HIGH-SPEED COMMUNICATION PROTOCOLS FOR ALL-OPTICAL WAVELENGTH DIVISION MULTIPLEXED COMPUTER NETWORKS

By

Krishna Moorthy Sivalingam

June 1994

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE

AND THE FACULTY OF THE GRADUATE SCHOOL

OF THE STATE UNIVERSITY OF NEW YORK AT BUFFALO

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Abstract

Optic fiber is rapidly gaining significance as the primary medium in computer and telecommunication networks. The principal advantages of optic fiber over copper cable are its higher transmission speed and lower error rates. In addition, parallel multiple channels are made available on a single fiber using a technique called Wavelength Division Multiplexing (WDM). This dissertation studies the design, analysis and implementation of communication protocols which fully exploit the high speed, low error rate characteristics and the availability of multiple channels on a single fiber. The protocols studied are used to support interprocessor communication in an optically interconnected multiprocessor system.

The first part of the dissertation studies *media access* protocols for a passive star coupled WDM network. The star topology was shown to have advantages over the bus topology with respect to larger system size and better fault tolerance. The protocols studied are based on an architecture where each node has a single tunable transmitter and a fixed receiver. It evaluates the advantages of these schemes with respect to performance, cost and implementation.

Photonic amplification significantly increases the number of nodes that can be connected to the network. This motivates reconsideration of bus based optical networks. The second part of the dissertation studies the performance of the bus and star topologies with amplification with respect to system size, protocol performance, fault tolerance, cost, and implementation.

A scalable, hierarchical, all-optical multiprocessor architecture that supports distributed shared memory (DSM) has been considered earlier. The performance of the associated cache coherence protocols and the DSM organization is largely dependent on media access protocol performance. The third part of the dissertation examines protocols for this environment. The protocols studied are optimized for the bi-modal traffic characteristics of a DSM system. Reservation based protocols are compared to static allocation protocols.

Analytical modeling techniques based on stochastic processes and semi-Markov models are used to evaluate protocol performance. All the analytic models are validated through discrete-event simulation.

Acknowledgments

I thank my adviser Dr. Patrick Dowd for his incessant support, encouragement, guidance, and advice that made this dissertation possible. His emphasis on the quality of research and writing have been extremely valuable in writing this dissertation.

I thank Dr. Sreejit Chakravarty and Dr. Wennie Shu for accepting to be on my committee and sparing precious time for discussions. Their comments, involvement and support have played a significant role in this research. I also thank Prof. Aura Ganz of University of Massachusetts, Amherst, for suggesting changes that assisted in improving this dissertation.

The following acknowledges research done jointly with other members of the group: The research on pre-allocation media access protocols for WDM networks was developed with Dr. Kalyani Bogineni. The protocol synchronization mechanisms for a hierarchical network architecture were developed jointly with Dan Crouse. I would like to thank all members of Dr. Dowd's research group with whom I have had frequent discussions: Khaled Aly, Eric Blade, Kalyani Bogineni, John Chu, Dan Crouse, David Hoffmeister, and Jim Perreault.

My gratitude to my parents for their infinite patience and implicit faith in my capabilities is boundless and cannot be expressed in sufficient words. My thanks to my brother Prabhu are limitless for consistently invigorating me to remain focused on my research. This dissertation is dedicated to them. Special mention goes to my uncle Dr. Kandaswami and aunt Mrs. Kandaswami for taking exceptional care of me throughout my studies. I take this opportunity to thank K. Sivakumar and Ajay Shekhawat who have rendered help beyond measure. Last but not the least, I am extremely grateful to Thriveni for her unceasing cheer and steadfast optimism.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This dissertation studies the design, analysis and implementation of high speed communication protocols for optical networks. Networks based on a technique called wavelength division multiplexing that support multiple wavelength transmission on a single fiber are studied. These networks have many potential areas of application: local, metropolitan, and wide area networks, processor interconnection in multiprocessor systems, and heterogeneous cluster-based computing systems. This dissertation emphasizes the study of *media access* protocols for an *optically interconnected distributed shared memory multiprocessor system*. The following sections introduce the basic concepts of the issues studied in the dissertation.

1.1 Fiber optic networks

Optical fiber is becoming widespread as the communication medium and is expected to be the dominant mode of transmission in the future. With the maturing of optical fiber technology, transmission costs and in particular cost of high speed data links has dropped tremendously in the last few years.

Fiber optic networks are characterized by very high transmission speeds and very low error rates. Conventional wide area networks based on twisted pair or coaxial cable operate at speeds of 64 Kbps to up to 45 Mbps. Local area networks such as Ethernet operate at 10 Mbps. Copper media can support transmission rates up to Gbps. On the other hand, the fiber has a potential bandwidth of the order of 25000 to 30000 Gbps [1]. Also, the fiber optic networks are characterized by orders of magnitude better error rate (bit error rate of 10^{-15} compared to 10^{-5} of previous networks) [2]. Other advantages of optic networks include smaller size, lighter weight, lower attenuation, electromagnetic isolation, and greater repeater spacing [2].

The electronic processing capability of the computers has not increased to meet the high speed of the communication medium. In conventional networks, electronic processing speed is faster than the transmission speed. In fiber optic networks, transmission speed is much higher than electronic processing speed which is typically limited to a few gigabits per second. The communications bottleneck has thus shifted from the transmission medium to the processing medium. This requires redesign of the network protocols to efficiently utilize the high speeds and low error rates.

One technique to minimize the speed mismatch between processing and transmission is Wavelength Division Multiplexing (WDM) [1]. Multiple optical channels can be formed on a single fiber using this technique, where data can be simultaneously transmitted on *different wavelengths* on the same fiber. This is achieved using tunable transmitters and/or receivers at each network node. Thus, it is possible to establish concurrent communication between more than one source-destination pair using a single fiber. The number of such channels is typically of the order of 20-30 channels [1], but the feasibility of having more than 100 channels has been demonstrated [3]. Each of these channels operate closer to the maximum speed of the electronic processing units which is of the order of Gigabits per second.

1.2 Access Protocols for Star

The OSI model [4] for computer communication defines a hierarchical layering of communication protocols. Each layer performs a set of network functions which are used by the layers above this layer. The lowest layer is the *Physical* layer to the physical medium. The *Data Link* layer and *Network* layer are the next higher layers. The *Data Link* layer is subdivided into *Media Access* and *Logical Link Control* layers. The *Network* layer is subdivided to provide two main functions: *Routing* and *Congestion Control*. The *Transport* layer lies above the Network layer and uses the services offered by the Network layer.

Media access protocols provide access arbitration for the network nodes to the transmission medium. Access schemes vary according to the network interconnection topology. Some common topologies used for local area networks are *star*, *bus* and *ring*. In the star topology, all nodes transmit to a star coupler which transmits data back to all the nodes. In the bus topology, nodes are connected by pair of unidirectional buses carrying data. Their is a linear ordering of the nodes based on the node positioning on the bus. The ring topology interconnects nodes in a circular fashion.

Power budget considerations for fiber optic networks [5] showed that the system size supported by the bus was very small whereas the star could support hundreds of nodes. This led to research interest in the design of access protocols for star networks [6–13].

WDM star networks can be configured as *single-hop* networks where each node can communicate directly with other nodes. *Multi-hop* networks impose a virtual topology such as Shufflenet on the physical star topology [14]. Multi-hop networks usually have a cost advantage over single-hop networks since the former does not require wavelength tunable devices. Single-hop networks has the potential advantages of *optical self-routing* and *all-optical* communication between source and destination. This dissertation focuses on single-hop networks.

Media access protocols developed for single-hop WDM star networks have been classified into *reservation* and *pre-allocation* based protocols. Reservation techniques require at least one of the channels to operate as a *control channel* which is used to reserve access to the remaining channels (designated as data channels). Pre-allocation strategies operate without a separate control channel – all channels are used for data transmission. The latter approach appears promising due to its low implementation complexity and potential low cost. The first part of this dissertation studies the design and analysis of pre-allocation access protocols for the WDM star.

1.3 Access Protocols for Hierarchical Architecture

The star topology could perhaps support up to hundreds of nodes without any form of amplification. The goal is to provide communication for networks consisting of thousands of nodes. This can be achieved in two possible ways: use optical amplification, or build a hierarchical network from smaller sized clusters. Typical networks would combine the advantages of both these approaches to increase system size.

The improvements of optical amplifiers [2,15] have the potential of extending the system size of bus based networks, motivating their reconsideration. The bus with amplification has the capacity to support hundreds of nodes [2, 16]. However, the star was still shown to retain its excellent fanout relative to the bus with optical amplifiers. In addition to providing implicit node ordering, the bus is characterized by lower fiber but higher coupler requirements [17], and requires amplifiers even for a reasonable network size. The star, on the other hand, is characterized by large fanout and high fault tolerance, but requires more fiber. The second part of this dissertation evaluates the performance of bus and star topologies for fiber optic networks with optical amplifiers.

