot if it has not been reserved by a request from another station during the delay due to transmission and propagation. If a request is unsuccessful, a request for another unreserved slot is broadcast. Depending on the amount of bandwidth needed, several reservation requests can be made. For a full-duplex connection, the other station also broadcasts one or more reservation requests. Connection tear down is accomplished by simply broadcasting reservation release commands on the control channel for reserved (slot, channel) pairs.

Setup for class 2 connections is accomplished by using a special START packet. Due to continuous acknowledgments on the control channel once a connection has been set up, a timer only needs to be set for the START packet. A free running timer is used to keep from retransmitting a packet before the round-trip propagation delay is over. This avoids unnecessary retransmissions caused by acknowledgments on the control channel. For simplicity, one set of sequence numbers is used for each source-destination pair in this implementation. Connection tear down uses a special END packet.

3.7 Acknowledgments

In DMACS, circuit connections are collisionless. They usually have real-time constraints, so retransmissions due to lost or errored packets are undesirable. This eliminates the need for acknowledgments for class 1 traffic. Class 2 and class 3 traffic is subject to collisions, so acknowledgments are necessary for reliable service. However, reliable service is optional. In this implementation, acknowledgments are not sent for class 3 traffic or class 2 traffic with real-time constraints.

The acknowledgments are sent in control packets as explained in Section 3.3. A selective repeat ARQ protocol is used with the lowest j packet numbers that have not been received being sent in each acknowledgment. This implementation uses a small value of j ($j = 2$) since the acknowledgments are guaranteed to occur at regular intervals on the control channel. The receiver sends acknowledgments for all active class 2 connections desiring reliable service. This is done in round-robin order in this implementation.

Unacknowledged packets in class 2 queues are marked as outstanding; transmission continues until m packets are marked as outstanding in a single queue, where m is the size of the sliding window. At this point, the W flag is set. If a retransmission is necessary, a collision is assumed to have occurred (as opposed to a lost or errored packet), and the queue enters the backoff state.

3.8 Example of Data Packet Transmission

An example of the destination queue checking sequence for $N = C = 4$ is given in Table 3.1. Three of the status tables for station 0 at a particular instant are shown in Table 3.2. The request matrix keeps track of outstanding reservation requests and does not affect the transmission algorithm, so it is not shown.

Given the status tables in Table 3.2, suppose that station 0 is executing the transmission algorithm for slot 0. No circuit destination exists since all entries in row 0 of U are 0. The TDMA destination for this slot is itself, but $S_{0,0} = 1$ which means that the channel is reserved. If the arrival threshold has not been exceeded, class 2 and class 3 queues for destinations 1 and 2 are searched sequentially for a ready packet. A class 2 queue that is

Table 3.1: Destination Queue Checking Sequence for $N = C = 4$

Slot	TDMA Destination	Random Destinations		
Station 0:				
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2
Station 1:				
0	3	0	1	2
1	0	1	2	3
2	1	2	3	0
3	2	3	0	1
Station 2:				
0	2	3	0	1
1	3	0	1	2
2	0	1	2	3
3	1	2	3	0
Station 3:				
0	1	2	3	0
1	2	3	0	1
2	3	0	1	2
3	0	1	2	3

Table 3.2: Reservation Matrix S, Circuit Connection Matrix U, and Collision Threshold Array T for Station 0 (Matrix rows are indexed by slot number and columns are indexed by channel number.)

S	0	1	2	3
0	1	0	0	0
1	0	0	0	0
2	0	1	0	1
3	1	1	1	0

U	0	1	2	3
0	0	0	0	0
1	0	0	1	0
2	0	0	1	0
3	0	0	0	0

slot	T
0	0
1	0
2	0
3	1

chosen must not have the backoff or waiting flag set. A ready packet in a class 2 queue has priority over a class 3 queue. No transmission can occur to random destination 3 because the collision threshold has been exceeded ($T_3 = 1$).

For slots 1 and 2, station 0 simply transmits a packet to circuit destination 2 assuming that a ready packet is waiting. If no packet is ready, it checks to see if the TDMA destination channel is reserved. For slot 1, no channels are reserved by another station. For slot 2, the TDMA destination (2) is not reserved. However, the only random destination that is not reserved is 0. As with slot 0, no transmissions are allowed to random destination 3 because the collision threshold has been exceeded.

For slot 3, there is no circuit destination and the TDMA destination channel is not reserved, so station 0 transmits a class 2 or class 3 packet to destination 3 if one is ready and the waiting flag is not set. Note that even though the collision threshold is exceeded, a transmission is allowed to destination 3 when it is the TDMA destination. If no packet is ready or W is set, and the arrival threshold has not been exceeded, the random destination queues are searched. However, in this case, all other channels are reserved, so no transmission can take place.

3.9 Possible Extensions

DMACS can operate with only a TT per station instead of a TT and FT. This would be an effective way to reduce the cost of a large network without greatly reducing performance. With this scheme, each cycle is composed of $N+1$ slots. Each station broadcasts its control packet on the control channel wavelength during a different slot using its TT. The control channel now has to operate at the same bit rate as the data channel. However, since control packets are much smaller than data packets, the idle time between them allows time for processing. The cost of a FT at every station has now been eliminated at the expense of using a data slot to transmit control information once per cycle. For a large cycle size, i.e. a large number of stations, this would not significantly reduce throughput.

DMACS can also be implemented with only a TR per station instead of two FRs. With one TR, control slots are alternated with data slots, so that each TR tunes back and forth between the control channel and its home channel in order to receive both control and data packets. If the control packets are small compared to data packets and tuning times are negligible, the loss in efficiency is minimal. However, this might not reduce costs due to the expense of a TR.

It may be necessary to operate with less than $N + 1$ channels because of the limited ranges of tunable lasers. If this is the case, home channels are shared between stations. Each station receives all traffic destined for the stations sharing its channel. The TDMA destination does not exist when $tdma \geq C$ (as defined in Section 3.5). This means that only C of the N stations have a TDMA destination in each slot. For instance, if there are twice as many stations as channels, $(N/2) + 1$ wavelengths are used, and only half the stations have a TDMA destination in each slot. All three of the above modifications can easily be combined.

An alternative to the acknowledgment scheme is to employ a TR at each station to follow the TT and listen for collisions on the data channels. This would allow detection of collisions after one roundtrip from the hub. In the initial version of the protocol, the control channel was used for acknowledgments because the increase in equipment cost would likely outweigh the increase in performance.

Chapter 4

Analytical Results

Precise analytic modeling of the protocol is extremely difficult due to its complexity. However, some simple, yet useful results can be derived. In Section 4.1, the blocking probability for class 1 traffic is presented. The remaining sections deal with the collision probability, throughput, and latency for class 2 and class 3 traffic.

4.1 Class 1 Blocking Probability

Class 1 traffic is connection-oriented with guaranteed bandwidth. A useful measure of performance for class 1 traffic is the probability that a new connection will be blocked due to the unavailability of adequate bandwidth. A single channel can be modeled as a group of $m = (k - 1)N$ "servers," where k is the number of cycles per grand cycle. Each server corresponds to reserving a single slot or $1/(kN)$ times the channel bandwidth. The amount of bandwidth corresponding to one slot varies with k and N, so the number of slots reserved must be scaled accordingly. For instance, if each connection needs two slots, then a channel corresponds to $m = (k - 1)N/2$ servers. Assuming Poisson arrivals with rate λ and exponentially distributed holding times with mean $1/\mu$, the model becomes an $M/M/m$ queuing system assuming that connection requests are queued (in an infinite buffer) and continuously try to find a free slot. To maintain independence between channels, all connections are assumed to be simplex. The probability that a new connection request finds all servers busy and must wait in queue is given by the *Erlang C formula* [47] which states that

$$P_Q = \frac{p_0(\rho)^m}{m!(1-\frac{\rho}{m})} \tag{4.1}$$

where m is the number of servers and $\rho = \lambda/\mu$. The probability of finding the system empty, p_0, is given by

$$p_0 = \left[\sum_{n=0}^{m-1} \frac{\rho^n}{n!} + \frac{\rho^m}{m!(1-\frac{\rho}{m})}\right]^{-1}$$

If a connection request that finds all slots full is lost (not reattempted), the model is an $M/M/m/m$ queuing system since there can be at most m customers in the system. The blocking probability in this case is given by the *Erlang B formula* [47] which states that

$$P_b = \frac{\frac{\rho^m}{m!}}{\sum_{n=0}^{m} \frac{\rho^n}{n!}} \tag{4.2}$$

This formula also holds for an $M/G/m/m$ system where the service time has an arbitrary probability distribution [47].

As an example, consider a network with 25 stations, $k = 5$, and 1 Gb/s data channels. Using the Erlang B formula of (4.2), the blocking probability for one channel is plotted in Figure 4.1 versus the offered connection traffic ρ in erlangs for three cases: each connection needs 8 Mb/s (1 slot), 16 Mb/s (2 slots), or 32 Mb/s (4 slots). These cases correspond to $m = 100$, $m = 50$, and $m = 25$, respectively. As expected, the case with $m = 100$ allows the lowest blocking probability for a given utilization. Note that for a fixed number of stations, k can be adjusted to provide the desired bandwidth per reserved slot and thus decrease the amount of wasted bandwidth in slots reserved for a connection.

4.2 Class 2 and 3 Collision Probability

Class 2 traffic is connection-oriented without guaranteed bandwidth, and class 3 corresponds to connectionless, datagram traffic. No collisions occur for packets from these classes if the arrival threshold or the collision threshold for the channel being used has been

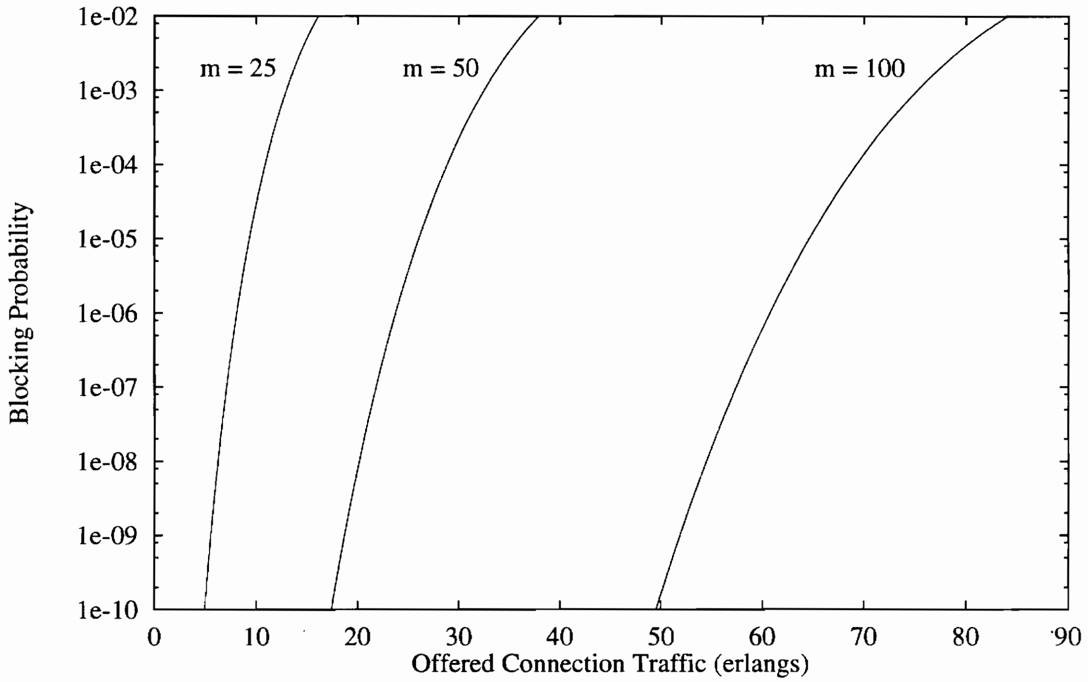


Figure 4.1: Class 1 blocking probability for one channel versus the offered connection traffic ρ in erlangs for a network with 20 stations, $k = 5$, and 1 Gb/s data channels. The bandwidth needs of each connection for the three cases are 8 Mb/s ($m = 100$), 16 Mb/s ($m = 50$) , and 32 Mb/s ($m = 25$).

exceeded during the previous grand cycle. However, collisions can occur when operating below these thresholds. In this section, a formula for the probability of collision given that a station is transmitting a class 2 or class 3 packet is derived.

Assume that a station transmits a class 2 or class 3 packet to the i^{th} destination checked in its destination queue checking sequence. Furthermore, assume that no thresholds are exceeded and no channels are reserved for class 1 traffic in the current slot. These assumptions imply that the packet is successful if no other station transmits to this destination. Thus, the probability of success is given by

$$P_s = \prod_{\substack{j=1 \\ j \neq i}}^{N} (1 - p_j) \tag{4.3}$$

where p_j is the probability that a station transmits to the j^{th} destination in its destination queue checking sequence. (For simplicity, it is assumed that a station might transmit to itself.) Let p_q be the probability that a packet is ready for transmission in a queue, independent of all other queues at that station.[1] Assuming that class 2 and class 3 traffic for the same destination shares a single queue and p_q is equal for all queues, then

$$p_j = (1 - p_q)^{j-1} p_q$$

Substituting into (4.3), the probability of success becomes

$$P_s = \prod_{\substack{j=1 \\ j \neq i}}^{N} [1 - (1 - p_q)^{j-1} p_q] \tag{4.4}$$

The probability of collision is given by

$$P_c = 1 - P_s \tag{4.5}$$

---

[1] In reality, the independence assumption is violated by the fact that all queues at a station share the same server and the service rate of one queue depends on the contents of the other queues.

P_c can now be calculated for a given value of p_q. However, a formula for P_c as a function of the total utilization due to class 2 and class 3 traffic is more meaningful. (A station is considered to be utilized during every slot in which it transmits a packet, regardless of the outcome of the transmission.) The probability that all queues at a station are empty can be calculated using the principle of inclusion and exclusion [60]. With N queues at each station, the probability that all queues are empty is

$$p_e = 1 - \begin{pmatrix} N \\ 1 \end{pmatrix} p_q + \begin{pmatrix} N \\ 2 \end{pmatrix} p_q^2 - \begin{pmatrix} N \\ 3 \end{pmatrix} p_q^3 + \cdots + (-1)^N p_q^N \tag{4.6}$$

The probability that a station has at least one packet to transmit is $1 - p_e$, so the total utilization due to class 2 and class 3 traffic is

$$\rho_{2,3} = N(1 - p_e) \tag{4.7}$$

Figure 4.2 is a plot of the TDMA destination ($i = 1$) collision probability versus $\rho_{2,3}$ for network sizes of 8, 32, and 128 stations.[2] For all three values of N, a low utilization is needed to keep the collision probability reasonably low. The point at which it becomes objectionable is where the thresholds should be set. In practice, stations will be restricted from transmitting to certain destinations when the corresponding queues have the backoff or waiting flag set. Thus, the number of collisions will be decreased. However, the reservation of (slot, channel) pairs for class 1 traffic will increase the number of collisions for a given amount of class 2 and class 3 traffic because of the reduced bandwidth.

4.3 Throughput

Assume that class 1 traffic has its maximum number of slots reserved, so one out of every k cycles in each grand cycle are available for class 2 or class 3 transmissions. Also,

---

[2] The collision probability is slightly greater for random destinations, i.e. it increases with increasing i.

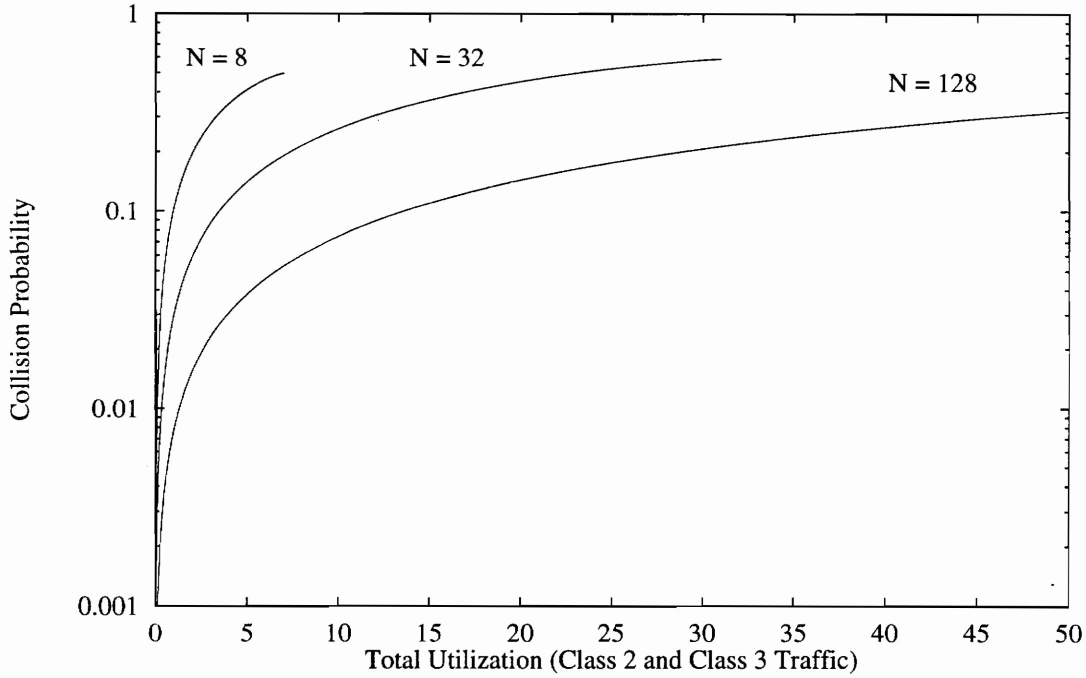


Figure 4.2: The TDMA destination collision probability as a function of the total utilization due to class 2 and class 3 traffic. Three different values for the number of stations N are shown: 8, 32, and 128.

assume that the arrival threshold has been exceeded, as it will be for high utilization. With these assumptions, an upper bound on the throughput for class 2 and class 3 traffic is

$$\frac{1}{k} \cdot \frac{N-1}{N} \quad \frac{packets}{slot \cdot channel} \tag{4.8}$$

where k is the number of cycles per grand cycle. Here it is assumed a station does not transmit to itself. The maximum overall throughput is

$$\frac{N-1}{N} \quad \frac{packets}{slot \cdot channel} \tag{4.9}$$

which equals 1 in the limit as $N \to \infty$. In Section 6.1, a realistic maximum for the overall throughput obtained through simulation is given.

4.4 Class 2 and 3 Latency

In this section, two cases for calculating the class 2 and class 3 latency are considered: (1) the arrival threshold is always exceeded, or (2) no thresholds (arrival or collision) are exceeded. In the first case, each channel can be treated as a TDMA system and thus the total latency can be calculated, but the second case is more complex, leading only to the calculation of the excess or surplus delay.

4.4.1 Arrival Threshold Exceeded

The total latency for class 2 and class 3 traffic can be calculated if it is assumed that the arrival threshold is always exceeded, i.e. traffic is heavy. In this case, each channel becomes a TDMA system. It is also assumed that all class 1 slots are reserved. First, the case where class 2 and class 3 have equal priorities (or alternatively, there is no class 3 traffic) is considered. After that, the case with different priorities is considered.

Single Priority System

Focusing on one channel, suppose there are N traffic streams of class 2 and class 3 traffic with a common destination arriving at each station according to a Poisson process with rate $\lambda/(kN)$ each. Packets are equal-length, and each slot is one time unit and can carry a single packet. The queue for a single stream can be modeled as an $M/D/1$ queuing system with vacations [47]. The service rate is $\mu = 1/(kN)$ and the utilization is $\rho = \lambda$. When there are no packets in a queue at the beginning of its assigned TDMA slot, the server takes a vacation for kN slots. The average waiting time in queue is given by

$$W = \frac{\lambda kN}{2(1-\lambda)} + \frac{kN}{2} = \frac{kN}{2(1-\lambda)}$$

Since the service time is one slot, the average total delay in slots becomes

$$T = \frac{kN}{2(1-\lambda)} + 1 + D_p \tag{4.10}$$

where D_p is the propagation delay. With the assumption that no class 1 slots are reserved, $\mu = 1/N$ and $\rho = \lambda/k$. The vacation interval is only N slots, so the average total delay becomes

$$T = \frac{N}{2(1-\frac{\lambda}{k})} + 1 + D_p \tag{4.11}$$

Nonpreemptive Priority System

Now consider the case where class 2 has higher priority than class 3 and each class has its own queue. Since a packet in transmission is not interrupted, this is a nonpreemptive priority system. The assumption that all class 1 slots are reserved is maintained. Suppose that each of the N traffic streams is split into class 2 and class 3 traffic such that the arrival rates of the two new Poisson streams are $n_2\lambda/(kN)$ and $n_3\lambda/(kN)$, respectively. Also, assume $0 \leq n_2, n_3 \leq 1$ and $n_2 + n_3 = 1$. The utilization for the two classes is now $\rho_2 = n_2\lambda$

and $\rho_3 = n_3\lambda$. The average waiting time in queue for the higher priority class, class 2, is given by [47]

$$W = \frac{\lambda kN}{2(1-n_2\lambda)} + \frac{kN}{2}$$

where the last term is due to the vacation interval. For class 3 traffic, the queuing delay that results from class 2 packets arriving while a class 3 packet is waiting in queue must be added to the queuing delay that results from packets already in queue. The average waiting time in queue for class 3 packets is

$$W = \frac{\lambda kN}{2(1-n_2\lambda)(1-n_2\lambda-n_3\lambda)} + \frac{kN}{2}$$

Adding the service time of one slot and the propagation delay, the average total delay for class 2 packets becomes

$$T = \frac{\lambda kN}{2(1-n_2\lambda)} + \frac{kN}{2} + 1 + D_p \tag{4.12}$$

Similarly, for class 3 packets, the average total delay is

$$T = \frac{\lambda kN}{2(1-n_2\lambda)(1-n_2\lambda-n_3\lambda)} + \frac{kN}{2} + 1 + D_p \tag{4.13}$$

4.4.2 No Thresholds Exceeded

In this section, a formula for the excess or surplus delay for reliable class 2 and class 3 traffic using the selective repeat ARQ protocol is given. This delay is the time from the first transmission attempt to the successful transmission of a packet. It is zero if no collisions or errors occur. The average waiting time in queue for a selective repeat ARQ protocol is calculated in [61], but the system under consideration violates the assumptions of that model. When operating below the threshold values, each queue cannot be treated separately due to the effect of other queue contents on the service rate of a single queue. For simplicity, it is assumed that class 2 and class 3 have equal priorities and share the same queue. Alternatively, it could be assumed that only class 2 is reliable and class 3 has a lower priority. In that case, the delay calculation would only apply to class 2 traffic.

Let P_c be the probability of collision or error, and assume that it is the same when transmitting to any TDMA or random destination. Then r is the number of retransmissions with probability

$$(1 - P_c)(P_c)^r$$

The value of r is the number of transmissions before the first successful transmission where the probability of success for each transmission is $1 - P_c$. This value has a negative binomial distribution [62], so the mean of r equals

$$\frac{P_c}{1 - P_c}$$

The delay from a transmission to a retransmission has three components:

1. The round-trip propagation delay (D_{p_r})
2. The delay from reception to transmission of an acknowledgment (D_a)
3. The delay from reception of the acknowledgment to the retransmission (D_t)

The acknowledgment on the control channel is guaranteed to be collision-free. Ignoring errors and assuming that an acknowledgment for the sender can always be sent in the next control packet, D_a is simply the time to transmission of the receiving station's next control packet which is at most N slots including the transmission time. When the acknowledgment is received, the queue goes into backoff if a retransmission is necessary. This means that the next transmission to this destination occurs in the slot where it is the TDMA destination. Assuming retransmissions are given higher priority than new packets and the retransmission queue is empty, D_t is the time until reaching the TDMA slot for this destination which is at most kN slots including the transmission time. (This assumes that all class 1 slots are reserved. With no class 1 slots reserved, D_t is at most N.) Therefore, the average excess delay is given by

$$D_e = \frac{P_c}{1 - P_c}(D_{p_r} + D_a + D_t) \tag{4.14}$$

Using the maximum values of D_a and D_t for each retransmission, the average excess delay becomes

$$D_e = \frac{P_c}{1 - P_c}[D_{p_r} + (k + 1)N] \tag{4.15}$$

Figure 4.3 is a plot of D_e as a function of the probability of collision or error, P_c, using (4.15). Three different values for the number of stations N are shown: 8, 32, and 128. The round-trip propagation delay D_{p_r} has been set at 40 slots which corresponds to a network with all stations 8 km from the hub, 1 Gb/s data channels, and 500-byte data packets. Also, $k = 5$ has been used. Since D_e is proportional to N, it becomes excessive at smaller values of P_c as N increases. However, the values of P_c for the expected range of operation (see Figure 4.2) lead to reasonably low excess delays.

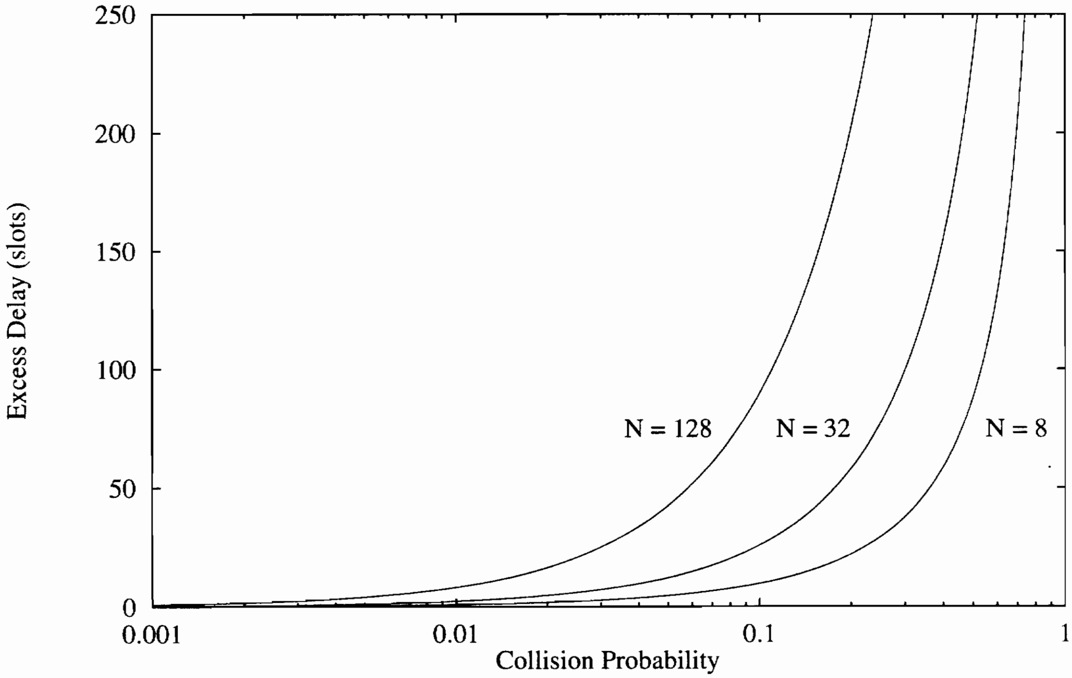


Figure 4.3: The average class 2 and class 3 excess delay as a function of P_c, the probability of collision or error, for $N = 8$, $N = 32$, and $N = 128$. The maximum values of $D_a = N$ and $D_t = kN$ are used for each retransmission. The round-trip propagation delay D_{p_r} is 40 slots and $k = 5$.

Chapter 5

Simulation Model

This chapter discusses the simulation model that has been developed for the DMACS and DISA protocols. A discrete-event simulation program has been written in C++. An overview of the program structure is given in Section 5.5. First, the performance metrics used for evaluation, the system parameters that affect the simulation, the workload developed for the model, and the methods used for transient removal and stopping criteria are discussed.

5.1 Performance Metrics

The performance of the DMACS and DISA protocols is evaluated in terms of the throughput in packets/(slot × channel)[1] and the mean packet latency. In addition to the overall throughput and latency, the metrics are calculated separately for class 1, class 2, and class 3 traffic.

5.2 System Parameters

The following parameters affect the performance of the network. The values for fixed parameters are given in parentheses. If a parameter affects only DMACS or DISA, the affected protocol is listed in brackets.

---

[1] In a slotted single-channel network, throughput would be expressed in packets/slot. Due to the multi-channel environment, division by the number of channels is used to normalize throughput to a value between 0 and 1.

- Speed of data channels (1 Gb/s)
- N, number of stations
- Percentage of stations designated as servers (25%)
- C, number of data channels (= number of stations)
- Tuning time (0)
- Distance from hub (8 km for each station)
- Data packet size (500 bytes)
- Maximum queue size (1024)
- Window size for reliable class 2 traffic (128)
- Maximum reservation commands per control packet (5) [DMACS]
- Maximum acknowledgments per control packet (10) [DMACS]
- Retransmission timer value (5 × propagation delay) [DISA]
- Piggyback timer value (propagation delay) [DISA]
- Collision threshold [DMACS]
- Arrival threshold [DMACS]
- Cycles per grand cycle (5) [DMACS]
- Flow control threshold ($3C/8$) [DISA]
- Error rate (0)

The speed of the data channels is chosen to be 1 Gb/s because of electronic limitations [16]. The designation of a subset of the stations as servers is used when simulating a non-uniform traffic pattern. The differences between uniform and non-uniform traffic are discussed in Section 5.3. The impact of having fewer data channels than stations is not studied. Tuning time is assumed to be zero because it is not known at this time what the magnitude will be and a significant tuning time would simply decrease throughput. Using two tunable transmitters could eliminate the performance penalty by overlapping tuning with transmission. The 8 km distance from the hub is equivalent to that of a large LAN with uniformity assumed for simplicity. A data packet size of 500 bytes is chosen as a compromise between the large percentage of small packets in a current typical LAN application [63] and the throughput decrease due to tuning time. With 500 byte packets and 1 Gb/s channels, the transmission time for a single packet (slot length) is 4 microseconds. If very fast tunable transmitters can be fabricated, a smaller packet size would be desirable to decrease waste within slots and decrease the cycle time. It is also desirable to have much smaller packets for voice applications [47]. However, the speed of the data channels is fast enough to meet the necessary timing requirements for voice. The bandwidth wasted by transmitted packets smaller than 500 bytes is ignored in the simulation. A maximum queue size is imposed due to memory limitations. Packets arriving when the queue is full are simply discarded. Discarding does not become significant until extremely high arrival rates are simulated. The window size for reliable class 2 traffic is chosen to be rather large (128 packets) to limit its effect on performance. It is fixed during all simulations for simplicity.

The retransmission timer and piggyback timer are part of the acknowledgment scheme of DISA. In the original description of the DISA protocol, acknowledgments were done through slot extension [18]. This scheme is not practical in a LAN/MAN due to the long propagation delay. Using a suggestion in [18], acknowledgments in the implementation of

DISA evaluated here are piggybacked onto data packets. The piggyback timer is set at the receiver if no packet is available for piggybacking. If no packet is sent before the timer expires, an empty data packet is sent with the acknowledgment. The retransmission timer is the normal timer set at the sender. The selective repeat ARQ protocol used in DMACS is also used in DISA for a fair comparison. As in DMACS, each acknowledgment carries the lowest two packet numbers that have not been received. DISA uses a single local flow control threshold based on the number of arrivals plus collisions per cycle. If the threshold is exceeded, no transmissions to random destinations are allowed in the next cycle. The value of $3C/8$, the optimum value suggested in [18], is used. A collision is assumed to have occurred if a retransmission is necessary. In the implementation of DISA studied here, one timer per destination is used, and acknowledgments are only piggybacked on class 2 traffic.