A scalable, hierarchical all-optical architecture has been developed to achieve the system objectives of low-latency and low-cost through a combination of spatial reuse of WDM channels [18]. The intention is to provide a reconfigurable structure that supports distributed shared memory (DSM) [19] and capitalize on any reference locality above uniform [20]. Each processor is associated with a portion of the system address space which can be accessed by other processors through message passing mechanisms. However, the communication appears to the user and operating system as a shared memory reference. Cache coherence protocols (snooping or directory based) have to be provided to maintain cache consistency across all the processors [21]. The performance of the DSM organization and the cache coherence protocol is dependent on the media access protocol. Careful attention has to be devoted to the design of the access schemes in an effort to minimize latency and deliver the performance advantage to the application level.

The objective of this hierarchical architecture is to achieve scalability yet avoid the requirement of multiple wavelength tunable devices per node. Furthermore, single-hop communication is achieved: a packet remains in the optical form from source to destination and does not require intermediate routing. This all-optical characteristics is achieved for both data transport *and* control.

Access protocols that support the DSM organization and associated cache coherence mechanisms is studied as the third part of this dissertation. The protocol studied in this dissertation, called FatMAC, is a hybrid approach that combines the advantages of receiver pre-allocation and reservation access strategies. The protocol has the following characteristics: low implementational complexity, collisionless transmission, low-latency at low loads and stability at high loads, support of varying propagation delays through interleaving cycles, and support of variable packet sizes without segmentation. This protocol reserves access on a pre-allocated channel through control packets.

The dissertation is organized as follows. Chapter 2 provides background information on the OSI model and WDM networks. Chapter 3 presents the media access protocols studied for a WDM star network. Chapter 4 studies the performance of star and bus topologies for a WDM network with optical amplification. Chapter 5 presents the hierarchical architecture and evaluates the performance of media access protocols for this network. Chapter 6 summarizes the research and identifies future scope of this work.

Chapter 2

Photonic Networks

The objective of this chapter is two-fold: first, to present an introduction to fiber optic networks and wavelength division multiplexed (WDM) networks in particular; and second, to discuss media access protocols for multiple access, shared medium WDM networks.

In conventional networks, electronic processing is much faster than the communication network. The conventional networks were characterized by high bit error rates and low bandwidth. Communications bandwidth was a severe technological bottleneck. Protocol processing was elaborate and used redundant error checking at different levels to ensure error-free communication. Protocol and network design incorporated efficient bandwidth utilization as the main objective. In this process, communication latency was usually traded off to conserve bandwidth which was the crucial network resource.

With the introduction of optics for communication, there now exists an enormous imbalance in the technologies of communications and computing. The limitations of electronics place an upper limit on the processing speed - up to two gigabits per second with silicon and up to five gigabits per second with gallium arsenide circuitry. Fiber optics is characterized by very high bandwidth of the order of 25000 Gigabits of capacity at three transmission bands (0.85, 1.3 and 1.55 μm) [2]. Fiber optic networks are also characterized by bit error rates as low was 10^{-15}. The bottleneck in the network has now shifted to the electronic processing. Protocol and network design have to be reconsidered with reduced latency as one of the main goals. Bandwidth can be traded off to ensure low latency communication.

The classical approach of optic network usage involves replacing the communication medium in existing networks using fiber. The non-classical approach involves redesign of existing networks using the special characteristics of optic networks. One such characteristic is the simultaneous data transmission on different wavelengths on a single fiber. This technique is called wavelength division multiplexing (WDM). This is the focus of the first part of the chapter.

The particular emphasis in this dissertation is on *all-optical* networks where the information is maintained throughout its route in the optical domain. The performance limitations imposed by the electronic processing are observed only at the two end-points of the networks. The first step toward utilizing the properties of optical networks is the design of protocols for accessing the network in a shared environment. This is the focus of the second part of this chapter and the rest of this dissertation. The main principles in redesigning protocols for the high bandwidth, low error-rate photonic environment can be summarized as follows:

- ▹ Utilize the low error rate and high bandwidth of optics
- ▹ Reduce the speed mismatch between electronics and optics
- ▹ Maintain *all-optical* communication
- ▹ Achieve low-latency response
- ▹ Sacrifice bandwidth to reduce latency

Protocol design for computer communication is typically based on a layered approach. The different layers of the protocol model provide different functionalities. Section 2.1 provides a brief introduction to the layered approach in protocol design proposed by the Open Systems Interconnection model. This dissertation is concerned with the development of protocols at the *media access* level for a photonic network. Section 2.2 provides the background to WDM networks and devices. Section 2.3 discusses previous research and classification of access protocols for WDM networks.

2.1 OSI Model

This section briefly describes the Open Systems Interconnection (OSI) model for computer networks which was approved in 1983 as an international standard by the International Organization for Standardization (ISO) and by the CCITT.

There are two requirements to data communication between two hosts: (i) data delivered by an end user should arrive at the destination correctly and in timely fashion; (ii) data ultimately delivered to the end user is recognizable and in proper form for its correct use. A number of *network protocols* handle the first part of the problem and *higher-level protocols* solve the second part.

In a typical network, the higher-level protocols are executed only at the end users. The intermediate nodes execute the network protocols. These two protocols are further broken down into a series of sub-layers leading to the OSI Reference Model as shown in Fig. 2.1. The key concept behind layered communication is the hierarchy between the layers and the fact that each lower layer provides a set of services to the layer above it.

The *physical* layer and *data link* layer ensure error-free communication along a single link between two nodes in the network. The function of the *physical* layer is to ensure that

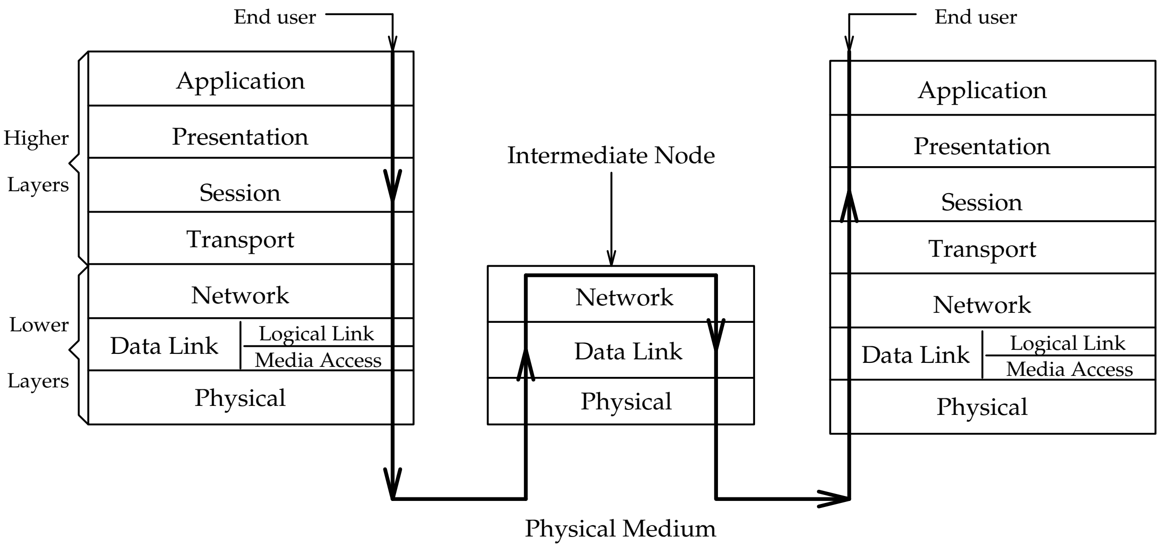

Figure 2.1: OSI model for computer communication.

a bit entering the transmission medium at one end reaches the other end without error. The purpose of the *data link* layer is to ensure that data blocks are reliably transmitted over the network using the underlying physical layer services. The data link layer is further layered into two sub-layers: *Media Access Control* (MAC) Layer and *Logical Link Control* (LLC) Layer. The *Media Access* protocol provides a mechanism for the nodes to gain access to the transmission medium. The *Logical Link* protocol ensures orderly, correct delivery of packets between neighboring nodes in a network.

The *network* layer uses the data link layer services and provides a two-fold service: *routing* data through the network and providing *flow/congestion control* to prevent network resources from being used up. *Routing* provides a path or multiple paths that must be set up to connect a source to a destination. *Congestion control* provides mechanisms needed to guard against statistical fluctuations in traffic which may deplete network resources (such as nodal buffers and transmission links) resulting in congestion. These three layers are grouped and referred to as the *network-level* protocols and are executed at each intermediate node of the data transmission path from source to destination.

The *transport* layer is the lowest of the higher-level protocols and the first end-to-end layer. This is the layer that ensures reliable, sequenced exchange of data between the two end users. It provides the necessary and reliable end-to-end data transmission service for the session layer and other layers above it. It uses the services of the network layer but provides a transparent service so that the details of the network layer and below are hidden from the transport service users. The session layer and other higher level protocols provide user-oriented services and are not of importance to this proposal and will not be discussed any further. The most commonly used transport and network layer protocols are the TCP/IP protocol suite [22].

The layering of protocols as described above imposes an enormous overhead on computer communication. A packet needs to be handled by least three different protocol layers. This results in huge overhead due to information exchange between functions operating at different layers and leads to high buffering requirements. Some of the error checking and flow control functions are done at more than one layer. This may not suitable for a high speed environment where efforts are directed towards reduced protocol processing and elimination of the opto-electronic speed mismatch.

One possible solution to this problem is to compact the layers and reduce protocol execution. For example, the data link layer and network layer can be compressed to a single layer which handles media access, link control, flow control and congestion control. Similarly, routing through intermediate nodes can be eliminated by providing virtual point-to-point links between all pairs of nodes. The result will be a compacted layer structure that provides faster communication.

This dissertation studies *low complexity* protocols for the Media Access sublayer of the OSI model. The next section provides the background for multi-wavelength communication using wavelength division multiplexing.

2.2 WDM Networks

Wavelength-division (WDM) multiplexing has become a popular choice to circumvent the electronic/optic speed mismatch by partitioning the traffic. Multiple channels, each operating at a rate that the interface electronics can sustain, are formed which eliminate the need that every node receive and process all network traffic as is usually the case with a single channel [1]. Each channel can be switched independently and operate at speeds closer to the electronic interface. WDM networks are established using tunable transmitters and/or receivers to switch between the multiple channels created on the single optical fiber. This dissertation considers the case of both single channel and multiple channel Broadcast-Select WDM networks [1] operating in a multiple access environment.

2.2.1 Optical Components

WDM networks are established using wavelength tunable optical devices. The next sections briefly discuss the principles of wavelength tunable receivers and transmitters. This is followed by a discussion of optical fiber amplifiers including erbium doped fiber amplifiers (EDFA). A complete discussion of this topic may be found in [2].

Tunable Receivers

Wavelength selectivity can be achieved with either a coherent receiver, or a tunable receiver with direct detection. The first approach is more expensive, but has higher channel selectivity: channels can be placed closer together so a greater number of channels are formed in the tunable range. A lower cost alternative is a tunable filter with direct detection.

The requirements for a tunable filter are [2]: (a) tunable range or the number of channels it can tune to, and the frequency selectivity of the response after tuning, (b) time required to tune from one channel to another, (c) attenuation, (d) controllability, (e) cost, (f) polarization independence and (g) size and power consumption.

The filters are typically based on *interference effects* that are wavelength selective. Non-coherent, wavelength tunable filters can be constructed with a variety of techniques: Fabry-Perot and Mach-Zehnder approaches of wavelength dependence of interferometric phenomena, wavelength dependence of coupling through acousto-optic or electro-optic techniques, and resonant amplification that provides gain as well as wavelength selectivity [23, 24]. Fabry-Perot and Mach-Zehnder filters have been constructed where 30 channels have been separated with milli-second switching speed [1]. Active filters have either electro-optic or acousto-optic control. A filter bandwidth of 1 nm has been achieved with both approaches [23]. However, acousto-optic devices have a tuning range across the full 1.3 - 1.56 μm range, while the electro-optic devices are limited to about 15 nm [23, 25]. Acousto-optic devices have switching speeds (μs) slower than electro-optic devices (ns). Acousto-optic and electro-optic Filters based on polarization beam splitters (PBS) have also been investigated [26–28]. The acousto-optic PBS devices have large tuning ranges and have tuning time in the range of μs. The electro-optic PBS devices have similar range but have much smaller tuning times ($0.1\mu s$). Tunable filters based on *switched-grating* approach have been studied in [29–32].

Acousto-optic tunable filters (AOTF) for WDM have been described in [24, 25, 28, 33]. AOTFs have advantages of broad tuning range, fast electronic tunability, narrow filter bandwidth, and multiple wavelength selectivity which make them preferable for WDM networks. An optical cross-connect based on AOTFs is described in [1, 25] and can be used to route signals on different wavelength channels independently among different optical fibers in a WDM system.

Tunable Transmitters

Tunable laser diodes have been an area of active investigation and a summary may be found in [2, 34]. The ideal device is a single-frequency laser diode that can span up to the spectral width of 200 nm at 1.3 or 1.5 μm and is rapidly tunable. The different forms of tunability can be listed as: (a) *overall tuning range*, (b) *continuous tuning range*, which is the width of the largest tuning range, and (c) *range of continuous control*, the range over which control currents and voltages vary monotonically without jumps.

Tunable lasers can be achieved through thermal, mechanical, injection-current and acousto-optic means [23, 35]. Thermal and mechanical methods achieve slow tunable devices (ms – μs). Injection-current techniques are capable of tuning speeds of a few ns. External-cavity tunable lasers use an anti-reflection coating on one facet of a Fabry-Perot laser diode and using one of the tunable filters discussed above, between the active laser region and some external mirror reflection. These lasers have tuning-speed problems due to the speed of the tunable filter and the delay for the laser active region to see the new filter setting. WDM networks have become possible through recent advances in narrow linewidth Distributed Feedback (DFB) and Distributed Bragg Reflector (DBR) tunable lasers and filters [34, 36]. The range of a DFB laser has a limit of 10-15 nm due to heating and non-radiative combination [37]. A range of 30 nm has been achieved using a four-section tunable laser in [36]. Phase matched tunable laser diodes use a scheme by which the emitted wavelength is made to depend on the *difference* in the indices of the two current-injected regions rather than on the index. A tuning range of 57 nm has been achieved using this technique [38].

Spectral slicing is a low cost alternative to tunable laser diodes that has been recently introduced for this environment [39–43]. A tunable transmitter with C channels is constructed using C LEDs and a WDM multiplexer. The multiplexer is used to extract the desired wavelength for each channel and block the remaining spectrum of the LED. A system based on spectral splicing has been implemented using off-the-shelf components [42]. Spectral slicing is suitable for short-distance WDM applications when the number of wavelengths required is small.

Wavelength control based on the property that the wavelength of the gain peak of Er^{3+}-doped fiber amplifier decreases as the degree of inversion increases in the fiber. Wavelength tuning over a range of 8 nm has been achieved using this technique [44]. A new intracavity tuning mechanism based on electro-optic reflection Mach-Zehnder reflectometer provides high speed tuning and wide tuning range of 40 nm [45]. Wavelength tuning with a single control current in a tunable twin-guide DFB laser and a tuning range of 57 nm has been reported in [46].

An array of surface emitting laser diodes fabricated on the same wafer can provide a tunable transmitter [47–49]. Tunability is achieved by tuning only one wavelength at a time. The tuning time is determined by the speed of switching from one laser to another.

Optical Amplifiers

Photonic amplification is essentially achieved using two practical approaches: *laser diode amplifiers* and *fiber doped amplifiers*. The latter approach is more popular because of the simplicity of manufacture and fiber coupling, polarization independence, wide bandwidth and comparative freedom from crosstalk. Laser diode amplifiers are more expensive to manufacture and have high a level of crosstalk. However, fiber doped amplifiers operate only in the 1.5 μm band while laser diode amplifiers operate equally well in both the 1.3

and 1.5 μm region.

Amplifiers can play three basic roles in lightwave systems [2]:

- ▹ The power amplifier to boost the output from a transmitter
- ▹ The line amplifier placed between the transmitter and the receiver along the fiber
- ▹ The receiver preamplifier to boost the incoming signal before receiver decoding

Amplifier placement has a significant impact on the system power budget and has been studied in [2].

Two types of laser diode amplifiers are possible: Fabry-Perot amplifiers and traveling wave amplifiers. For network purposes, the traveling-wave amplifiers are more interesting because of their flatter overall gain spectrum and ease of control. Fiber doper amplifiers typically use erbium ions co-doped with alumina. The overall 3 dB bandwidth at 1.5 μm is over 35 nm. For the power amplifier operation, power output values as high as 500 mW has been achieved. Values as high as 46 dB total gain and 10.2 dB gain increase per milliwatt increase of pump power have been reported.

Semiconductor amplifiers are very promising for future optical communication systems because of their high gain, broad amplification bandwidth, low power consumption, and compactness. A 4-channel array at 1.3 μm with 20 dB gain and less than 1 dB polarization sensitivity has been reported in [50].

Pr^{3+} doped fiber amplifier is a candidate for use as an optical amplifier in 1.3 μm transmission. This has been indicated by demonstration of a laser diode pumped PDFA with a gain of 28.3 dB. The main issue for improving PDFA performance is to increase the gain coefficient as experimented in [51,52].

This section summarized the characteristics of the optical devices required to establish WDM systems. The next section studies the optical multiple access topologies for a WDM network.

2.2.2 Network Topology

This section studies the topology of the network to interconnect the nodes. A variety of optical interconnection topologies can be used to achieve a multiple access environment for the photonic network as shown in Fig. 2.2 [53]. The commonly studied topologies are the bus and the star. The bus topology has been further classified as singly folded unidirectional bus, doubly folded unidirectional bus, and dual bus. The bus topology has been favored for local network interconnection since degradation due to propagation delay can be reduced by taking advantage of the physical ordering [53]. The unidirectional nature of the photonic bus establishes a *physical ordering* among the nodes which is exploited by the Demand Assignment Media Access (DAMA) protocols [53].