Finally, the error rate of the network is assumed to be zero. Therefore, no packets are lost or corrupted. Actual bit error rates of optical-fiber transmission systems are extremely low, typically ranging from 10^{-9} down to 10^{-15} [16].

5.3 Workload Description

A workload for the simulation program has been developed based on the traffic expected in a future high-speed LAN/MAN [14]. The specifications for class 1, class 2, and class 3 traffic are given in the following sections along with the types of traffic modeled.

5.3.1 Class 1

Class 1 traffic is connection-oriented with guaranteed bandwidth. Three types of class 1 connections are modeled as shown in Table 5.1. Video teleconferencing is a full-duplex connection, while the other two are simplex connections. A high-speed data connection is intended to model interprocessor communication for a parallel computation or the transfer

Table 5.1: Class 1 Connection Types and Bit Rates

Application	Bit rate
Video teleconferencing	2 Mb/s (1 Mb/s each way) [64]
High-quality video	100 Mb/s [14, 65]
High-speed data	200 Mb/s

of computation results from a server to a workstation [14]. Packets are generated at constant intervals to attain the specified bit rate. A uniform distribution is used for determining the type of connection. The mean holding time is 15 minutes and is assumed to have a two-stage hyperexponential distribution with a standard deviation of 16.67 minutes. This allows some long connections, but the majority are relatively short as expected in a LAN/MAN. For the case of non-uniform traffic, high-quality video connections occur only from a server to a workstation. High-speed data connections can go in either direction but occur only between a server and a workstation or a server and a server. The source of the data is a server 80 percent of the time and a workstation 20 percent of the time. Video teleconferencing connections have a uniform source-destination distribution throughout all simulations.

Since connections are not blocked in the simulation when the required bandwidth is not available, the connection bit rate is adjusted based on the number of reserved slots. In addition, packets are not generated until after the first slot is reserved. For both class 1 and class 2 traffic, the time between the start of successive connections is assumed to be constant. It is calculated based on the overall arrival rate desired. Note that the actual time simulated is on the order of seconds, as discussed in Section 5.4, which reduces the sensitivity of results to a constant time between the start of connections and the holding time parameters.

Table 5.2: Class 2 Connection Specifications

Parameter	Voice	File transfer
Mean train size in packets (geometric)	19	33 (reply only)
Mean time from reply to request (exponential)	–	2 ms
Activity ratio (% of time in ON state)	0.8	0.2
Peak bit rate	64 Kb/s	50 Mb/s
Average bit rate	51.2 Kb/s	10.3 Mb/s
Mean holding time (hyperexponential)	10 minutes	
Standard deviation of holding time	20 minutes	

5.3.2 Class 2

Class 2 traffic is connection-oriented without guaranteed bandwidth. Two models have been chosen for class 2 connections: file transfer and variable bit rate voice. The models are based on the concept of packet trains [66]. In a voice conversation, the user switches between a talking and a listening state, so voice has been modeled as an ON/OFF source [64, 67]. The detailed specifications for a voice with silence detection model are given in [64]. The voice model can also be used as an aggregate of remote login, electronic mail, and database access connections. Packet trains have been shown to exist for TCP connections carrying those types of traffic [66]. The file transfer model is based on the one given in [67] with an average page size of 16 Kbytes. A peak rate of 50 Mb/s has been chosen as a high estimate, and the train interarrival time is large enough to allow for memory access delay. Both types of connections are full-duplex, so source locality [66] is modeled appropriately. The detailed specifications for each type of connection are given in Table 5.2. Where applicable, the random distribution used for sampling is given in parentheses. All other parameters for the connections can be derived from the ones shown.

The direction of trains alternates to model request-reply chains in file transfers [67] and talking back and forth in voice connections. The file transfer request always has a train size of one packet; the mean train size of 33 packets, shown in Table 5.2, is for the reply only.

The activity ratio and the peak bit rate for file transfer applies only to the actual transfer of file data. The average bit rate includes the request packets. Car interarrival times (between packets in a train) are constant, and train interarrival times are sampled from an exponential distribution. The use of constant car interarrival times is a simplification given in [66], but it is likely that packets from an active source are generated at constant time intervals [68]. The mean holding time of 10 minutes is shorter than a class 1 connection. It is used on the basis of measurements of TCP connections in a LAN [69]. The standard deviation used for the two-stage hyperexponential distribution is greater than class 1 because a larger number of short connections are expected [69]. A uniform distribution is used for determining the type of connection. For the case of non-uniform traffic, file transfers always occur between a workstation and a server. The source of the transfer is a server 80 percent of the time and a workstation 20 percent of the time. Voice connections have a uniform source-destination distribution throughout all simulations.

Voice connections are not made reliable because it is not desirable to retransmit lost or corrupted packets due to delay requirements. Therefore, acknowledgments are only needed for file transfer connections. To model an actual connection, an entire train must be received before the next train in the other direction is generated.

5.3.3 Class 3

Class 3 corresponds to connectionless, datagram traffic. It is expected to consist of control messages and short transactions, so it is modeled by a Bernoulli distribution. For each station, the probability of generating a packet in a slot is determined by the mean class 3 arrival rate. Class 3 traffic has a uniform source-destination distribution throughout all simulations.

5.4 Transient Removal and Stopping Criteria

The simulation of the network and protocol operation is a *steady-state simulation* since the two measures of performance being estimated, throughput and mean packet latency, are defined as a limit as the length of the simulation goes to infinity [70]. Since the system begins in an empty state, the *initial transient* before reaching "steady-state behavior" should be removed in order to not bias the estimations.

To produce an average number of connections in the network, each simulation run begins by generating the connections that would exist at 15 minutes (the longer of the mean holding times for connections) had the simulation begun at time 0. However, no slots are reserved for connections and all queues are empty, so it is still necessary to remove the initial transient. This is accomplished through *initial data deletion* [70, 71]. Initially, a test developed by Schruben [72] was used to detect any remaining initialization bias, but the test did not seem to have much effect in this case, so a "warm up" period of one second was used for all simulations. The length of the transient does depend on the input parameters such as arrival rate, but one second is long enough to discard a very large number of observations. In addition, the stopping criteria, which is discussed below, minimizes the effect of any remaining transient.

To decide when to stop the simulation, the simulation runs use a sequential procedure proposed by Adam [73] as modified by MacDougall [71]. The procedure is based on the method of batch means. The modification by MacDougall is a final reduction to 30 or fewer batches to improve coverage of the mean as recommended by Schmeiser [74]. Adam's procedure is used to separately calculate 90 percent confidence intervals for the overall packet latency, the class 1 latency, the class 2 latency, and the class 3 latency. Since the distribution of connections in the network greatly affects the latencies and the amount of simulated time is on the order of seconds, it is necessary to perform *replications* [70] in addition to using batch means to achieve accurate results. For each simulation, a minimum of 5 replications is performed with Adam's procedure being used for each replication. After each replication, the program resets itself to the initial empty state and continues with the next replication. New 90 percent confidence intervals are constructed using the results from each replication. The simulation is stopped when the relative half-widths for the overall, class 2, and class 3 latencies are 10 percent, 15 percent, and 20 percent respectively. (The class 1 latency does not vary much due to the nature of the traffic.) A maximum of 30 replications is imposed for practical reasons. In addition, each replication is run for a minimum of one second past the warm up period and a maximum of 10 seconds. In summary, the final stopping criteria method is a combination of Adam's batch means procedure and the replication-deletion method [75].

5.5 Program Description

The discrete-event simulation program consists of approximately 4,000 lines of C++ code. The g++ 2.5.7 compiler from the GNU project was used, and all simulations were run on either a DECstation 5000/25, a DECstation 5000/125, or a DECstation 3100 running ULTRIX 4.3A. The command-line options for specifying the input parameters are given in Appendix A. The random number generator and random variate generation routines along with routines to calculate quantiles of the normal and t distributions have been borrowed from MacDougall's smpl system [71]. Additional seeds for the generator spaced 100,000 samples apart are taken from [76]. Separate, nonoverlapping streams are used for the 17 different random variables in the program instead of subdividing one stream [62]. Dynamic memory management is performed using a Pool class given in [77] to improve performance over the standard `new` and `delete`. A Queue class and a BitVector class have been borrowed from Lippman [78].

The events that can occur in the program are as follows:

- **Class1**: queues a packet for a class 1 connection at the source station and schedules the next Class1 event for this connection. If this is the beginning of a connection, a start request is queued, and the next connection is scheduled.
- **Class2**: queues a packet for a class 2 connection at the source station and schedules the next Class2 event for this connection. If this is the beginning of a connection, the next connection is also scheduled.
- **Class3**: generates class 3 packets.
- **TransmitData**: gets data packets from all stations and sends them to the appropriate channels.
- **TransmitCtrl**: gets a control packet from the station whose turn it is to transmit on the control channel and sends it to the control channel.
- **ReceiveData**: gets data packets from all the channels and sends them to the appropriate stations.
- **ReceiveCtrl**: gets a control packet from the control channel and sends it to all stations.
- **SenderTimeout**: signals the occurrence of a retransmission timeout to the appropriate station.
- **RecvrTimeout**: signals the occurrence of a piggyback timeout to the appropriate station.

The main loop of the program gets the next event from the event list and calls the routine for that event. The event list has been implemented as a heap [79] instead of a linked-list to improve performance. The event-list size is usually quite large since it is proportional to the number of connections in the network.

The following classes are the major ones of interest in the program. Base classes, if any, are given in brackets after the class name.

- **Station**: implements the functions of one station: transmitting and receiving data and control packets, queuing data packets, managing the status tables, etc.
- **Queue2**: implements the special features of a class 2 queue. Unacknowledged packets desiring reliable delivery are saved and put in the queue for retransmission if necessary.
- **SelRepARQSender**: used by Queue2 to manage the sliding window for the selective repeat ARQ protocol.
- **SelRepARQRecvr**: used by Station to manage the receive buffer for the selective repeat ARQ protocol.
- **DataPkt**: base class that contains fields common to all data packets. It is used directly for class 3 packets.
- **DataPkt1** [DataPkt]: adds a pointer to a CC class object (see below) that is used for connection management.
- **DataPkt2** [DataPkt]: adds a sequence number, request numbers for acknowledgments, and other fields needed for a class 2 data packet.
- **CtrlPkt**: used by Station to construct a control packet. The Res and Ack classes are used for reservation commands and acknowledgments, respectively.
- **Token**: base class used to pass information when scheduling an event. It is used directly to store a sequence number for the SenderTimeout event.

- **Connection** [Token]: adds a holding time field and functions common to both circuit (class 1) and virtual circuit (class 2) connections.
- **CC** [Connection]: stores state information and calculates the time between packets for a circuit connection.
- **VC** [Connection]: stores state information and calculates the time between packets for virtual circuit connections. Trains are generated, and the direction of data flow in the connection is reversed when necessary.
- **Bma**: implements the batch means analysis procedure.
- **Events**: manages the event list.

A few template classes are used in the program: PriorityQueue (an abstract base class), Heap (derived from PriorityQueue), Queue, and Channel. The Channel class is used to implement both the data and control channels.

5.6 Model Validation and Verification

The two steps in measuring the goodness of a simulation model are *validation* and *verification* [62]. Validation ensures that the assumptions of the model are reasonable, and verification ensures that the model implements the assumptions correctly. Due to the unavailability of real-system measurements or applicable theoretical results, validation has been performed solely through intuition. The assumptions used in developing the model as well as the input parameter values and distributions all seem reasonable. They are based on measurements of existing similar traffic on systems, where possible. The output values produced by the configurations simulated are consistent.

To verify the program, separate programs were written to test the BitVector, Queue, Channel, Distribution, RandomInt, Bernoulli, Geometric, Exponential, Hyperexp, SampleStat, and Bma classes. The Bma class was tested using an $M/M/1$ queuing system. Tracing was used to verify proper event occurrences and memory management. Degeneracy tests were performed with two stations and only one class of traffic. The first phase of experiments described in Section 6.1 doubled as a continuity test, and seed independence was verified through the repetition of experiments with different seeds for each random number stream. Antibugging checks, additional checks that will print a message if an error occurs, are scattered throughout the program as an extra precaution.

Chapter 6

Simulation Results and Analysis

The experimental design was divided into three phases. Phase I determined optimum threshold values and maximum throughput for the DMACS protocol. Phase II studied the effects of the number of stations, the mean packet arrival rate, the traffic distribution, and the class distribution on the DMACS protocol performance. Phase III compared the performance of the DMACS and DISA protocols. A total of 312 simulation runs were performed.

6.1 Phase I

The configuration for phase I is a 16-station network with a class distribution of 80 percent class 1, 12.5 percent class 2, and 7.5 percent class 3 traffic. This traffic mix was chosen because it is anticipated that most of the traffic in a future high-speed network will be connection-oriented and class 2 connections will be more numerous but use less bandwidth than class 1 connections. On the average, three times as many class 2 connections as class 1 connections are created with this mix. For example, with an arrival rate of 0.5 packets/(slot × channel), there is an average of four class 1 connections per station and 12 class 2 connections per station. Both uniform and non-uniform traffic distributions were simulated with the mean packet arrival rate varied from 0.1 to 0.9 packets/(slot × channel) in increments of 0.1. Three cases were initially chosen to determine at what point the thresholds should be applied: (1) both thresholds off, (2) arrival threshold = 0 and collision threshold off, and (3) collision threshold = 2 and arrival threshold off. With the arrival threshold (AT) at 0, the system becomes fixed TDMA on every channel except for preemption due to circuit connections. A collision threshold (CT) of 0 would achieve the same effect, so a low value was chosen to quantify its effect on a channel by channel basis.

The results for throughput and mean packet latency are shown in Figures 6.1 through 6.8. Figures 6.1 through 6.4 show throughput versus arrival rate, while Figures 6.5 through 6.8 show latency versus throughput. Note that the scales for arrival rate and throughput change from the overall graphs to the class-specific graphs because only a fraction of the traffic is considered in the class-specific graphs. Each point marked with a symbol corresponds to the average of three simulation runs using different seeds. Performing three simulations for the same input parameter values decreases the effect of random sampling variation and provides smoother curves. Even though arrival rates up to 0.9 were used as the offered loads, the accepted loads in the simulations were less due to blocking and reduced bit rates for class 1 connections and inhibited packet train generation for class 2 connections. The accepted loads are the arrival rates shown in Figures 6.1 through 6.4, with each marked point corresponding to an overall offered load from 0.1 to 0.9.

The fourth line on the plots corresponds to the arrival threshold value used in phases II and III. This value is 0.25 packets/(slot × channel) or an actual threshold of 320 packets/grand cycle for a 16-station network. No collision threshold was used in phases II and III. When the results are distinguishable, the AT = 320 curve shifts from the threshold-off curve to the AT = 0 curve as expected. (The small differences at higher throughputs are due to random sampling variation.) The delay advantage of AT = 320 over AT = 0 at low loads is more pronounced with non-uniform traffic because a TDMA system tends to perform better with a regular traffic pattern. Note that there is not a single optimum value for the arrival threshold. The value used depends on the traffic in the network and

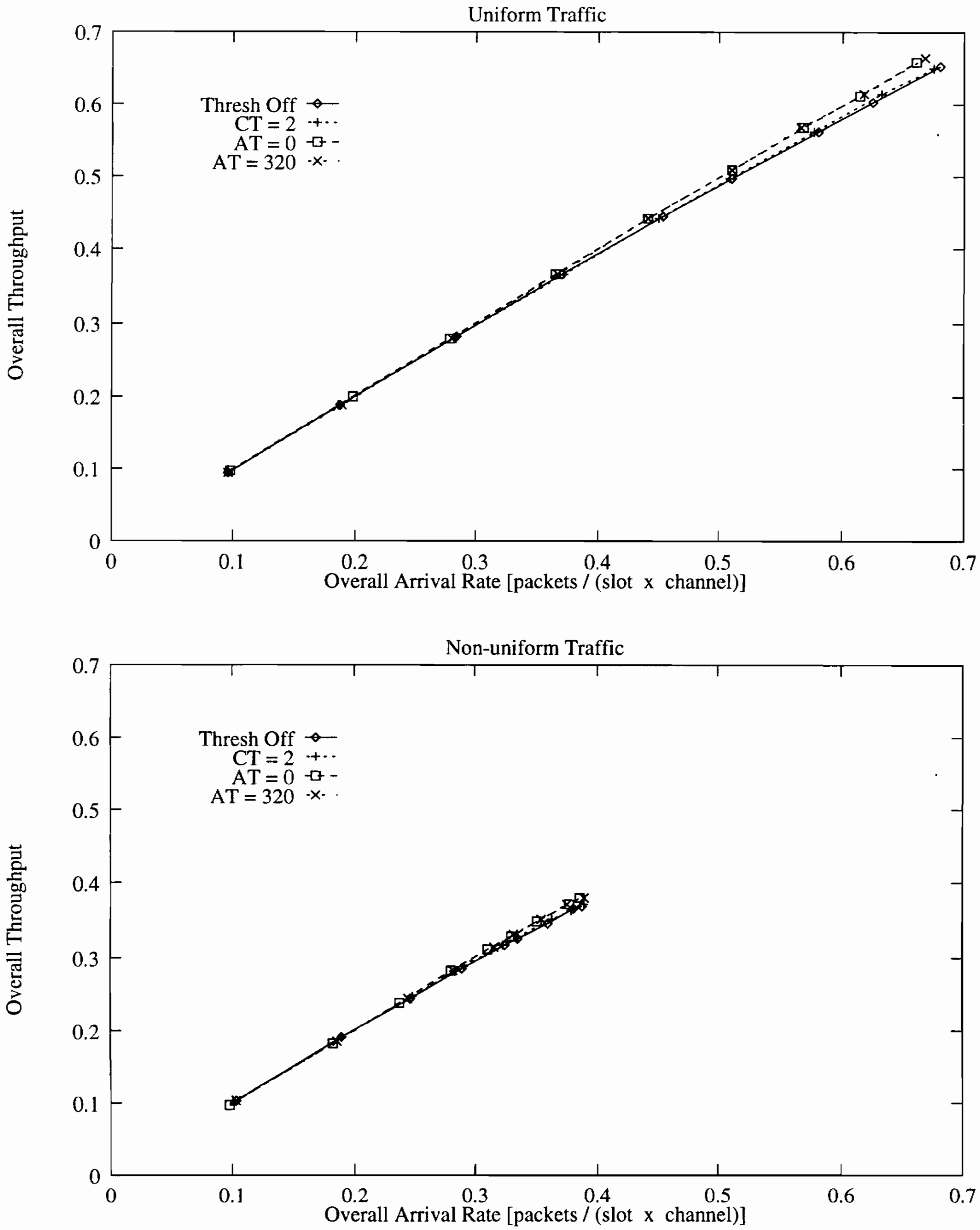


Figure 6.1: Phase I — Overall throughput with uniform and non-uniform traffic.

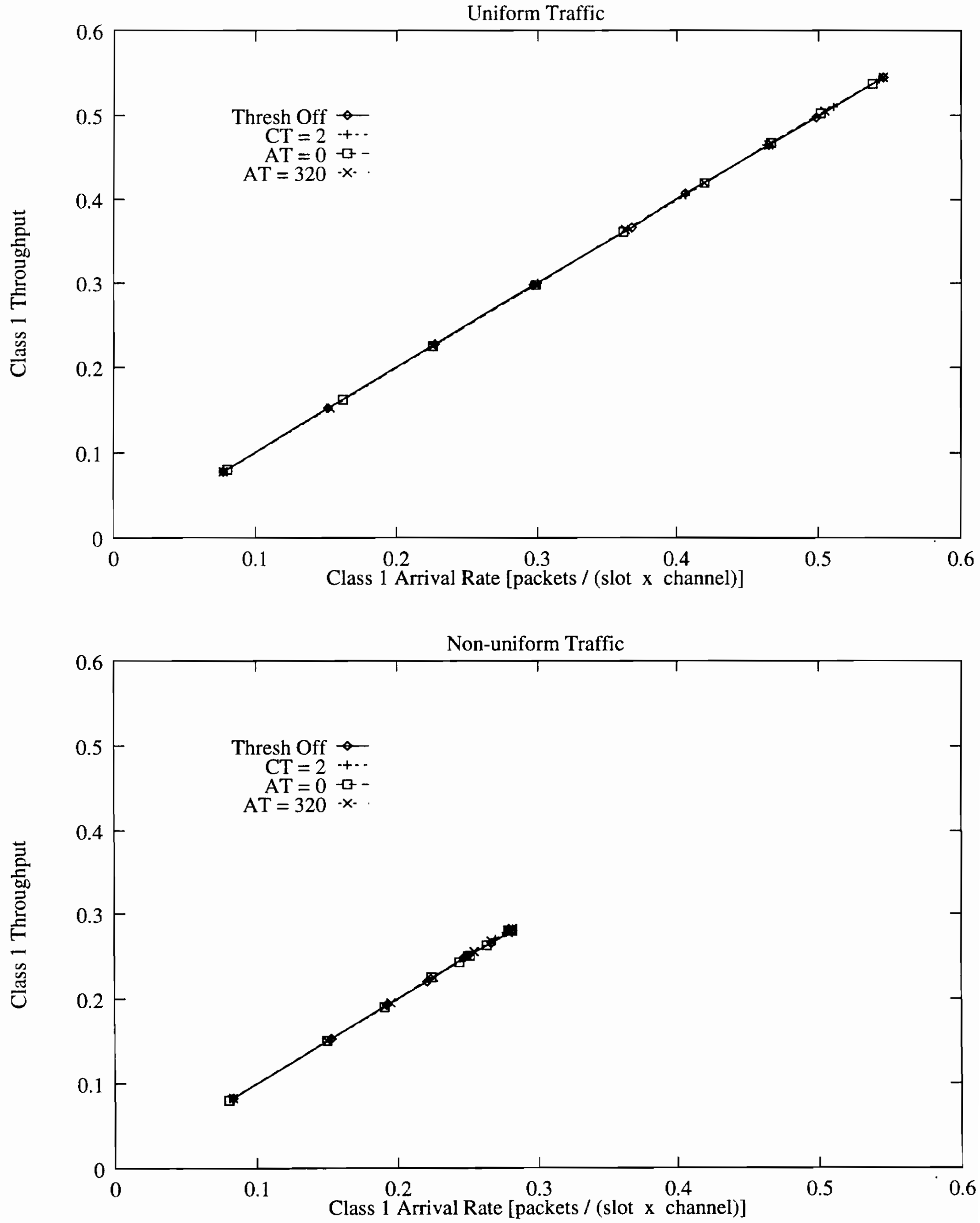


Figure 6.2: Phase I — Class 1 throughput with uniform and non-uniform traffic.

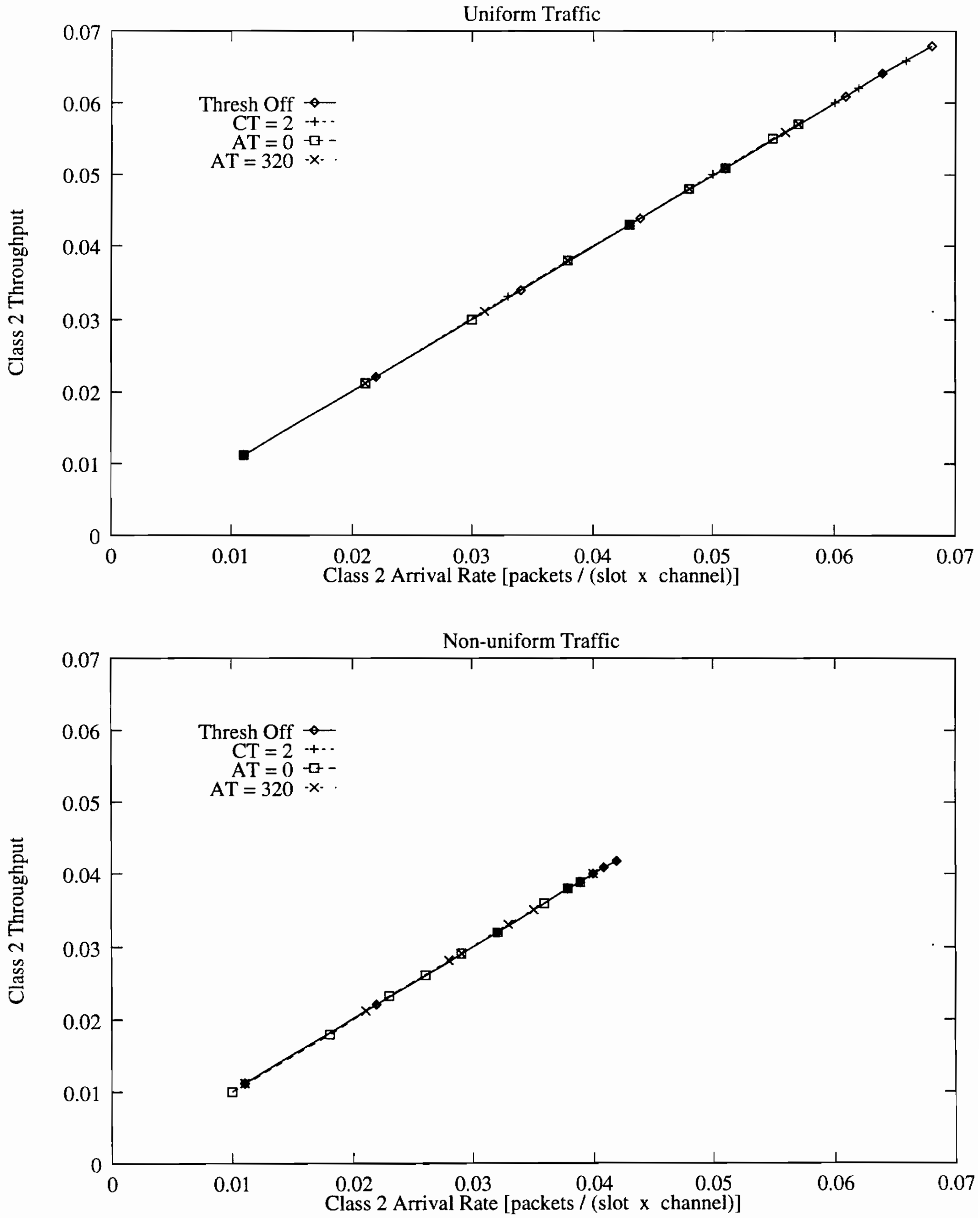
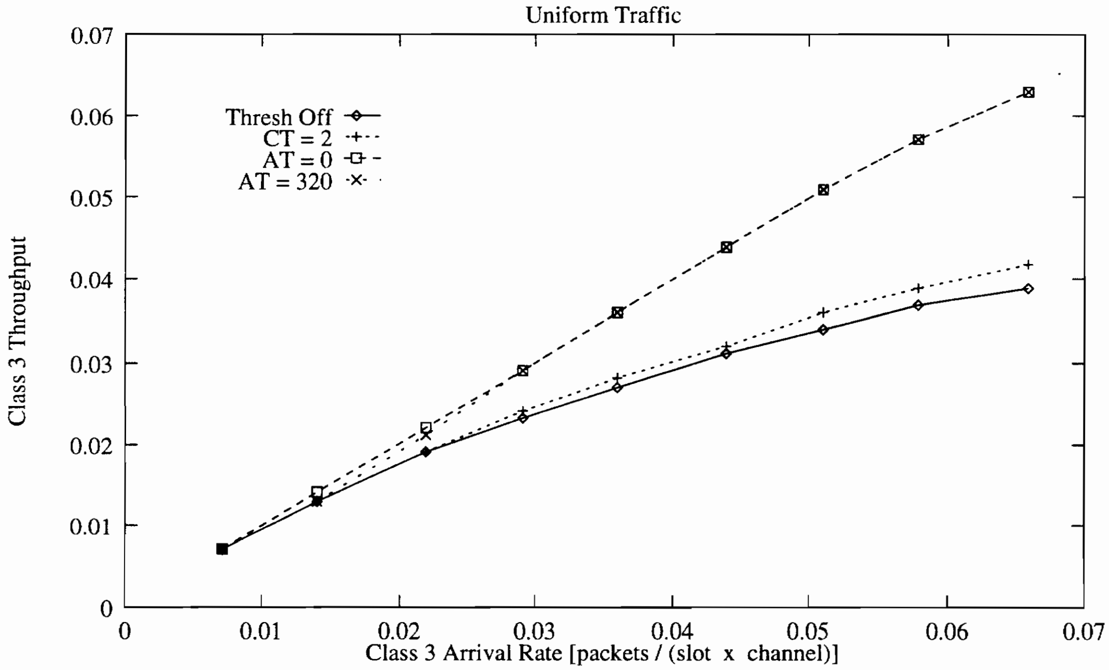


Figure 6.3: Phase I — Class 2 throughput with uniform and non-uniform traffic.



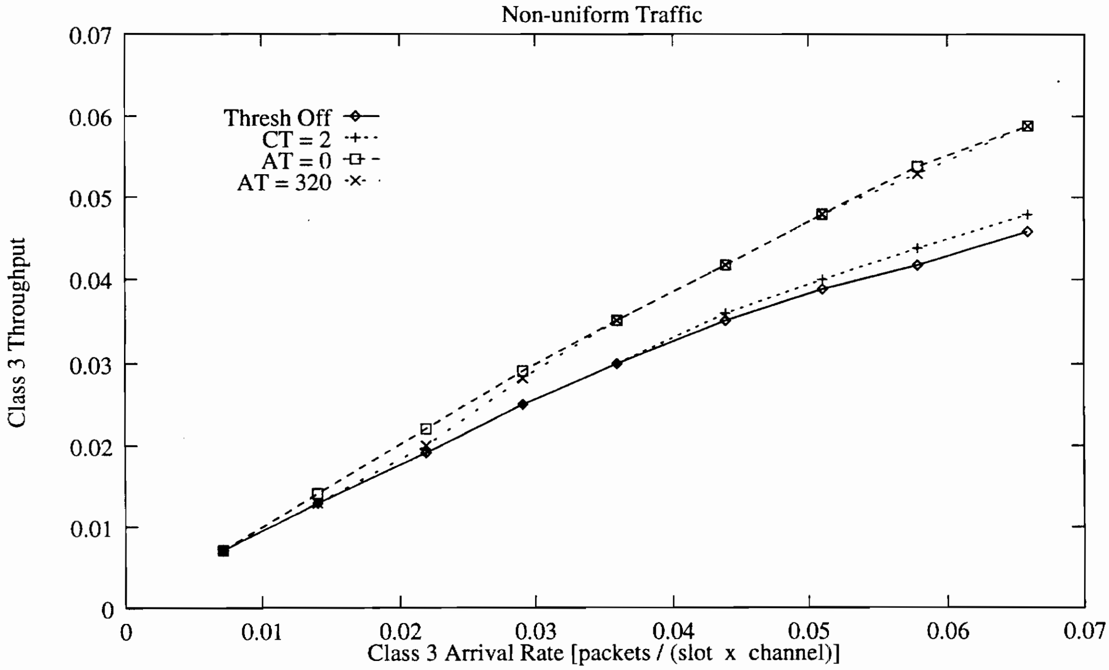


Figure 6.4: Phase I — Class 3 throughput with uniform and non-uniform traffic.