Fig. 2.2(a) shows the singly folded unidirectional bus configuration (FUB). The FUB has two unidirectional sections – the *outbound* bus for data transmission and the *inbound* bus for data reception. This configuration has been used in protocols including D-Net [54]. Fig. 2.2(b) shows the doubly-folded unidirectional bus (DFUB). This configuration has been used in Expressnet [55]. Both FUB and DFUB configurations require a sense tap and a transmitter on the outbound bus and a receiver on the inbound bus. A dual-bus configuration (DUB) is shown in Fig. 2.2(c) with two independent buses carrying traffic in opposite directions. This approach has been used by Fasnet [56] and DQDB (*IEEE 802.6)*. Each node has a sense tap, transmitter and a receiver on each bus doubling system cost. In a single channel system, the sense tap and receiver functions can be implemented using a single device [56]. In multi-channel systems where devices can be fixed or tunable, the sense tap and receiver may be implemented separately. The physical layout of the star is shown in Fig. 2.2(d). Each node has an outbound fiber to and an inbound fiber from the

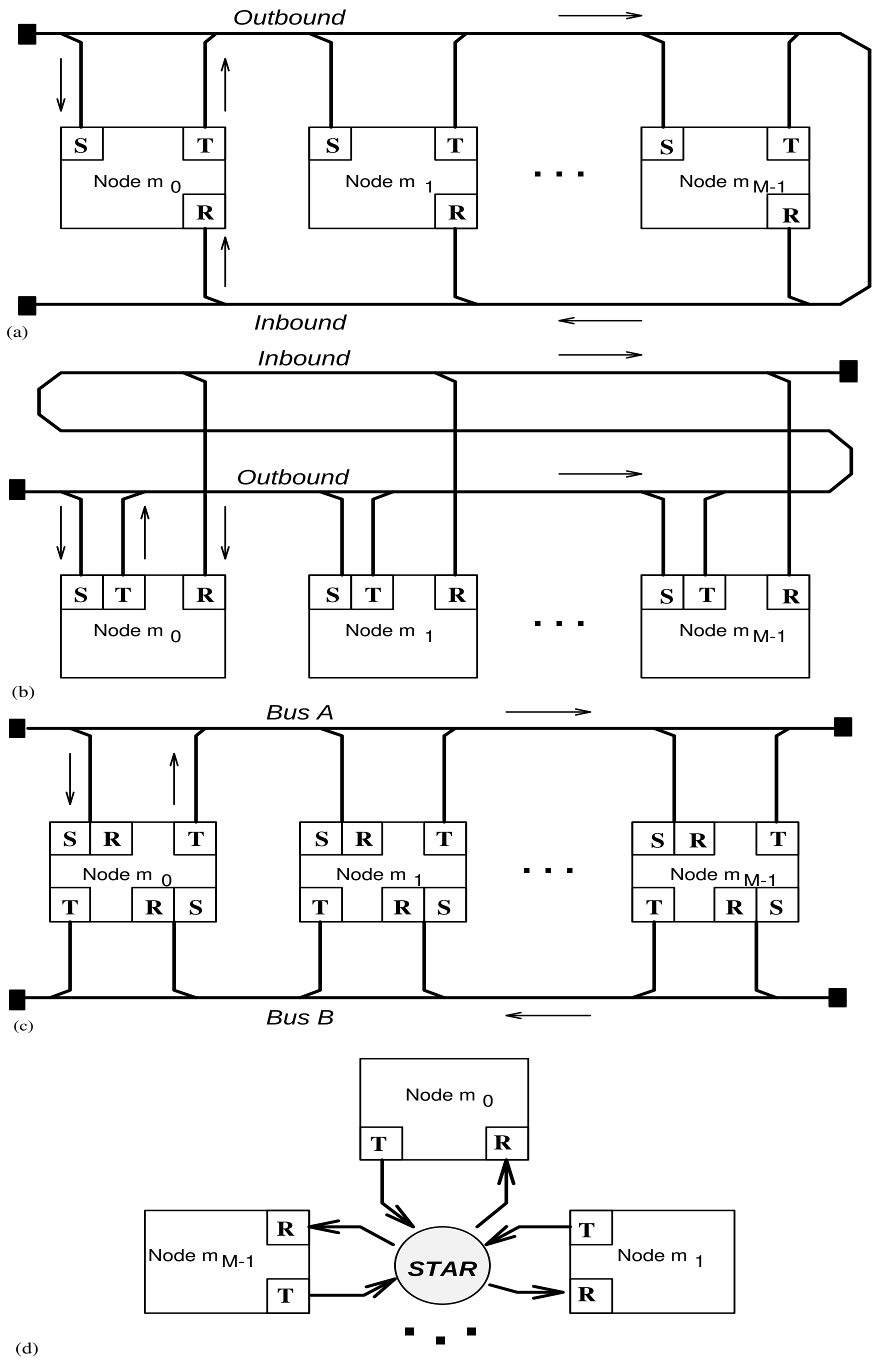

Figure 2.2: Optical multiple access channel configurations: (a) folded unidirectional bus, (b) doubly folded unidirectional bus, (c) dual unidirectional bus, and (d) star-coupled configuration.

star.

The optical power budget (OPB) characteristics determine the number of nodes that can be connected in a network while maintaining a specified bit-error-rate. The number of nodes attached to a multiple access channel is bound by the saturation traffic and the optical power budget. The high bandwidth of fiber precludes the saturation limitation and the principal factor of optical fanout is the power budget. Power budget considerations for fiber optic networks [5] showed that the system size supported by the bus was very small whereas the star could support hundreds of nodes. The system sizes of bus and star networks with erbium doped fiber amplifiers (EDFA) have been studied in [2, 16] and the star was also shown to have better fanout for multi-channel systems. A FUB network with an optimized number of amplifiers (Gain of 15 dB) and active reciprocal couplers was shown to have a fanout of over 400 nodes, whereas a star network with receiver pre-amplification was shown to support over 2000 nodes [2].

The star topology exhibits the following characteristics which make it a better alternative the bus topology:

1. System size limitations in terms of the optical power budget were compared in [5,57] and the star-coupled configuration was shown to exhibit superior fanout characteristics over bus-based topologies. However, bus based networks have renewed interest due to recent developments on optic amplifiers.

2. Star-coupled networks have high fault-tolerance due to their passive nature and complete unity distance connectivity [57,58].

This section provided a background on WDM networks including tunable transmitters, receivers and optical amplifiers. The following section examines the WDM media access mechanisms for the star topology.

2.3 WDM Media Access

This section discusses classification of WDM networks. This is followed by a discussion of multiple access protocols for a WDM star coupled network.

2.3.1 Network Classification

WDM Networks have been classified using a variety of criterion based on their characteristics. Three classes of WDM networks are Wavelength Routing (WR), Broadcast-Select (BS) and Wavelength Switching (WS) [1]. In a WR network, the transmitted wavelength completely determines the path in the network. This needs wavelength selectable devices. A Broadcast-Select network is passive with no internal wavelength selectivity. Selectivity is achieved by using tunable transmitters and/or tunable receivers. BS network can be configured to provide optical multiple access channels [59, 60], or virtual point-to-point interconnection in the wavelength domain [14, 61–71]. Wavelength switching can be done in two ways: dynamically where WDM routing is changed in the network to switch from one path to another, or by wavelength conversion where a signal is transferred from one to another on an optical fiber. WS devices based on AOTFs [25, 72] or wavelength converters [73, 74] could be used to realize WS networks. WMCH [75] and FHA [18] architectures are based on BS passive star-coupled configuration.

This proposal considers a BS network operating in a multi-access environment. As mentioned above, BS networks can be classified as *single-hop* multiple access [1] or *multi-hop* virtual topology networks [61].

Single-hop networks: provide a direct path between source and destination without intermediate routing. Packets are transmitted on an all-optical path from source to destination. Multiple nodes can simultaneously access the medium. Single-hop networks require at least one tunable device to accomplish transmission. Their main advantage is the simplicity of node addition/deletion and the elimination of intermediate routing and electronic processing. Single-hop networks typically require at least one wavelength tunable device per node.

Multi-hop networks: represent regular or irregular graph topologies implemented in the wavelength domain. In this case, a "virtual" topology is embedded on the physical star topology [61]. They require intermediate routing nodes and hence increasing packet latency due to electronic processing [14]. Multi-hop networks typically do not require wavelength tunable devices and have a potential cost-advantage over single-hop networks. However, the number of channels may be linked to the number of nodes limiting the system size [14].

It is possible to embed any virtual topology on the physical star topology. This is achieved by setting the wavelengths of the transmitters and receivers of the nodes. For example, consider a set of 3 nodes interconnected using a physical star topology. Let each node have two transmitters and two receivers; and let $\{\lambda_i | 1 \leq i \leq 6\}$ represent the set of available wavelengths. The following wavelength allocation establishes a virtual *ring* topology:

Node	T_1	T_2	R_1	R_2
1	λ_1	λ_2	λ_4	λ_5
2	λ_3	λ_4	λ_1	λ_6
1	λ_5	λ_6	λ_2	λ_3

The perfect shuffle implementation on a star topology is shown in Fig 2.3 and has been investigated in [14]. The performance of a multi-hop network is determined by its virtual topology. A summary of access protocols for single-hop and multiple-hop networks may be found in [76, 77].