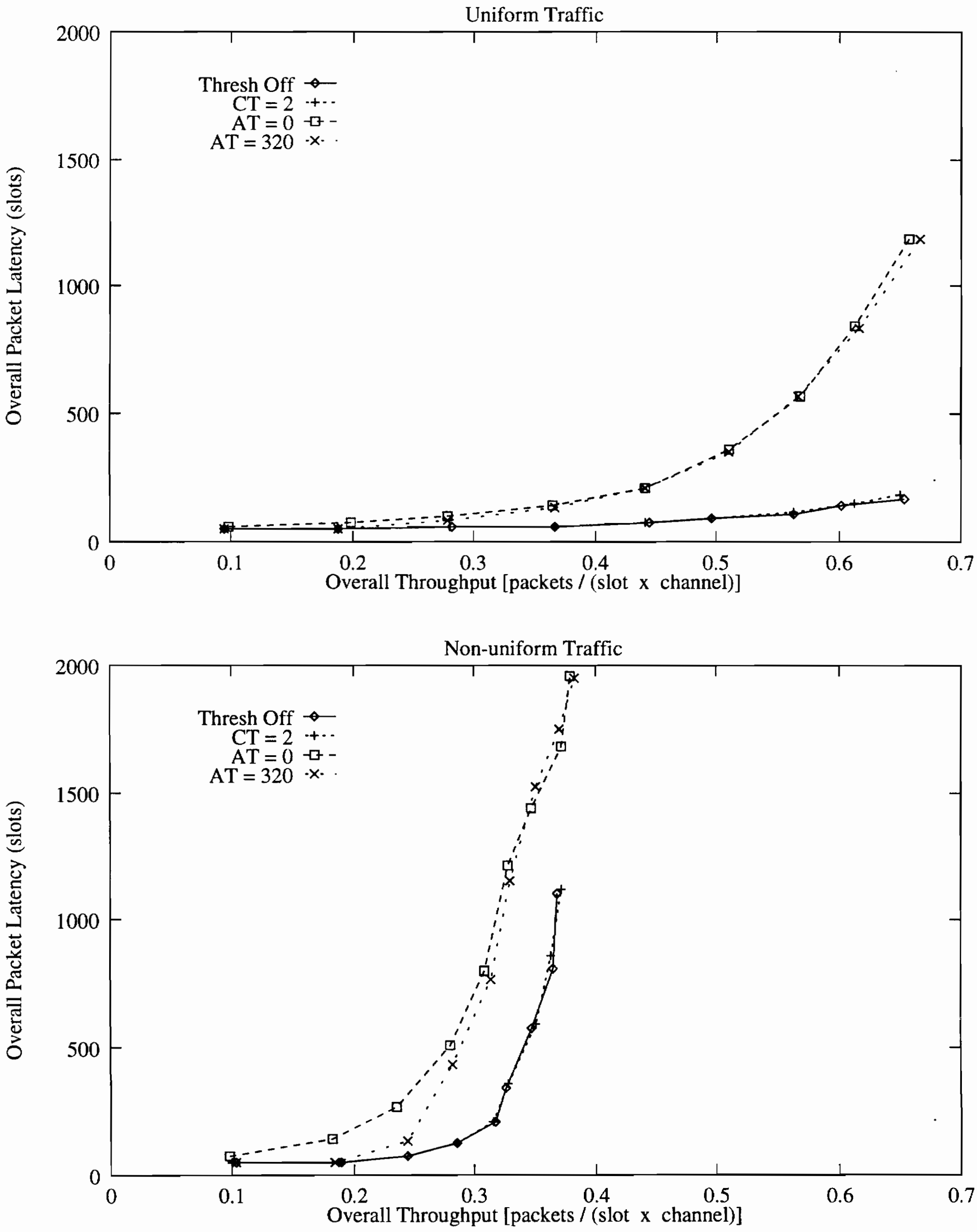
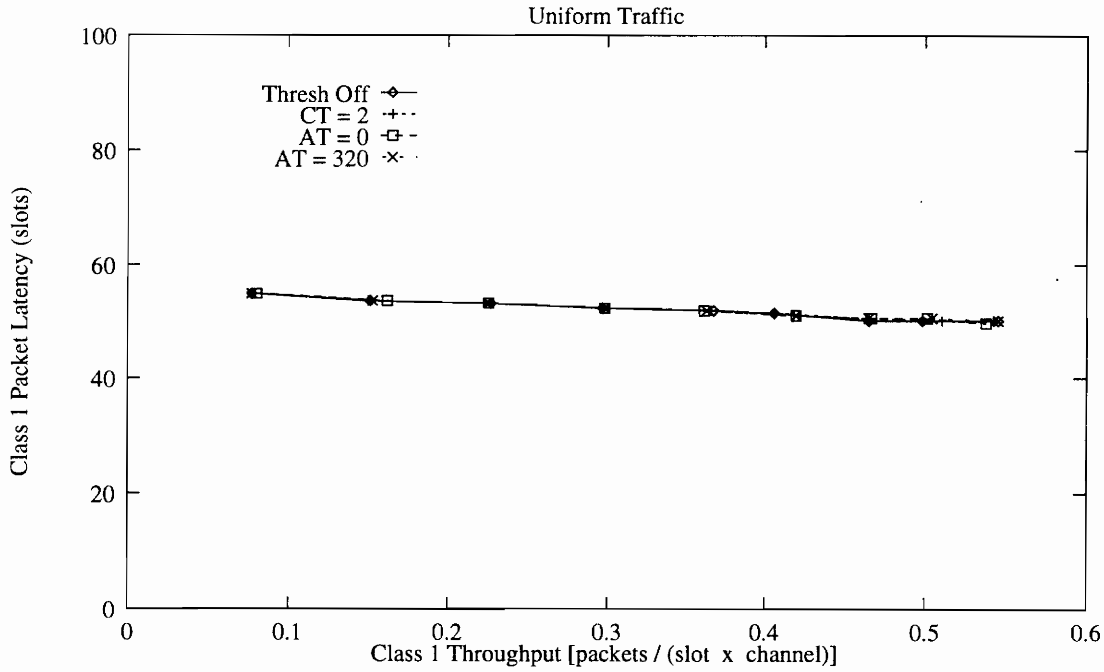


Figure 6.5: Phase I — Overall mean packet latency with uniform and non-uniform traffic.



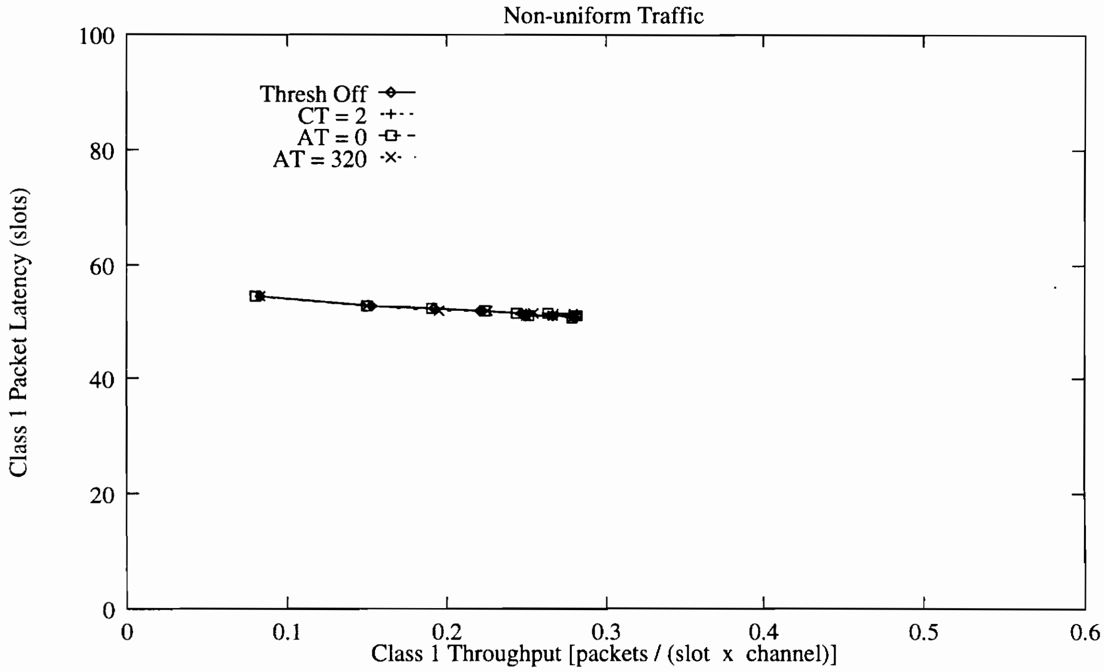
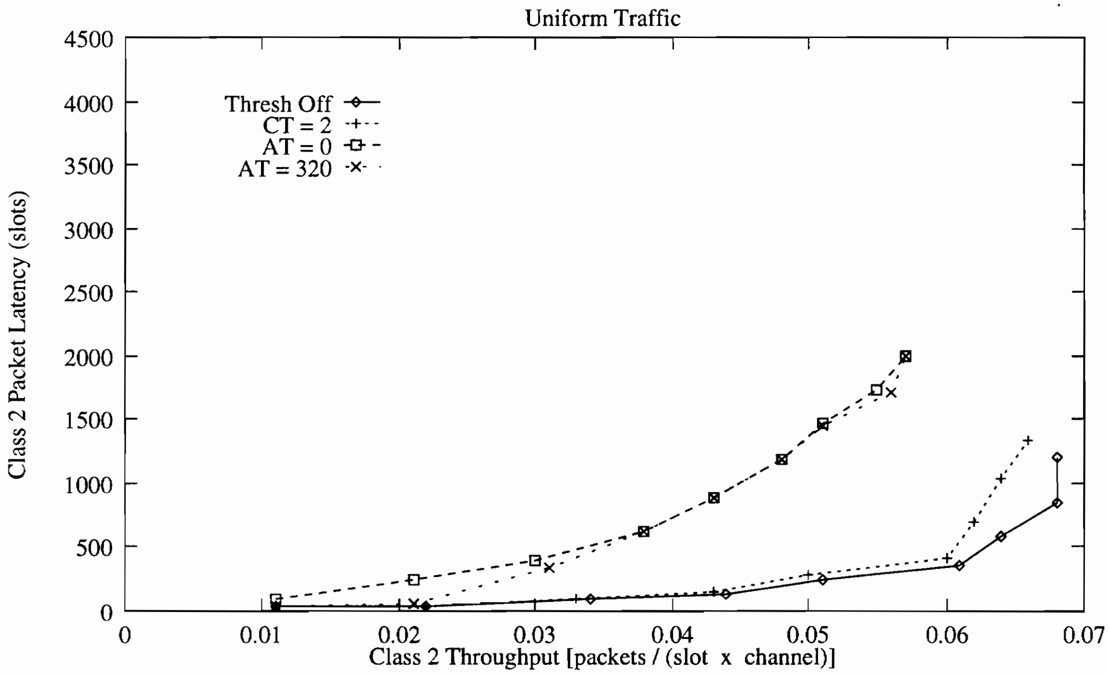


Figure 6.6: Phase I — Class 1 mean packet latency with uniform and non-uniform traffic.



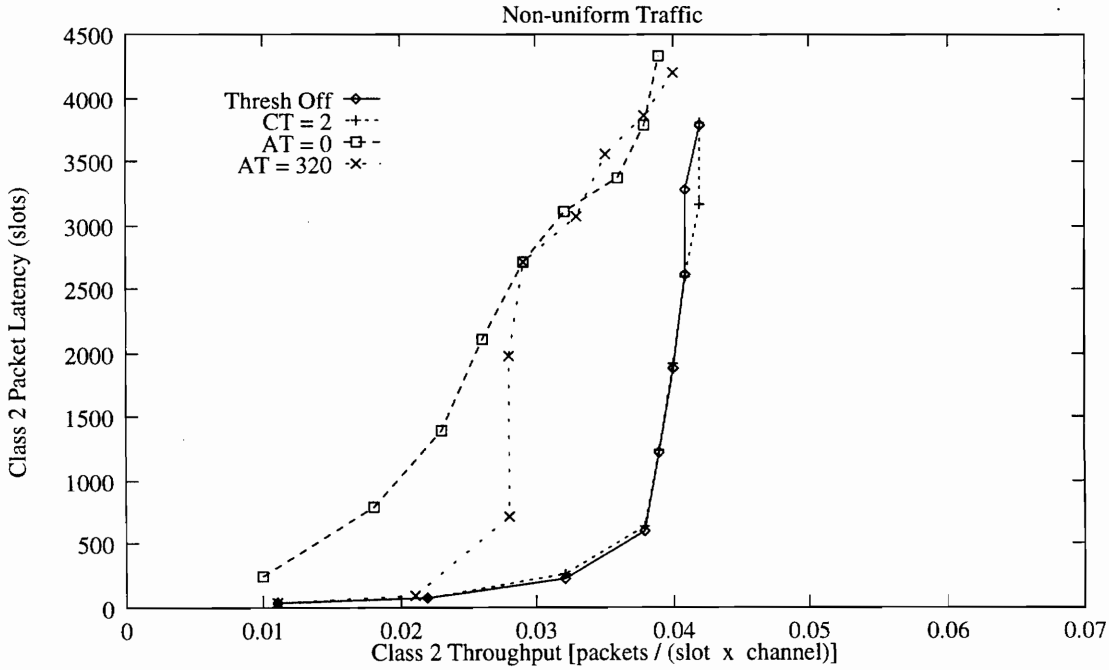


Figure 6.7: Phase I — Class 2 mean packet latency with uniform and non-uniform traffic.

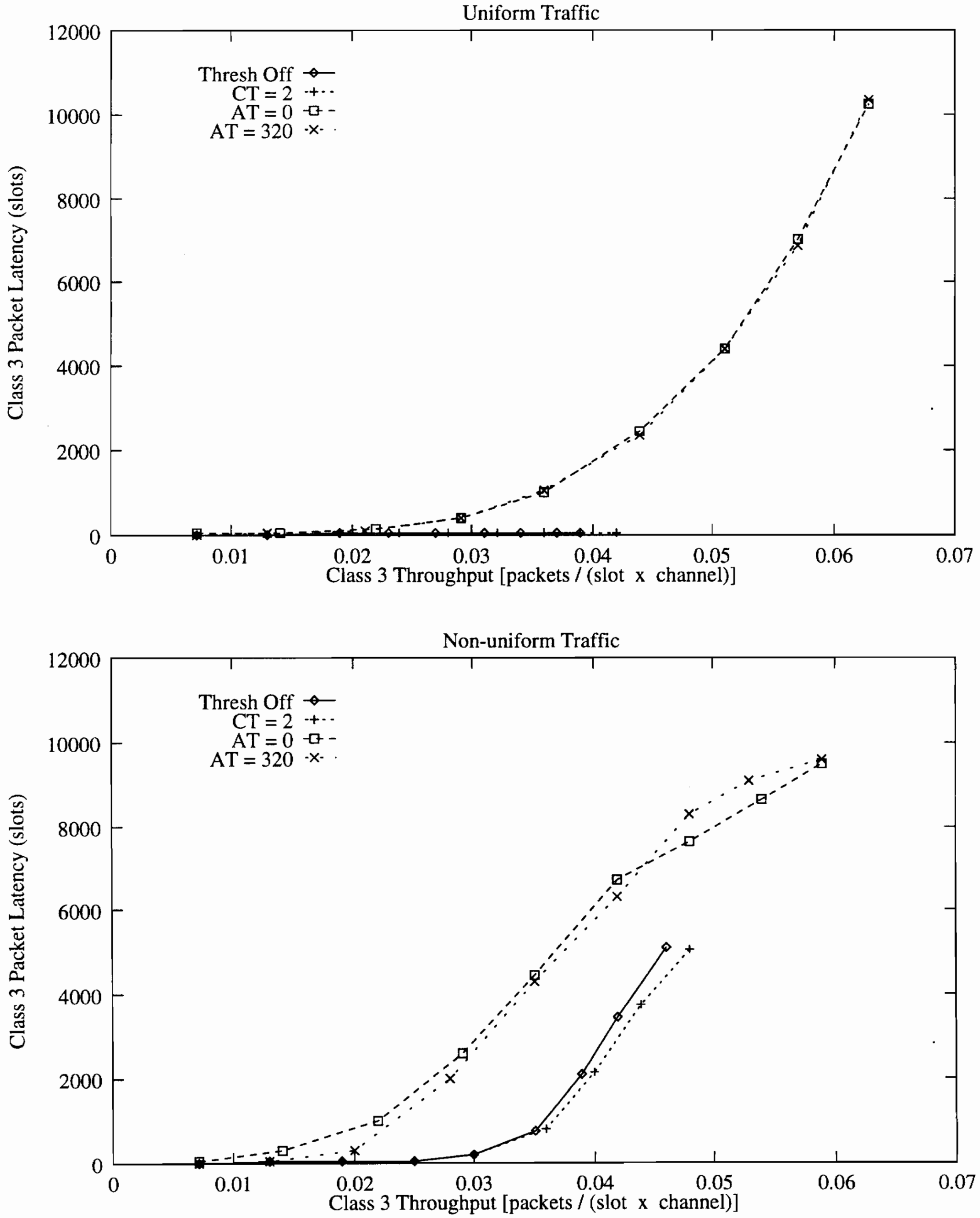


Figure 6.8: Phase I — Class 3 mean packet latency with uniform and non-uniform traffic.

the quality of service desired. The threshold of 0.25 provides a good balance between the desired small latency at low arrival rates and the desired stability at high arrival rates for the configuration simulated.

The collision threshold was intended to decrease collisions and increase throughput at a relatively low load. The arrival threshold would then be set at a higher level to control traffic system-wide. However, it is obvious that the collision threshold is not effective. By looking at traces of the number of collisions per cycle, it is apparent that the flow control needs to take effect for a longer period than one grand cycle when the collision threshold is crossed. Despite the backoff policy for class 2 traffic, a large number of collisions occurs immediately after the grand cycle where transmissions are not allowed to random destinations. The same principle would apply to the arrival threshold if it was crossed frequently due to bursty traffic. The problem is further aggravated by the fact that no backoff occurs for class 3 traffic due to the absence of feedback. This might not be a problem with a longer flow control period. If necessary, feedback for class 3 traffic could easily be added to each control packet.

As a general rule, non-uniform traffic results in lower throughput and higher latency. The class 1 results, shown in Figures 6.2 and 6.6, are equal for all cases. The class 1 latency with non-uniform traffic remains the same as the latency with uniform traffic as expected. As can be seen in Figures 6.3 and 6.7, the CT = 2 and threshold-off cases achieve higher class 2 throughput with lower delays than the cases with the arrival threshold on. Since connections are being modeled, random access performs better than the fixed TDMA schedule. It is apparent that when the arrival threshold is crossed, the TDMA schedule should be adjusted periodically to accommodate the traffic demands. The CT = 2 and threshold-off cases do not fare as well with class 3 traffic, as seen in Figure 6.4. The throughput is limited by collisions and the fact that class 3 is given lower priority than class

2. When the arrival threshold is crossed in the AT = 0 and AT = 320 cases, no collisions occur and higher throughput is achieved because class 3 traffic is completely uniform and is suited to the TDMA schedule. Note that the arrival rates in Figure 6.4 are the same for both uniform and non-uniform traffic because the class 3 arrival rate is not inhibited by falling throughput. As seen in Figure 6.8, the class 3 latency for the CT = 2 and threshold-off cases is much lower than the other two cases. However, this difference is somewhat exaggerated because class 3 is not made reliable and possible retransmissions from the MAC layer or a higher layer have not been modeled. It is expected that class 3 queue lengths and delays would be commensurate with the AT = 0 and AT = 320 cases if class 3 was made reliable.

At first, it seems odd that the CT = 2 and threshold-off cases achieve higher class 3 throughput for non-uniform traffic than uniform traffic. This can be explained by the fact that class 3 traffic remains uniform throughout the simulations and more packets are successfully transmitted in the non-uniform case by stations with less class 1 and class 2 traffic. On the average, however, the class 3 packets that do get through in the uniform case experience smaller delays. One final observation is that the class 3 latency with non-uniform traffic for the AT = 0 and AT = 320 cases begins to increase less rapidly at a throughput of 0.04, as seen in Figure 6.8. This is due to discarding that occurs because of the maximum queue size of 1024 packets. With larger queue sizes, the average latency with non-uniform traffic would increase beyond the average latency experienced with uniform traffic.

The simulations in this phase provide an idea of the actual maximum throughput achievable with a realistic traffic mix. Even with the arrival threshold constantly in force, the system starts to become unstable at arrival rates of 0.6 to 0.7 packets/(slot × channel) for uniform traffic and 0.3 to 0.4 packets/(slot × channel) for non-uniform traffic. In other words, the overall throughput is less than the arrival rate because class 3 arrivals are not restricted, and a steady state is never reached. With a 100 percent class 3 workload, the

Table 6.1: Phase II — Factors and Levels for the $2^4 \times 3$ Factorial Design

Variable	Factor	Level -1	Level 1
S	number of stations	8	32
A	mean packet arrival rate	0.1	0.4
T	traffic distribution	uniform	non-uniform
C	class distribution (1,2,3)	85%,10%,5%	75%,15%,10%

throughput could approach 1, the theoretical maximum derived in Section 4.3. However, a throughput of 0.6 packets/(slot × channel) is a realistic maximum since connections in the network have been modeled.

Dynamically adjusting the TDMA schedule with the arrival threshold in force could increase the maximum throughput and decrease latency for class 2 and class 3 traffic at the expense of increased complexity. It is possible that an improved flow control policy for the collision threshold, an improved backoff policy for class 2 traffic, and/or a backoff policy for class 3 traffic could provide better performance at heavy loads than the fixed TDMA schedule imposed by the global arrival threshold.

6.2 Phase II

Phase II consists of a $2^4 \times 3$ factorial design [62] with the factors and their respective levels given in Table 6.1. The overall mean packet latency is used as the response for the experiments. In a $2^4 \times 3$ factorial design, three replications of the 2^4 experiments are performed to determine the effect of the 4 factors, each of which has two levels, and to isolate experimental errors. There are a total of 2^4 effects including interactions and the overall mean. A straightforward additive regression model for this design is of the form

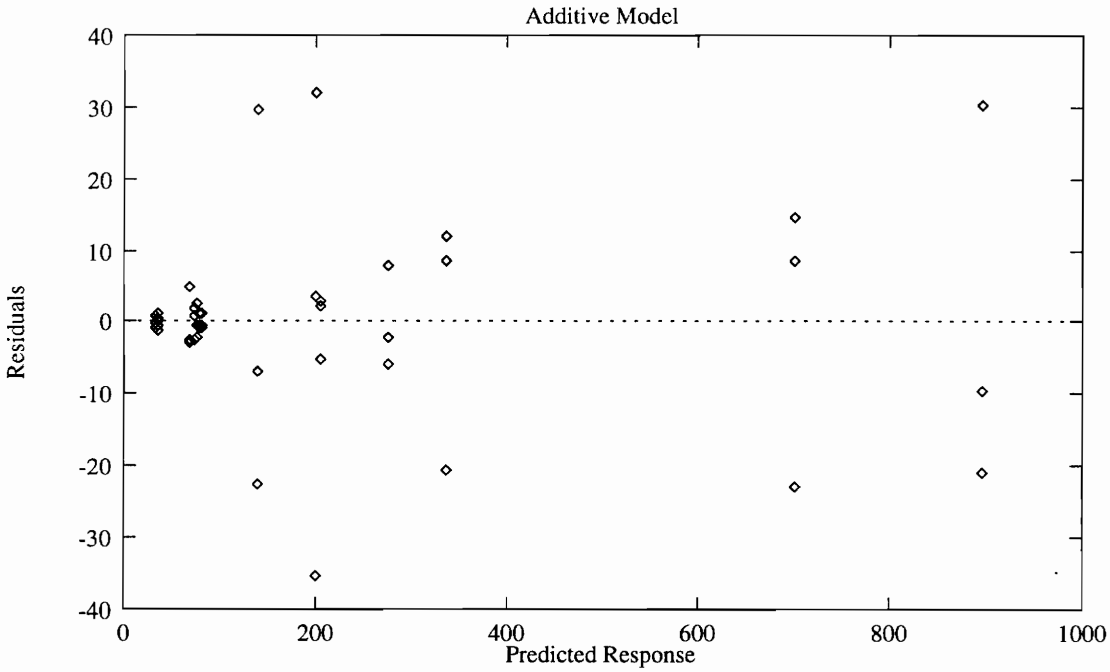
$$y = q_0 + q_S x_S + q_A x_A + q_T x_T + q_C x_C + q_{SA} x_S x_A + \cdots + e \tag{6.1}$$

where y is the estimated response, q_0 is the arithmetic mean, q_i is the effect of factor i, x_i is the level of factor i, and $q_{ij\ldots k}$ is the interaction between factors i, j, ..., k. The assumptions that are made for an additive model are as follows [62]:

1. Errors are statistically independent.
2. Errors are normally distributed with zero mean and constant standard deviation.
3. Effects of factors and errors are additive.

To test whether assumptions 1 and 2 were satisfied, a scatter plot of the residuals versus predicted response and a normal quantile-quantile plot for the residuals [62] were prepared. The scatter plot, shown at the top of Figure 6.9, should not have any trend if the errors are independent. Also, the spread should be comparable for all values of the predicted response if the standard deviation is constant. There is no visible trend, but the spread at the left side of the plot is less than the rest of the plot indicating that the standard deviation of the errors is not constant. Since the residuals are about one order of magnitude less than the response, this difference in spread is probably not significant. The normal quantile-quantile plot shown at the top of Figure 6.10 is less satisfactory. The plot should be approximately linear if the errors are normally distributed. The shape of the plot indicates that the error distribution has longer tails than the normal distribution.

Due to this violation of the assumptions of an additive model, three transformations for the data were explored: log, square root, and inverse. The square root transformation applies to a Poisson distributed variable. The inverse transformation is part of the family of power transformations. No transformation stood out in the standard deviation versus mean response graphs, so the log transformation was used since the ratio of the maximum observed latency to the minimum observed latency was fairly large and the spread in the residuals increased slightly with increasing response. Using the logarithm of the responses



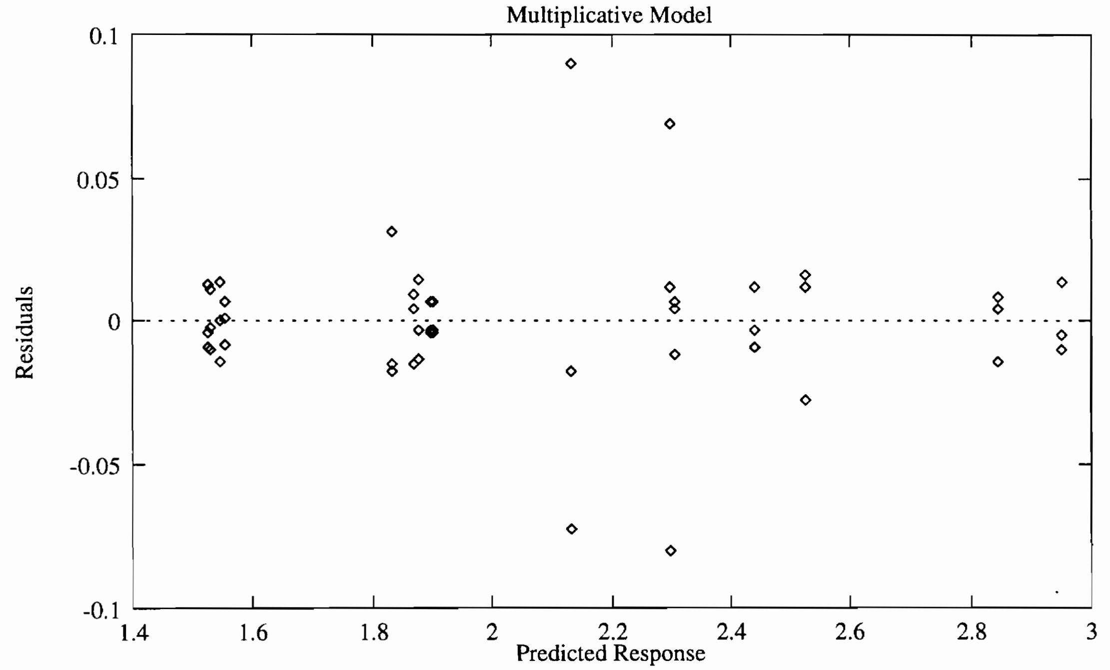


Figure 6.9: Phase II — Residuals versus predicted response for the additive and multiplicative models.