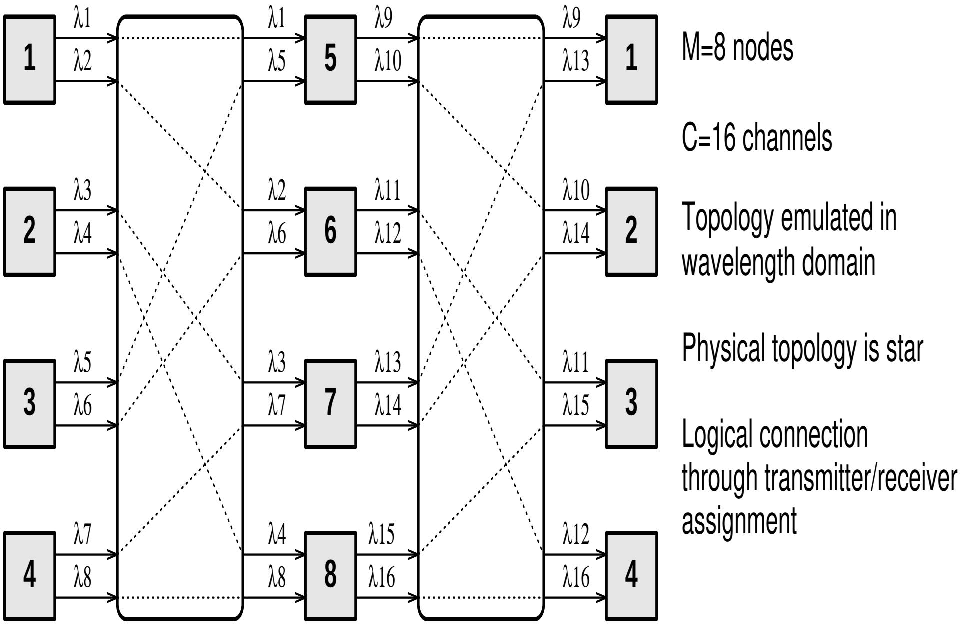

Figure 2.3: Shufflenet implementation on the physical star topology.

Depending on the architecture, *optical self-routing* can be achieved where a node receives only data destined to it and the system has non-blocking crossbar connectivity [57]. Optical self-routing partitions the traffic relaxing the design constraints on the receiver subsystem. This is crucial since the photonic network has higher throughput support than the packet processing capabilities.

This dissertation focuses on photonic star-coupled passive networks that use WDM to create wavelength division multiple access (WDMA) channels. The reasons for choosing this approach are:

- ▷ Provide an all-optical network
- ▷ higher topological connectivity
- ▷ low system complexity
- ▷ improved fault tolerance

▹ avoid intermediate processing and delay

The next section studies multiple access protocols for a single channel system with broadcast. This is followed by a discussion of protocols studied for the multi-channel environment.

2.3.2 Single Channel Multiple Access

Multiple access protocols for single channel broadcast networks have been extensively studied. A summary of multi-access protocols for these networks may be found in [78–81]. The protocols are typically classified as contention based and non contention based protocols. Random access or contention based protocols allow more than one node to access the medium at the same time resulting in possible collisions. The protocol usually defines a contention resolution mechanism in the event of a collision. Static access or non-contention based protocols pre-determine the time instants a node may access the network avoiding collisions. The simplest random access protocols are Aloha (A) and Slotted Aloha (SA) [81]. Time Division Multiplexing (TDM) is a typical example of a static access protocol [81].

Performance evaluation of access schemes are usually based on packet delay (latency), channel utilization, and access fairness. Packet delay is the average time between packet generation at the source and reception at the destination. Channel utilization is the fraction of time the network is utilized for useful data transmission. Network throughput is defined as the number of packets that are sent successfully per unit time. Access fairness defines whether all nodes share the medium equally without undue advantage to any particular set of nodes. The performance of protocols is typically studied with varying number of nodes and the system traffic load.

The performance of random access schemes can be characterized as: (a) excellent

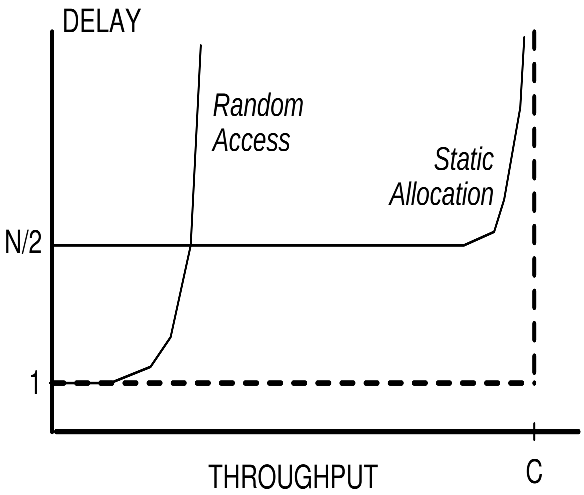

Figure 2.4: Performance characterization of Random and Static Access protocols.

packet delay under light loads, (b) potential instability and low utilization under heavy traffic, (c) possible unfair medium sharing. The performance of static access schemes can be characterized as: (a) high packet delay under light traffic, (b) stable performance and high utilization under heavy traffic, and (c) fair access to all nodes. Fig. 2.4 summarizes the delay-throughput of the two approaches. Other extensions to these two approaches include reservation-SA, reservation-TDMA, Carrier Sense Multiple Access (CSMA), contention tree protocol [81]. These protocols have performance characteristics that lie between these SA and TDM.

The next section examines extension of these principles to multiple channel WDM networks.

2.3.3 WDM Multiple Access

Multiple access protocols for WDM networks can be classified into two major categories: *Reservation Based* and *Pre-Allocated Based* strategies.

Reservation Based: Reservation strategies designate one of the C channels as the *control channel* which is used to coordinate data transmission between the nodes. The remaining channels are used for data transmission and are referred to as *data channels*. Each node typically has (i) a transmitter and receiver which are tuned to the control channel; and (ii) a transmitter/receiver which can tune to the data channels.

For example, if node A has to send a packet to node B, it transmits a control packet on the control channel which is monitored by all receivers. The control packet contains the destination node and the data channel in the control packet. Conflicts can occur when more than one node attempts transmission at the same time on the control channel or the same data channel. The media access protocol ensures that the control packet is transmitted successfully. Node B, on receiving this control packet, sets up its receiver to receive the data packet on the channel indicated in the control packet. Media access protocols are now required to provide arbitration and resolve conflicts on the data channels too.

A typical control channel based MAC protocol for the multiple channel WDM network is designated as X/Y, where X and Y are implemented as the MAC protocols for the control and data channels respectively. Some combinations of such protocols introduced in [6] are A/A, SA/A, A/CSMA, CSMA/A. A discussion of control channel protocols for WDM networks may be found in [76].

Pre-Allocation Based: Unlike reservation based schemes, pre-allocation based protocols do not require a separate *control channel*. All channels are used for data transmission. This approach requires either a tunable receiver or a tunable transmitter. If the transmitter is tunable, the receiver is fixed and receives data only on one channel (*home channel*). A transmitter tunes to the home channel of the receiver and transmits the packet. If $M > C$, then a home channel is shared among more than one node. If $M = C$, each node has its own channel and self-routing is achieved. If the receiver is tunable, the transmitter is fixed on one channel (*home channel*) and the receiver tunes to the transmitter's home channel. The latter approach requires coordination for the receiver to tune to the transmitter's home channel. We have investigated the first approach because it is simpler and tunable transmitters are less expensive than tunable receivers.

The idea of pre-allocation was initially considered in [9, 82, 83]. The reservation approach has attracted a lot of research attention as presented in [6, 76]. Reservation protocols promise greater flexibility in the availability of data channels. However, some protocols proposed link the number of channels to the number of nodes in the system [8,12]. The number of WDM channels that may be available theoretically may be as high as 100. A more realistic number determined by device characteristics limits the number of channels between 4 to 16. Let M denote the number of nodes in the network and C denote the number of multiple-access channels. The protocol design in this dissertation is targeted towards a system with many more nodes than the number of channels ($M \gg C$) for small values of C. A pre-allocation approach that does not utilize a control channel is considered here. This approach appears to be very promising due to its low implementational and operational complexity.

The simplest configuration in terms of system cost uses one tunable element and one fixed element. The protocols studied in this thesis are based on one tunable receiver and fixed receiver per node. The various choices for this configuration are examined in the next section.

2.3.4 Node Architecture

Each node in a WDM network requires at least one tunable transmitter or tunable receiver. A node can have more transmitter/receiver pairs that can be tunable or fixed. The particular configuration is determined chiefly by the access protocol. Fig. 2.5 shows a set of possible

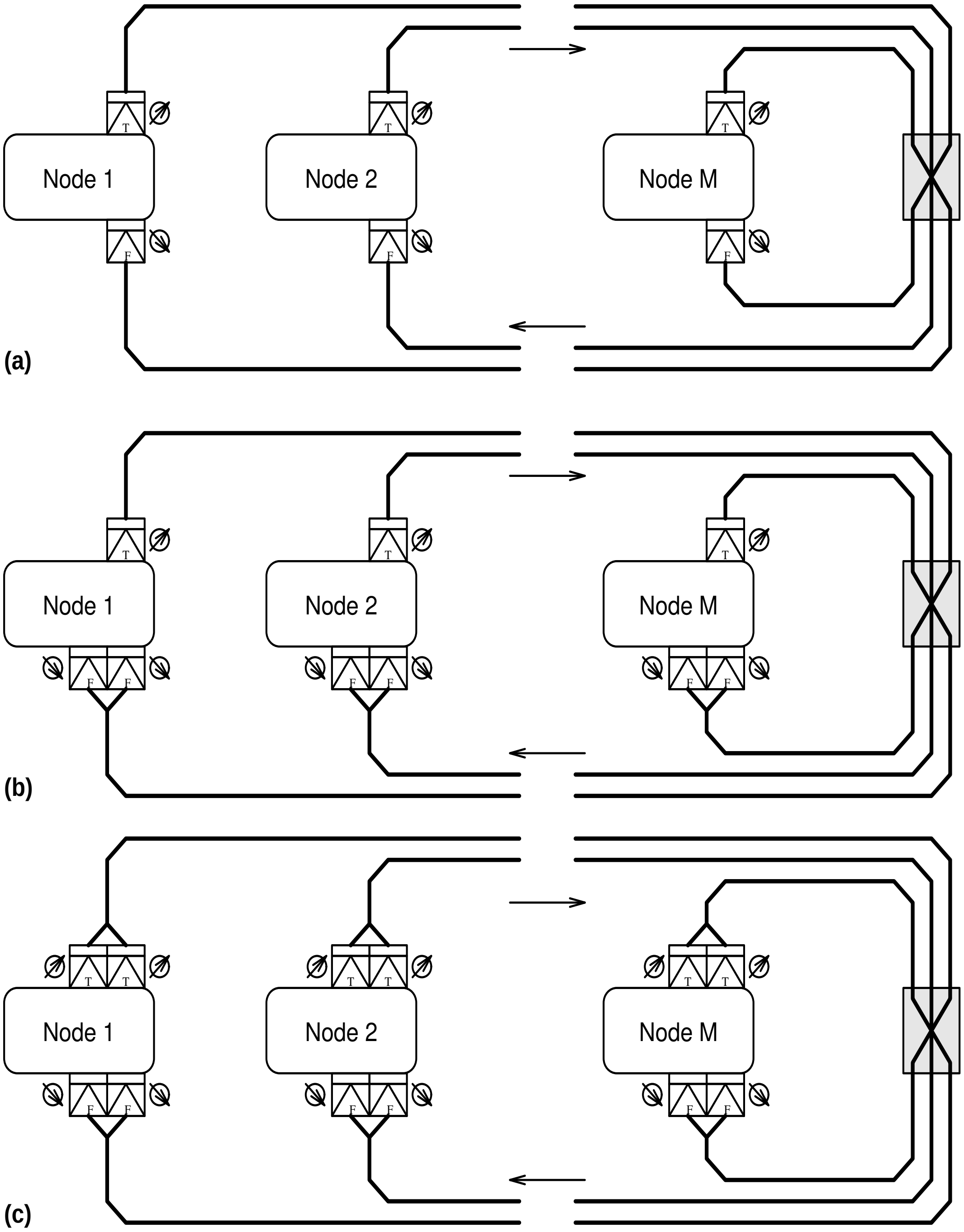

Figure 2.5: Transmitter/receiver configurations for a WDM network.

network architectures.

Fig. 2.5(a) shows a system where each node has a single transmitter and a receiver. There are three possible configurations with this organization: tunable transmitter and fixed receiver, fixed transmitter and tunable receiver, and tunable transmitter/receiver. One particular configuration uses a tunable transmitter and one fixed receiver [84]. Each node listens to one channel referred to as its *home channel*. The home channel allocation to the nodes can be determined based on a uniform policy or based on traffic requirements. This approach has potential for low cost since tunable receivers are more expensive than tunable transmitters or fixed receivers.

Fig. 2.5(b) shows a system with a tunable transmitter and two receivers. There are various choices depending on the tunability of the receivers. One particular configuration uses one fixed receiver and one tunable receiver per node as studied in [10]. The protocol is based on a control channel. The fixed receiver remains tuned to the control channel. The transmitter first sends a control packet on the control channel followed by data on one of the other channels.

Fig. 2.5(c) shows a system with a pair of transmitters and receivers. One possible configuration uses two fixed transmitters, one fixed receiver and one tunable receiver per node. The protocol based on this system [8] allocates uses one control channel. Each node is assigned a data channel on which it transmits data. The fixed receiver is tuned to the control channel. A backlogged node transmits a control packet in its slot in the next control cycle and then sends its packet on its own data channel. The receiver has to tune to that channel to receive the packet.

The above discussion presented the various choices for the node architecture. This dissertation examines protocols for the architecture shown in Fig. 2.5(a) with a single tunable transmitter and fixed receiver per node. This architecture is also used in a hierarchical network architecture used in a distributed shared memory (DSM) optically interconnected multiprocessor system [18, 85].

2.4 Summary

This chapter provided a background on WDM photonic networks. The OSI model of computer communication was presented and the significance of efficient media access protocols considered. Previous research on access protocols for WDM star networks and their classification was presented. The next chapter presents media access protocols studied for this dissertation for a passive-star coupled network.

Chapter 3

Access Protocols for WDM Star Networks

In this chapter, access protocols for a passive star-coupled WDM network are examined. A multiple access environment can be achieved through a variety of optical channel topologies [53] as shown in Fig. 2.2. Two main topologies studied for local networks are the bus and star. This chapter examines WDM star networks without photonic amplification. The star has excellent fanout and fault tolerance characteristics compared to the bus which forms the chief motivation for examining protocols for a passive star coupled WDM network.

Section 3.1 discusses the factors affecting protocol design and the impact of reservation and pre-allocation approaches on design and performance. Section 3.2 provides the description of the pre-allocation protocols studied as part of this dissertation. Section 3.3 provides the performance analysis of the protocols based on semi-Markov analytic models and discrete-event simulation models.

3.1 Factors affecting protocol design

Access protocols for the star have been classified as *reservation* and *pre-allocation* protocols as discussed in Section 2.3.3. The source node in a reservation protocol uses some form of pre-transmission coordination to determine the channel of communication. The pre-allocation protocols do not require such coordination and allocate channels to either receivers or transmitter. The factors identified below are used to examine the suitability of the pre-allocation protocols. In general, the problems are not new but have taken on increased importance due to the relation of their magnitude to the nominal data packet transmission time in high-speed networks:

Propagation delay: is the time required for light energy to travel from source to destination. It has been shown that large ratios of transmission time to propagation delay result in degraded performance [53]. Protocol design has to consider the impact of propagation delay on protocol performance and implementation.

For example, light requires approximately 30 ms to propagate from New York to Los Angeles assuming light is traveling at about two-thirds of its maximum speed. Transmission of a 1Mb packet requires only 1 ms with a transmission rate of 1 Gbps. For smaller distances, such as with a LAN, this concern still holds since the unit of transfer is also smaller (of the order of kilobits).

Reservation based schemes require control packets which are typically much smaller than data packets. The ratio of control packet transmission time to propagation delay is larger than that for data packets resulting in degraded performance. This has motivated the investigation of pre-allocation protocols which do not require control packets.

Processing Latency: is defined as the processing time preceding packet transmission. It has two main components of delay – the *tuning latency* of the optical devices and *protocol processing overhead*.

1. Tuning Latency is defined as the time spent in tuning the wavelength tunable transmitters and receivers(filters). Tunable transmitters with tuning speeds of 15 ns and tuning range of 50 channels have been demonstrated [37]. Tunable filters have been shown to have tuning speeds varying from μs to ns [25].

   The tuning latency of these optical devices is not negligible and has a significant impact on overall system performance [86, 87]. In general, faster tuning requires higher cost. An objective of this work is to achieve more cost effective network solutions by developing techniques that perform well with slower devices by hiding the processing latencies.

   The proposed protocols use a tunable transmitter and a fixed receiver. The tunable transmitter may be replaced by a laser array transmitter where each laser is always tuned to a single wavelength. This reduces the impact of transmitter tuning time between channels.

2. *Protocol processing overhead* is incurred for both incoming and outgoing packets. Incoming packets require opto-electronic conversion, header decoding and checksum verification. Outgoing packets require time to execute the protocol algorithm, to select the packet for transmission and to construct and append the header. The protocols should be simple to reduce the time spent in processing which might include buffer manipulation, arbitration algorithms, status table updates, etc.

Control channel based schemes require all nodes to process all control channel traffic. This places a very high demand on the receiver subsystem, in terms of its throughput capability. Pre-allocation based schemes completely eliminate this requirement on the receiver subsystem and partition the data traffic so that each node processes only a fraction of total traffic.

Synchronization: needs to provided at the bit, slot and frame or network level.

The network is configured to operate in a multi access environment and a concern is bit synchronization delay at the receivers. In a point-to-point or virtual point-to-point network, a receiver and transmitter are locked by requiring the transmitter to send special synchronization frames when there are no data packets. In a multi-access approach, a receiver has to receive a packet from a different node following a packet from another node. The receiver is required to establish bit synchronization and clock acquisition with the new transmitter as rapidly as possible.

There is an overhead per packet to establish the beginning of a packet transmission. A preamble is typically used to establish the start of transmission. Both reservation and pre-allocation approaches require bit synchronization of the data frames. The reservation approach has an additional requirement of providing bit synchronization on the control channel which may be significant due to the relative duration of the preamble and the control packet. In general, longer packets are preferred to reduce the overhead per data bit.