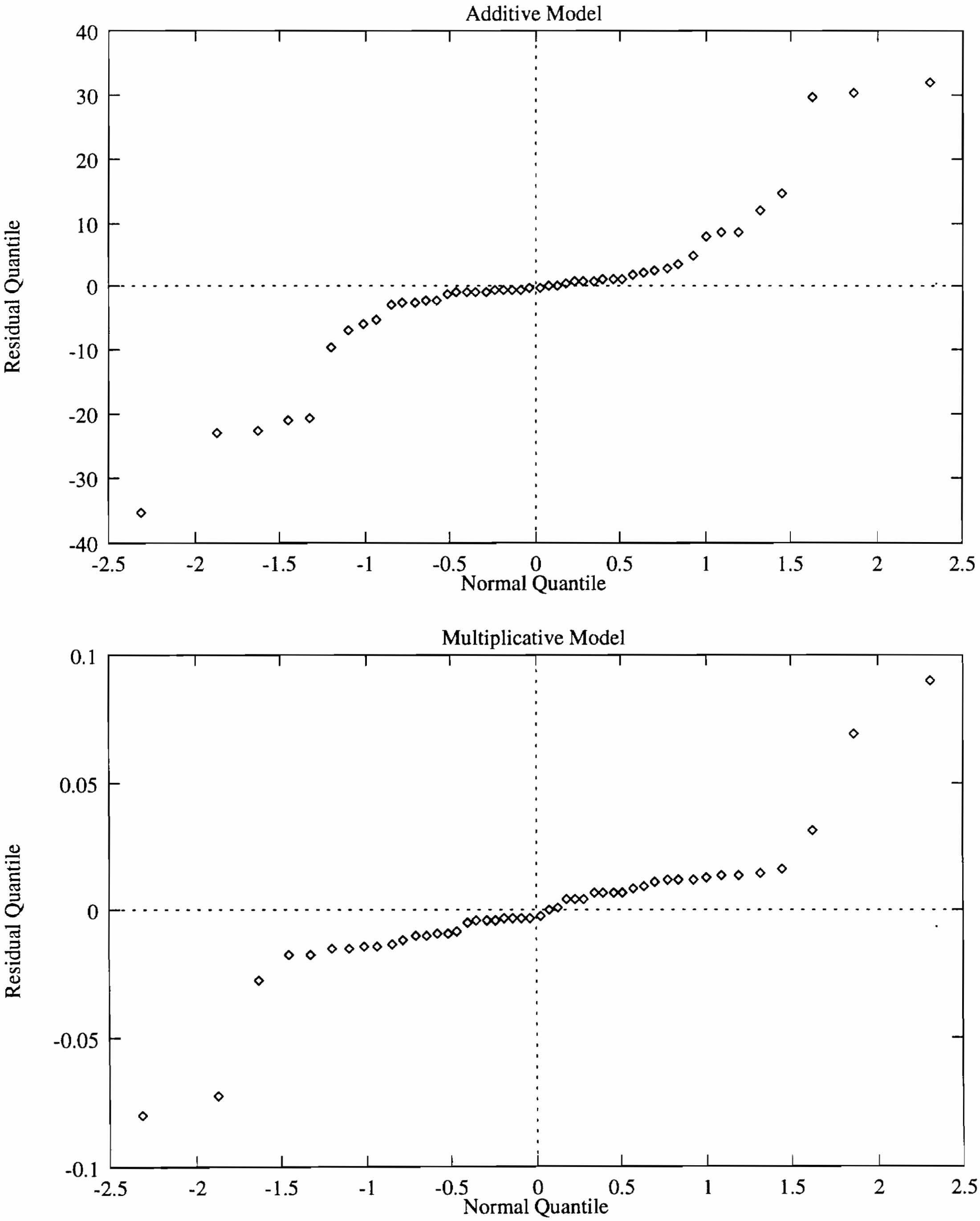


Figure 6.10: Phase II — Normal quantile-quantile plot for residuals of the additive and multiplicative models.

results in a multiplicative model, that is the effects of the factors and errors are assumed to multiply. With the transformed response, the model is of the form

$$\log(y) = q_0 + q_S x_S + q_A x_A + q_T x_T + q_C x_C + q_{SA} x_S x_A + \cdots + e \tag{6.2}$$

After the analysis, the antilog of the additive effects q_S, q_A, etc. is taken to produce the multiplicative effects $u_S = 10^{q_S}$, $u_A = 10^{q_A}$, etc. Physical considerations favor a multiplicative model over an additive model since it seems more appropriate to say the latency at one level is n times that at another level than to say it is n slots more. The first statement may apply to more than just the factor levels used for the study. As can be seen at the bottom of Figures 6.9 and 6.10, the scatter plot and normal quantile-quantile plot are more satisfactory for the multiplicative model because the scatter plot has an even spread and the normal quantile-quantile plot is closer to a straight line.

The results using the multiplicative model are shown in Table 6.2. Factor I is the mean response after the log transformation. A factor with more than one symbol represents an interaction. For example, factor SA corresponds to the interaction between the number of stations and the mean packet arrival rate. All effects are statistically significant since none of the confidence intervals contain zero. Most of the variation is explained by factors S, A, T, and C, and the interactions AT and AC. Since factor A (arrival rate) explains a large part of the variation (62.6 percent), it is clear that the performance of the protocol is less sensitive to varying traffic conditions (T and C) and the number of stations (S). Note that the arrival rate has been chosen to fall on either side of the arrival threshold, so it is expected to have a significant effect on the latency. Using the effects that have been calculated, the equation for the response (overall mean packet latency) becomes

$$y = 10^{2.066}\, 10^{0.144 x_S}\, 10^{0.352 x_A}\, 10^{0.118 x_T}\, 10^{0.082 x_C}\, 10^{-0.030 x_S x_A} \cdots 10^{e}$$

Table 6.2: Phase II — Effects and Variation Explained by the Multiplicative Model

Factor	Effect	Percentage of Variation	Confidence Interval for Effect
I	2.066		(2.057, 2.075)
S	0.144	10.4	(0.135, 0.153)
A	0.352	62.6	(0.343, 0.361)
T	0.118	7.1	(0.109, 0.127)
C	0.082	3.4	(0.073, 0.091)
SA	-0.030	0.5	(-0.0395, -0.0213)
ST	-0.042	0.9	(-0.0510, -0.0328)
SC	-0.032	0.5	(-0.0412, -0.0230)
AT	0.120	7.2	(0.111, 0.129)
AC	0.094	4.5	(0.0849, 0.103)
TC	0.034	0.6	(0.0252, 0.0434)
SAT	-0.041	0.8	(-0.0497, -0.0315)
SAC	-0.031	0.5	(-0.0400, -0.0218)
STC	-0.011	0.06	(-0.0202, -0.00202)
ATC	0.033	0.5	(0.0239, 0.0421)
SATC	-0.010	0.05	(-0.0187, -0.00055)
Errors		0.3	

or

$$y = (116.4)\, 1.39^{x_S}\, 2.25^{x_A}\, 1.31^{x_T}\, 1.21^{x_C}\, 0.93^{x_S x_A} \cdots 10^{e} \tag{6.3}$$

Here y is the estimated latency when the factors S, A, T, and C are at levels x_S, x_A, x_T, and x_C, respectively. The values -1 and 1 were assigned to the levels simulated.

The interpretation of the equation is as follows. The geometric mean of the packet latencies is 116.4 slots. The latency with 8 stations is 0.72 times (1.39^{-1}) that with 20 stations, and the latency with 32 stations is 1.39 times that with 20 stations. In other words, the latency with 32 stations is approximately twice (1.39/0.72) that with 8 stations. The other effects are interpreted similarly. However, since the effects of factors AT and AC are significant, the change in overall mean packet latency due to the arrival rate (A) is dependent on the traffic distribution (T) and the class distribution (C). Similarly, the

Table 6.3: Phase III — System and Workload Parameter Values for DMACS and DISA Comparison (a total of 8 different workloads are simulated using each protocol)

Parameter	Values	
number of stations	16	
mean packet arrival rate	0.1	0.4
traffic distribution	uniform	non-uniform
class distribution (2,3)	15%,85%	20%,80%

change in latency due to the traffic distribution or the class distribution is dependent on the arrival rate. The most important result obtained from phase II is that the arrival rate has a significantly greater effect on the latency than the other factors for the levels simulated.

6.3 Phase III

Phase III compares the performance of the DMACS and DISA [18] protocols. Since DISA does not support class 1 traffic, it was not included in the workloads simulated. Acknowledgments in DISA are piggybacked onto data packets and a local flow control threshold of $3C/8$ is used as described in Section 5.2. The specific input parameter values for the simulations are shown in Table 6.3. The percentage of class 2 traffic has been chosen to create a plausible number of connections in the network. For example, with an arrival rate of 0.1 packets/(slot × channel) and 15 percent class 2 traffic, there is an average of 2.9 class 2 connections per station. With an arrival rate of 0.4 and 20 percent class 2 traffic, there is an average of 15.4 connections per station. As a side effect, this causes the percentage of class 3 traffic to be so high that it diminishes the effect of the non-uniform traffic pattern since class 3 traffic always remains uniform. Note that an arrival rate of 0.4 is high enough to trigger the threshold in both protocols. If the arrival threshold in DMACS is crossed, the system becomes fixed TDMA in the next grand cycle. For the system to become fixed TDMA for a cycle in DISA, the local threshold has to be crossed simultaneously at every

Table 6.4: Phase III — Performance Comparison of DMACS and DISA Using the Method of Paired Observations (In all cases, the difference is computed as DMACS–DISA.)

Performance Metric	Sample Mean for Difference	95% Confidence Interval	Better Protocol
Overall Latency	53.1	(-12.4, 118.6)	similar
Class 2 Latency	-81.7	(-117.2, -46.2)	DMACS
Class 3 Latency	73.7	(-8.9, 156.3)	similar
Overall Throughput	0.040	(0.005, 0.075)	DMACS
Class 2 Throughput	0.0016	(0.0001, 0.0031)	DMACS
Class 3 Throughput	0.038	(0.005, 0.071)	DMACS

station. Since the arrival pattern remained relatively uniform in all the simulations, this did not create a large difference in the operation of the protocols. As in phases I and II, three repetitions were performed for each set of input parameter values.

The method of paired observations [62, 75] is used to quantify the difference in performance for the two protocols. The output metric value for DISA is subtracted from that of DMACS for each different workload. A confidence interval, known as a *paired-t confidence interval*, is constructed for the difference using the values from the 8 workloads. If the confidence interval contains zero, the performance of the protocols for that output metric is not significantly different. The results are shown in Table 6.4. As expected, the mean class 2 packet latency is less in DMACS because of the improved acknowledgment scheme. The acknowledgments in DISA are subject to variable queuing delays and collisions that reduce the rate of feedback when it is needed most. The class 3 latency is similar for both DMACS and DISA because class 3 traffic is not made reliable and thus requires no acknowledgments. Since the percentage of class 3 traffic is much greater than class 2, there is no difference in the overall latency. DMACS achieves slightly better throughput for both traffic classes. The global arrival threshold and the faster feedback from class 2 collisions on the control channel both contribute to the increase in throughput over DISA. Note that an improved flow control policy for the collision threshold could increase the performance of DMACS even further.

Phase III demonstrates the advantages of using a common control channel for acknowledgments and global flow control. In addition, DMACS directly supports constant bit-rate traffic which is a distinct advantage for multimedia traffic on a LAN or MAN.

Chapter 7

Conclusion

In conclusion, the most important analytical and experimental results are summarized in Section 7.1. Section 7.2 presents the novel features of DMACS not found in previous MAC protocols for single-hop WDM networks. A few topics for future research are given in Section 7.3.

7.1 Summary of Analytical and Experimental Results

The DMACS protocol has been evaluated both analytically and through simulation. The most important analytical results are the following:

- The collision probability for class 2 (virtual circuit) and class 3 (datagram) traffic is reasonably low in the region of operation below the arrival threshold.
- With an ideal traffic pattern, the overall throughput approaches 1 as the number of stations grows large.
- A low excess delay for reliable class 2 and class 3 traffic is achieved through acknowledgments on the control channel.

Through simulation with a realistic traffic mix, the following results were obtained:

- An arrival threshold of 0.25 packets/(slot × channel) provides a good balance between the desired low delay at light loads and the desired stability at heavy loads for the workloads simulated. It can be adjusted as needed to take effect when the probability of collision becomes excessive.

- The collision threshold is not effective on a channel by channel basis because the flow control needs to last more than one grand cycle.
- For the workloads simulated, the maximum stable throughput is approximately 0.6 packets/(slot × channel) for uniform traffic and 0.3 packets/(slot × channel) for non-uniform traffic.
- The packet arrival rate has a significantly greater effect on the packet latency than the number of stations, the traffic distribution, or the class distribution.
- DMACS achieves lower latency for class 2 traffic and higher overall throughput when compared to the DISA protocol on which it is based.

7.2 Novel Features of DMACS

DMACS is an extension of the DISA protocol requiring an additional fixed transmitter and receiver per station to access a common control channel. In addition, the control channel requires one additional wavelength. The benefit of these additions is a relatively low bit-rate common control channel that provides the following:

- reservations for constant bit-rate (class 1) traffic,
- acknowledgments for class 2 and class 3 traffic desiring reliable service, and
- system-wide and "per channel" flow control.

From a review of the literature, DMACS is the first MAC protocol based on no pretransmission coordination to make use of a common control channel.

The only previous MAC protocol to support different traffic classes, N-DT-WDMA [19], requires pretransmission coordination since tunable data receivers are used. Compared to this protocol, DMACS has the following advantages:

- $N+1$ wavelengths instead of $2N$ (where N is the number of stations),
- one tunable device per station instead of two,
- flow control integrated into the MAC layer, and
- tuning only done for data packets instead of both control and data packets.

It is important to note that the control slot size in DMACS is equal to the data slot size making synchronization more feasible than for protocols using many small control slots per data slot.

DMACS directly supports connection setup and tear down, different traffic classes, flow control, and packet resequencing, functions normally performed in the transport layer. As in N-DT-WDMA [19], the integration of traditional transport layer functions into the MAC layer is done as a first step in providing a single protocol layer between the physical layer and the higher application-driven layers.

7.3 Topics for Future Research

There is still much research to be done in the area of all-optical networks. Ideas for future research directly related to the DMACS protocol are as follows:

- perform queuing delay analysis for both class 2 and class 3 traffic,
- evaluate performance with fewer data channels than stations,
- create and study the impact of an improved flow control policy for the collision threshold,

- explore dynamic window sizing methods for reliable class 2 and class 3 traffic,
- study the impact of making class 3 reliable or implementing backoff for class 3 without reliability, and
- evaluate performance with non-uniform distances from the star coupler or with other physical topologies, such as a unidirectional bus or tree.

At the device level of all-optical networks, some of the most pressing problems include [80]:

- fast tunable, wide range lasers and filters,
- cost reduction for optical components, and
- interconnection of all-optical networks through wavelength routing.

Continued development of fast packet switching protocols for WDM networks is needed, and decisions will have to be made on the optimum protocol layering for such networks.

A final important issue in the development of all-optical metropolitan area and wide area networks is the availability of "dark" fiber, fiber that is installed but not in use today, i.e. "lit" [1]. A user can directly attach optical terminations to dark fiber without having to go through a carrier's electronic equipment. Only about 54 percent of the more than two million miles of fiber installed by long-distance phone companies in the United States is lit [16]. The issue of whether or not dark fiber will be readily provided is controversial. It appears that cable television companies and alternate access carriers are willing to provide fiber over metropolitan area distances [16]. How this issue plays out will have a significant impact on the future deployment of all-optical networks.

REFERENCES

[1] P. E. Green, Jr., "The future of fiber-optic computer networks," *IEEE Computer*, vol. 24, no. 9, pp. 78–87, Sept. 1991.

[2] N. R. Dono, P. E. Green, Jr., K. Liu, R. Ramaswami, and F. F.-K. Tong, "A wavelength division multiple access network for computer communication," *IEEE Journal on Selected Areas in Communications*, vol. 8, no. 6, pp. 983–993, Aug. 1990.

[3] R. Gidron and A. Temple, "Teranet: A multihop multichannel ATM lightwave network," in *Proc. IEEE International Conference on Communications '91*, pp. 602–608, 1991.

[4] L. G. Kazovsky, C. F. Barry, M. J. Hickey, C. A. Noronha, and P. T. Poggiolini, "A multi-Gbit/s optical LAN utilizing a passive WDM star: Towards an experimental prototype," in *Proc. IEEE Infocom '93*, vol. 1, pp. 48–56, 1993.

[5] C. A. Brackett, "Dense wavelength division multiplexing networks: Principles and applications," *IEEE Journal on Selected Areas in Communications*, vol. 8, no. 6, pp. 948–964, Aug. 1990.

[6] B. Mukherjee, "WDM-based local lightwave networks — Part I: Single-hop systems," *IEEE Network*, vol. 6, no. 3, pp. 12–27, May 1992.

[7] P. E. Green, Jr., *Fiber Optic Networks*, Englewood Cliffs, NJ: Prentice-Hall, 1993.

[8] A. S. Acampora and M. J. Karol, "An overview of lightwave packet networks," *IEEE Network*, vol. 3, no. 1, pp. 29–40, Jan. 1989.

[9] P. S. Henry, "High-capacity lightwave local area networks," *IEEE Communications*, vol. 27, no. 10, pp. 20–26, Oct. 1989.

[10] C. Partridge, "Protocols for high-speed networks: Some questions and a few answers," *Computer Networks and ISDN Systems*, vol. 25, no. 9, pp. 1019–1028, Apr. 1993.

[11] T. F. La Porta and M. Schwartz, "Architectures, features, and implementation of high-speed transport protocols," *IEEE Network*, vol. 5, no. 3, pp. 14–22, May 1991.

[12] R. Jain, "Myths about congestion management in high-speed networks," Tech. Rep. DEC-TR-726, Digital Equipment Corporation, Littleton, MA, Jan. 1991.