Most WDM media access protocols are slotted in time: pre-allocation slot access to the data channels, and reservation slot access to the control channel and usually slot access to the data channels. Reservation based schemes have an additional requirement of providing slot synchronization on the control channel.

The next issue in protocol implementation is the network synchronization mechanisms. This defines when each node accesses the network within a cycle. The objective is to develop a distributed synchronization algorithm with minimal centralized control. Several synchronization techniques have been studied for satellite systems using TDMA [88–90].

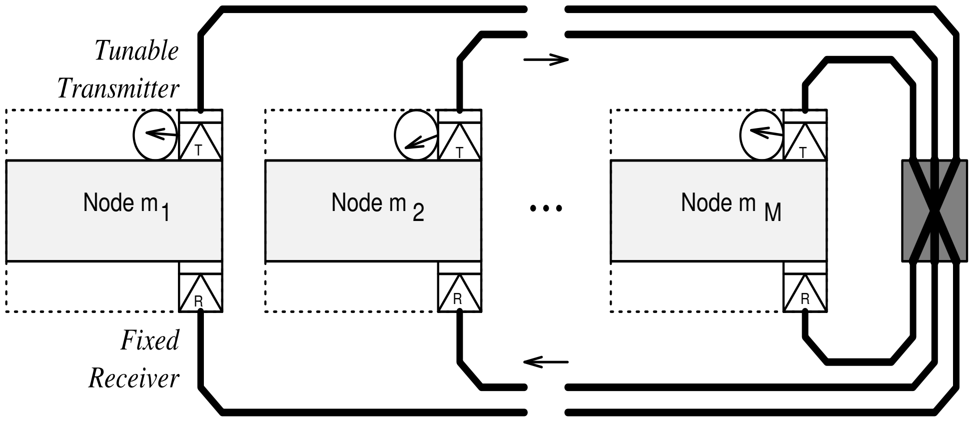

Figure 3.1: Network architecture based on the star topology with one tunable transmitter and fixed receiver per node.

The above discussion briefly compared reservation and pre-allocation in general terms. In particular, the motivation of the pre-allocation approach to remove the requirement of a control channel is described. The following sections examine the proposed protocols in terms of their sensitivity to propagation delay and processing latencies. The protocols do not have any additional advantage in terms of synchronization beyond the general benefit of pre-allocation that eliminates the requirement of a control channel.

3.2 Pre-Allocation Protocols

This section presents the design and analysis of a set of pre-allocation based protocols investigated as part of this dissertation. This section briefly describes the research and the complete details may be found in [84, 87, 91–93]. The following assumptions have been made in protocol design:

- ▷ The network architecture is based on the star topology with one tunable transmitter and one fixed receiver per node. Fig 3.1 shows the network architecture.

- ▷ All packets transmitted in the network are assumed to have the same length.
- ▷ System time is slotted on data packet boundaries. Packet transmission is synchronized to the start of each data slot.

Typical protocol operation is as follows: Each node's receiver remains tuned to its *home channel*. The source node with a backlogged packet tunes its transmitter to the destination's reception channel. The protocols differ in the mechanism in which different nodes access a particular channel. A source node can compute the home channel of the destination node independent of other nodes based on the channel allocation policy. Let M and C denote the number of nodes and the number of channels, respectively. Node m_j is assigned c_i as its home channel based on the allocation policy, where $0 \leq c_i \leq C-1$ and $0 \leq m_j \leq M-1$. The channel assignment scheme is termed *interleaved* when $i = j \bmod C$. The following paragraphs describe other parameters used to study system performance.

Processing latency (α): Let T denote packet transmission time and κ denote the sum of protocol processing time and maximum transmitter tuning time between channels. Processing latency, denoted by α, is defined as $\alpha \triangleq \left\lceil \frac{\kappa}{T} \right\rceil$. For $\kappa = 2\mu s$, packet transmission speed of 1 Gbps and packet size of 1000 bits, $\alpha = 2$.

Propagation delay (a): Propagation delay has been described in the previous section. The network may not be symmetric so all nodes may not be equidistant from the star. Let τ_i denote the propagation delay from node m_i to the star. Define τ as the maximum node-to-star propagation delay given by $\tau = \max\{\tau_i \mid 0 \leq i \leq M-1\}$, and normalized maximum propagation delay is denoted by $a \triangleq \left\lceil \frac{2\tau}{T} \right\rceil$ where T is the packet transmission time.

Media access schemes can be classified as *random access* and *static access* schemes as discussed in Section 2.3.2. The protocols initially investigated are two *random access* protocols, referred to as I-SA and I-SA* and two *static access* protocols referred to as I-TDMA and I-TDMA*. The proposed protocols have been named I-x to indicate that interleaved home channel allocation is used.

Protocol performance is studied in terms of *network throughput* and *average packet delay*. Network throughput is defined as the average number of packets transmitted per unit time. Packet delay is defined as the time elapsed between packet generation and successful transmission completion.

3.2.1 Random Access Protocols

In random access schemes, all nodes can access a channel at any time. A classic example of random access scheme is Slotted Aloha (SA). In SA, time is divided into equal length slots and a node can access the channel at the beginning of each slot. I-SA is a generalization of single-channel SA to the multi-channel WDM network.

I-SA [84] allows a data packet to be transmitted on the destination‘s home channel in the immediate slot following packet generation. Each node has a single queue of variable capacity to buffer packets generated when the transmitter is busy. The transmitter always attempts transmitting the packet at the head of the queue. Collision occurs when more than one node transmits on the same channel. Successful packet transmissions cannot be sensed by the source node since the receiver of the source node is tuned to its own home channel. Acknowledgments have to be sent by the destination to the sender to inform about successful reception.

If a collision occurs during packet transmission, the transmitter waits for a geometrically distributed number of slots before retransmitting. The transmitter decides to retransmit in subsequent slots with backoff probability denoted by η. SA based protocols have a potential instability problem due to high collisions under heavy traffic. Stabilizing mechanisms have been studied for SA to eliminate network collapse and provide a steady throughput [81].

Acknowledgments required in I-SA make the protocol potentially variable to propagation delay. The following paragraphs discuss two different acknowledgment schemes for I-SA.

Extended Acknowledgments: A packet slot in I-SA is composed of two phases with this approach referred as ESS [93]: data transmission phase and acknowledgment subslots. The acknowledgment subslot includes destination node transmitter tuning time and propagation delay for the acknowledgment. Nodes do not transmit data packets in the acknowledgment subslot to avoid collisions. If $M > C$, time multiplexing may be used to ensure collision-free acknowledgment transmission. Nodes resume data packet transmission at the end of the acknowledgment phase. This assumes that the acknowledgment information is small compared to the data packet size.

One I-SA data slot equals $(2a + \alpha + 1)T$ time units, so the channel utilization per successful slot is $\dfrac{1}{2a + \alpha + 1}$ since the initial tuning time can be overlapped. Note that the channel is being held by the source node transmitter during acknowledgment subslots. Since data transmission starts every $2a + \alpha + 1$ slots, there is an initial synchronization delay of $(2a + \alpha + 1)/2$ which may degrade performance when α or a is significant.

I-SA was first considered for a WDM-based optical backplane to support communication in a multiprocessor environment [94]. Propagation delay sensitivity was not initially a principal concern so acknowledgment was achieved by extending the slot to include the acknowledgment. However, in a local area environment, propagation delay is a significant concern. Also, the impact of the finite transmitter tuning time has to be considered. The protocol with ESS is very sensitive to propagation delay, protocol processing latency and tuning latency.

Explicit Acknowledgments: The acknowledgment subslots described above can be removed by requiring explicit acknowledgments from the destination node. The slot length is now equal to the normalized packet transmission time.

A generated data packet is stored in the buffer if the transmitter is busy. When the packet reaches the head of the queue, there is an initial wait before the transmitter decides to transmit the packet at the head of the queue. This is due to *transmitter contention* between the data packet and acknowledgment packets generated by the node.

Packet transmission is as follows: The data packet at the head of the queue is transmitted and the source node enters a *timeout* for the acknowledgment. The timeout period is estimated based upon a and α. The minimum timeout is $2a + \alpha + 1$ which includes propagation delay, transmitter tuning and transmission time for both data and acknowledgment. If the acknowledgment does not arrive by the end of its timeout period, the node follows a backoff policy as described for ESS. Upon successful receipt of the acknowledgment, the node transmits the next packet in the queue. Packet transmission is attempted until the data packet is successfully transmitted.

If a packet is successfully received by a node on its home channel *and* the packet is destined for the node, it generates an acknowledgment packet. The size of the acknowledgment is the same as that of the data packet. Each acknowledgment can provide feedback up to M/C nodes that receive it on their home channel. This is of particular advantage when there are many more nodes than channels. It also provides a form of redundant acknowledgment since all the M/C nodes that receive a packet may acknowledge a data packet.

Acknowledgments are subject to collision since they are transmitted as regular data packets. A data packet is unsuccessful if either the original data packet or the subsequent acknowledgment was lost due to collision. The acknowledgment can be piggybacked on the data packet but this is not considered here. Since there are at most M outstanding packets in the system, the scheme described above may be sufficient to support acknowledgments. Piggybacking plays a significant role when traffic at each node is pre-sorted on transmission channel.