[13] L. Kleinrock, "The latency/bandwidth tradeoff in gigabit networks," *IEEE Communications*, vol. 30, no. 4, pp. 36–40, Apr. 1992.

[14] R. Vetter and D. Du, "Distributed computing with high-speed optical networks," *IEEE Computer*, vol. 26, no. 2, pp. 8–18, Feb. 1993.

[15] H. R. van As, "Media access techniques: The evolution towards terabit/s LANs and MANs," *Computer Networks and ISDN Systems*, vol. 26, no. 6–8, pp. 603–656, Mar. 1994.

[16] R. Ramaswami, "Multiwavelength lightwave networks for computer communication," *IEEE Communications*, vol. 31, no. 2, pp. 78–88, Feb. 1993.

[17] G. N. M. Sudhakar, M. Kavehrad, and N. D. Georganas, "Access protocols for passive optical star networks," *Computer Networks and ISDN Systems*, vol. 26, no. 6–8, pp. 913–930, Mar. 1994.

[18] M. Carrato and P. W. Dowd, "Dynamic Interleaved Slotted Aloha: A multiple access protocol for wavelength division multiplexed photonic networks," in *Proc. SPIE (High-Speed Fiber Networks and Channels II)*, vol. 1784, pp. 217–228, 1992.

[19] P. A. Humblet, R. Ramaswami, and K. N. Sivarajan, "An efficient communication protocol for high-speed packet-switched multichannel networks," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 4, pp. 568–578, May 1993.

[20] A. Jajszczyk and H. T. Mouftah, "Photonic fast packet switching," *IEEE Communications*, vol. 31, no. 2, pp. 58–65, Feb. 1993.

[21] L. G. Kazovsky, C. Barry, M. Hickey, C. A. Noronha Jr., and P. Poggiolini, "WDM local area networks," *IEEE Lightwave Telecommunication Systems*, vol. 3, no. 2, pp. 8–15, May 1992.

[22] B. Mukherjee, "WDM-based local lightwave networks — Part II: Multihop systems," *IEEE Network*, vol. 6, no. 4, pp. 20–32, July 1992.

[23] J. A. Bannister, L. Fratta, and M. Gerla, "Topological design of the wavelength-division optical network," in *Proc. IEEE Infocom '90*, vol. 3, pp. 1005–1013, 1990.

[24] M. Gerla, M. Kovačević, and J. Bannister, "Optical tree topologies: Access control and wavelength assignment," *Computer Networks and ISDN Systems*, vol. 26, no. 6–8, pp. 965–983, Mar. 1994.

[25] I. M. I. Habbab, M. Kavehrad, and C.-E. W. Sundberg, "Protocols for very high-speed optical fiber local area networks using a passive star topology," *Journal of Lightwave Technology*, vol. LT-5, no. 12, pp. 1782–1794, Dec. 1987.

[26] P. C. Wong and T. S. Yum, "Design and analysis of a contention-based lookahead reservation protocol on a multichannel local area network," *IEEE Transactions on Communications*, vol. 36, no. 2, pp. 234–238, Feb. 1988.

[27] N. Mehravari, "Performance and protocol improvements for very high speed optical fiber local area networks using a passive star topology," *Journal of Lightwave Technology*, vol. 8, no. 4, pp. 520–530, Apr. 1990.

[28] H. W. Lee, "Protocols for multichannel optical fibre LAN using passive star topology," *Electronics Letters*, vol. 27, no. 17, pp. 1506–1507, Aug. 1991.

[29] J. M. Senior, J. M. McVeigh, and S. D. Cusworth, "Multichannel slot reservation protocol for use on WDM optical fibre LAN," *Electronics Letters*, vol. 27, no. 20, pp. 1875–1877, Sept. 1991.

[30] H. Shi and M. Kavehrad, "ALOHA/Slotted-CSMA protocol for a very high-speed optical fiber local area network using a passive star topology," in *Proc. IEEE Infocom '91*, vol. 3, pp. 1510–1515, 1991.

[31] G. N. M. Sudhakar, M. Kavehrad, and N. D. Georganas, "Multi-control channel very high-speed optical fiber local area networks and their interconnections using a passive star topology," in *Proc. IEEE Globecom '91*, vol. 1, pp. 624–628, 1991.

[32] G. N. M. Sudhakar, N. D. Georganas, and M. Kavehrad, "Slotted Aloha and Reservation Aloha protocols for very high-speed optical fiber local area networks using passive star topology," *Journal of Lightwave Technology*, vol. 9, no. 10, pp. 1411–1422, Oct. 1991.

[33] H. B. Jeon and C. K. Un, "Contention-based reservation protocols in multiwavelength optical networks with a passive star topology," in *Proc. IEEE International Conference on Communications '92*, vol. 3, pp. 1473–1477, 1992.

[34] F. Jia and B. Mukherjee, "The Receiver Collision Avoidance (RCA) protocol for a single-hop WDM lightwave network," in *Proc. IEEE International Conference on Communications '92*, vol. 1, pp. 6–10, 1992.

[35] F. Jia and B. Mukherjee, "The Receiver Collision Avoidance (RCA) protocol for a single-hop WDM lightwave network," *Journal of Lightwave Technology*, vol. 11, no. 5–6, pp. 1053–1065, May/June 1993.

[36] F. Jia and B. Mukherjee, "A high-capacity, packet-switched, single-hop local lightwave network," in *Proc. Globecom '93*, vol. 2, pp. 1110–1114, 1993.

[37] M.-S. Chen, N. R. Dono, and R. Ramaswami, "A media-access protocol for packet-switched wavelength division multiaccess metropolitan area networks," *IEEE Journal on Selected Areas in Communications*, vol. 8, no. 6, pp. 1048–1057, Aug. 1990.

[38] M. Chen and T.-S. Yum, "A conflict-free protocol for optical WDMA networks," in *Proc. IEEE Globecom '91*, vol. 2, pp. 1276–1281, 1991.

[39] K. Bogineni and P. W. Dowd, "A collisionless multiple access protocol for a wavelength division multiplexed star-coupled configuration: Architecture and performance analysis," *Journal of Lightwave Technology*, vol. 10, no. 11, pp. 1688–1699, Nov. 1992.

[40] P. W. Dowd and K. K. Bogineni, "Switching latency overlap techniques for WDM star-coupled media access protocols," in *Proc. IEEE Infocom '93*, vol. 1, pp. 65–74, 1993.

[41] R. Chipalkatti, Z. Zhang, and A. S. Acampora, "High speed communication protocols for optical star coupler using WDM," in *Proc. IEEE Infocom '92*, vol. 3, pp. 2124–2133, 1992.

[42] R. Chipalkatti, Z. Zhang, and A. S. Acampora, "Protocols for optical star-coupler network using WDM: Performance and complexity study," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 4, pp. 579–589, May 1993.

[43] I. Chlamtac and A. Fumagalli, "QUADRO-stars: High performance optical WDM star networks," in *Proc. IEEE Globecom '91*, vol. 2, pp. 1224–1229, 1991.

[44] I. Chlamtac and A. Fumagalli, "Quadro: A solution to packet switching in optical transmission networks," *Computer Networks and ISDN Systems*, vol. 26, no. 6–8, pp. 945–963, Mar. 1994.

[45] C.-S. Li, M.-S. Chen, and F. F.-K. Tong, "Architecture and protocol of a passive optical packet switched metropolitan/wide area network using WDMA," in *Proc. IEEE Globecom '92*, vol. 3, pp. 1891–1895, 1992.

[46] C.-S. Li, M.-S. Chen, and F. F.-K. Tong, "POPSMAC: A medium access protocol for packet-switched passive optical networks using WDMA," *Journal of Lightwave Technology*, vol. 11, no. 5–6, pp. 1066–1077, May/June 1993.

[47] D. Bertsekas and R. Gallager, *Data Networks*, Englewood Cliffs, NJ: Prentice-Hall, 2nd ed., 1992.

[48] J. M. Senior, S. D. Cusworth, and A. Ryley, "Performance investigation of a token passing access protocol for a multichannel optical fibre LAN," *Computer Communications*, vol. 11, no. 6, pp. 304–312, Dec. 1988.

[49] A. Ryley, S. D. Cusworth, and J. M. Senior, "Piggybacked token-passing access protocol for multichannel optical fibre LANs," *Computer Communications*, vol. 12, no. 4, pp. 213–222, Aug. 1989.

[50] I. Chlamtac and A. Ganz, "Channel allocation protocols in frequency-time controlled high speed networks," *IEEE Transactions on Communications*, vol. 36, no. 4, pp. 430–440, Apr. 1988.

[51] A. Ganz and Y. Gao, "A time-wavelength assignment algorithm for a WDM star network," in *Proc. IEEE Infocom '92*, vol. 3, pp. 2144–2150, 1992.

[52] G. N. Rouskas and M. H. Ammar, "Analysis and optimization of transmission schedules for single-hop WDM networks," in *Proc. IEEE Infocom '93*, vol. 3, pp. 1342–1349, 1993.

[53] A. Ganz and Z. Koren, "WDM passive star — Protocols and performance analysis," in *Proc. IEEE Infocom '91*, vol. 3, pp. 991–1000, 1991.

[54] K. M. Sivalingam, K. Bogineni, and P. W. Dowd, "Pre-allocation media access control protocols for multiple access WDM photonic networks," *Proc. ACM SIGCOMM '92* in *Computer Communication Review*, vol. 22, no. 4, pp. 235–246, Oct. 1992.

[55] K. Bogineni, K. M. Sivalingam, and P. W. Dowd, "Low-complexity multiple access protocols for wavelength-division multiplexed photonic networks," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 4, pp. 590–604, May 1993.

[56] K. M. Sivalingam, K. Bogineni, and P. W. Dowd, "Acknowledgement techniques of random access based media access protocols for a WDM photonic environment," *Computer Communications*, vol. 16, no. 8, pp. 458–471, Aug. 1993.

[57] K. M. Sivalingam, K. Bogineni, and P. W. Dowd, "Design and performance analysis of pre-allocation protocols for WDM photonic networks," in *Proc. SPIE (High-Speed Fiber Networks and Channels II)*, vol. 1784, pp. 193–204, 1992.

[58] M. J. Karol and B. Glance, "Performance of the PAC optical packet network," in *Proc. IEEE Globecom '91*, vol. 2, pp. 1258–1263, 1991.

[59] M. J. Karol and B. Glance, "A collision-avoidance WDM optical star network," *Computer Networks and ISDN Systems*, vol. 26, no. 6–8, pp. 931–943, Mar. 1994.

[60] F. S. Roberts, *Applied Combinatorics*, Englewood Cliffs, NJ: Prentice-Hall, 1984.

[61] M. E. Anagnostou and E. N. Protonotarios, "Performance analysis of the selective repeat ARQ protocol," *IEEE Transactions on Communications*, vol. COM-34, no. 2, pp. 127–135, Feb. 1986.

[62] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*, New York: John Wiley and Sons, 1991.

[63] E. Drakopoulos, "Performance and traffic analysis of a network-based distributed system," *Computer Communications*, vol. 16, no. 3, pp. 155–167, Mar. 1993.

[64] S. Kowtha and D. R. Vaman, "A generalized ATM traffic model and its application in bandwidth allocation," in *Proc. IEEE International Conference on Communications '92*, vol. 2, pp. 1009–1013, 1992.

[65] M. De Prycker, R. Peschi, and T. Van Landegem, "B-ISDN and the OSI protocol reference model," *IEEE Network*, vol. 7, no. 2, pp. 10–18, Mar. 1993.

[66] R. Jain and S. A. Routhier, "Packet trains — Measurements and a new model for computer network traffic," *IEEE Journal on Selected Areas in Communications*, vol. 4, no. 6, pp. 986–995, Sept. 1986.

[67] H. L. Pasch and I. G. Niemegeers, "Comparing network performance for constant and variable bit rate sources," *Computer Communications*, vol. 16, no. 1, pp. 27–38, Jan. 1993.

[68] W. Lau and S. Li, "Traffic analysis in large-scale high-speed integrated networks: Validation of the nodal decomposition approach," in *Proc. IEEE Infocom '93*, vol. 3, pp. 1320–1329, 1993.

[69] A. Schmidt and R. Campbell, "Internet protocol traffic analysis with applications for ATM switch design," *Computer Communication Review*, vol. 23, no. 2, pp. 39–52, Apr. 1993.

[70] A. M. Law, "Statistical analysis of simulation output data," *Operations Research*, vol. 31, no. 6, pp. 983–1029, Nov.–Dec. 1983.

[71] M. H. MacDougall, *Simulating Computer Systems: Techniques and Tools*, Cambridge, MA: The MIT Press, 1987.

[72] L. W. Schruben, "Detecting initialization bias in simulation output," *Operations Research*, vol. 30, no. 3, pp. 569–590, May–June 1982.

[73] N. R. Adam, "Achieving a confidence interval for parameters estimated by simulation," *Management Science*, vol. 29, no. 7, pp. 856–866, July 1983.

[74] B. Schmeiser, "Batch size effects in the analysis of simulation output," *Operations Research*, vol. 30, no. 3, pp. 556–568, May–June 1982.

[75] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis*, New York: McGraw-Hill, 2nd ed., 1991.

[76] G. S. Fishman, *Principles of Discrete Event Simulation*, New York: John Wiley and Sons, 1978.

[77] B. Stroustrup, *The C++ Programming Language*, Reading, MA: Addison-Wesley, 2nd ed., 1991.

[78] S. B. Lippman, *C++ Primer*, Reading, MA: Addison-Wesley, 2nd ed., 1991.

[79] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *Data Structures and Algorithms*, Reading, MA: Addison-Wesley, 1983.

[80] P. E. Green, Jr., "An all-optical computer network: Lessons learned," *IEEE Network*, vol. 6, no. 2, pp. 56–60, Mar. 1992.

Appendix A

Command-line Options for Simulation Program

Table A.1: Command-line Options for the DMACS/DISA Simulation Program (Class 3 traffic percentage is equal to 100 – class 1 percentage – class 2 percentage)

Option	Default	Description
-d	DMACS	Selects DISA protocol
-s num		Number of stations in network
-c num	INT_MAX	Collision threshold for DMACS in collisions/grand cycle
-t num	INT_MAX	Arrival threshold for DMACS in packets/grand cycle
-n	uniform	Selects non-uniform traffic pattern
-a num		Arrival rate in packets/(slot × channel)
-1 pct		Class 1 traffic percentage
-2 pct		Class 2 traffic percentage
-w num	1.0	Warm up time for each replication in seconds simulated
-r stream	0	First random number stream used (< 25)
-m file	off	Tracing option to record coll./gr. cycle on each channel
-v file	off	Tracing option to record arrivals/gr. cycle at each station

Examples of usage, where *dmacs* is the name of the program and *out1* and *out2* are files for simulation output:

```
dmacs -s 16 -a 0.5 -1 80.0 -2 12.5 out1

dmacs -s 8 -c 2 -t 80 -n -a 0.3 -1 75.0 -2 15.0 -w 0.5 -r 10 out2
```

VITA

Michael Charles Montgomery was born on May 25, 1971, in Oak Ridge, Tennessee. He graduated from Oak Ridge High School in the spring of 1989 and enrolled at Virginia Tech in the fall. As an undergraduate, he majored in computer engineering and graduated with Summa Cum Laude honors in the spring of 1993. During the summers of 1991 and 1992, he worked in the Scientific Workstation Support group at Oak Ridge National Laboratory.

A week after receiving his Bachelors degree, Michael married Heather Dugan, a native of Blacksburg and also a student at Virginia Tech. He began graduate studies in electrical engineering that summer and completed his Masters degree in July 1994.

Michael C. Montgomery