The implementation details of the above two schemes and the performance analysis based on semi-markov and simulation models is presented in [93].

For a single channel network, the maximum utilization with SA for an infinite population system was analytically shown to be approximately 0.36 [81]. For a C channel system, theoretical throughput bound with I-SA is $0.36C$. However, this was *not observed* in the studies mainly because of the finite population model, a single transmitter and head-of-line effect due to a single queue. Also, one slot in I-SA is equal to $(2a + \alpha + 1)T$ where T is packet transmission time. When $(2a + \alpha) \geq 1$, channel utilization per slot is poor and is given by $\dfrac{1}{2a + \alpha + 1}$.

Improved I-SA: The performance drawbacks (poor utilization and head-of-line effects) of I-SA motivated the development of I-SA* [86, 87]. The improvements are:

- ▷ I-SA* eliminates the head-of-line effect of a single queue by providing a set of C separate queues per node – one per channel
- ▷ it eliminates the ACK subslots at the end of each packet transmission by requiring an explicit ACK from the destination nodes
- ▷ a node can transmit a window of packets after tuning to a channel thereby reducing processing latency per packet.

Discrete-event simulation models were developed to study the performance of I-SA*. The performance of I-SA* was much better than that of I-SA as reported in [86, 87]. The

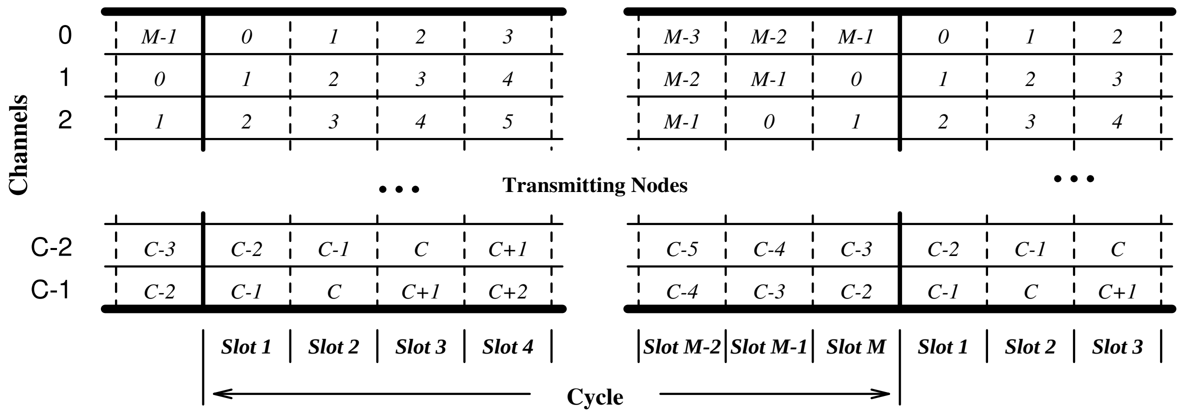

Figure 3.2: Allocation map for I-TDMA and I-TDMA* for $C < M$ and $\alpha = 0$.

head-of-line effects have been eliminated due to the multiple queues. The impact of tuning latency and propagation delay have been considerably reduced due to elimination of ACK subslots and transmission of a window of packets.

The following section describes the static access protocols studied for the star network.

3.2.2 Static Access Protocols

Static access schemes pre-determine the time instants a node may access the data channel. An example of such a scheme is Time Division Multiplexing (TDM) [81]. Time is divided into slots and exactly one node is allocated channel access during a slot. TDM avoids collisions and the complexity of supporting acknowledgments and retransmissions.

I-TDMA utilizes the TDM approach for scheduling access to the C channels. Time is slotted on each channel, and transmission is divided into cycles. Every node in the system has exactly one chance to transmit to a destination node on each channel per cycle (Fig. 3.2). The cycle length is M slots to allow each node to transmit once per cycle. Under heavy loads, all the slots in a cycle are used resulting in maximum utilization.

I-TDMA assumes a single queue per transmitter to buffer newly generated packets. The transmitter tunes to the channel required by the packet at the top of the queue and then transmits the packet. This tuning time can be overlapped with cycle synchronization time to reduce the impact of α [91].

Semi-markov analytic and discrete-event simulation models to analyze the performance of I-TDMA have been presented in [84,91]. I-TDMA is characterized by high delay under light traffic due to dependence of cycle length on the number of nodes. Under heavy traffic, it is stabler than I-SA and provides better performance. The theoretical bound for throughput has been shown to be 1 for a single-channel TDM system [81] (which is also the upper bound using any protocol). Similarly, the theoretical bound on throughput is C for a C-channel system. However, this was not achieved by I-TDMA. Head-of-line effects due to the single queue severely degraded the performance of I-TDMA [84]. This motivated the development of an improved version of I-TDMA called I-TDMA*.

I-TDMA* [92] is based on I-TDMA except that each node has C queues of variable capacity – one per channel. New packets are buffered in the corresponding channel queue. Fig. 3.2 shows the allocation map for $C < M$ and $\alpha = 0$. The source/channel mapping schemes vary with C, M and α and are described in detail in [86,87].

A node requires $\mathcal{T} = \alpha + 1$ time units to transmit a packet on a channel and a total of $C\mathcal{T}$ time units to transmit on all channels. The different allocation schemes are denoted by I-TDMA$^*_0$, I-TDMA$^*_1$ and I-TDMA$^*_2$. The cycle length for the schemes is M, $C\mathcal{T}$ and M respectively. The three schemes are described in detail below.

I-TDMA$^*_0$: This scheme is used when $C\mathcal{T} \leq M$ and cycle length in this scheme is $\mathcal{L} = M$ for $C < M$. If we assume a node does not transmit to itself, $\mathcal{L} = M - 1$ for $C = M$. The transmitter is active for $C\mathcal{T}$ slots and idle for $M - C\mathcal{T}$ slots in each cycle. However, all slots are utilized in each cycle. Performance is not degraded for all values of α which satisfy the condition $C\mathcal{T} \leq M$ since switching is completely overlapped with cycle synchronization.

The allocation scheme is described as follows. Node m_i is initially allocated channel c_0 at time $((i+1) \bmod M)$. Subsequent channels are allocated $\mathcal{T}$ units apart until all channels are allocated for this cycle. The node remains idle for $M - C\mathcal{T}$ slots after which channel 0 is allocated again. Since channels are allocated $\mathcal{T}$ units apart, the transmitter tunes to the next channel in this interval and a packet can be transmitted immediately without further tuning delay in the allocated slot.

Under light traffic, the average delay is $M/2$ slots irrespective of α and this includes cycle synchronization delay. Under heavy traffic, every node has a packet to transmit on each channel. Each slot of the cycle is utilized and maximum system throughput $\mathcal{S}_{max} = C$ is achieved under extreme load conditions. System throughput is defined as the number of packets transmitted per slot across all C channels.

If $C\mathcal{T} > M$, two ways of designing the allocation scheme are considered next. The first is an extension of I-TDMA$_0^*$ and results in an extended cycle. The second scheme retains the cycle length as M losing interleaving as a trade-off.

I-TDMA$_1^*$: This allocation scheme is an extension of I-TDMA$_0^*$ described above. The cycle is extended to allocate permission for each node to transmit on all channels and $\mathcal{L} = C\mathcal{T}$. Subsequent channels are allocated to a node $\mathcal{T}$ units apart as before. Transmitter tuning is overlapped with cycle synchronization and average packet delay under light traffic is $\mathcal{L}/2$. Note that this delay is only due to cycle synchronization and there is no tuning delay. Only M out of $\mathcal{L}$ time slots per channel cycle are utilized for data transmission. Maximum per channel utilization is therefore $\frac{M}{\mathcal{L}}$ and hence maximum system throughput is given by $\mathcal{S}_{max} = \frac{M}{\mathcal{T}}$.

The allocation map is identical to I-TDMA$_0^*$ except that the cycle length is $C\mathcal{T}$. Node

m_i is initially allocated channel c_0 at time $((i + 1) \bmod \mathcal{L})$ and subsequent channels are allocated $\mathcal{T}$ units apart. This scheme wastes slots; it is a tradeoff to retain a simple allocation scheme.

I-TDMA$_2^*$: The cycle length is retained as M in this scheme. After transmitting a packet on channel $c[j, i]$ during slot i, node m_j begins tuning to channel $c[j, 1 + (i + \alpha) \bmod (M)]$ if its transmit buffer for this channel is non-empty. The slots between i and $1 + (i + \alpha) \bmod (M)$ are not used and are lost since there must be at least α slots between consecutive transmissions.

Note that if $x(\alpha + 1) = M$ for some positive integer x, an extra slot must be skipped when cycle boundaries are crossed so the same set of channels is not visited each cycle to retain fairness.

This scheme works better than I-TDMA$_1^*$ under light traffic since cycle length is M compared to $C\mathcal{T}$ for I-TDMA$_1^*$. The average delay at lighter loads is $\alpha + \frac{M}{2}$ with I-TDMA$_2^*$ and $\frac{C\mathcal{T}}{2}$ with I-TDMA$_1^*$.

At higher traffic rates when queues are always full, a node can transmit at most $\left\lceil \frac{C}{\mathcal{T}} \right\rceil$ packets per cycle since each packet requires $\mathcal{T}$ time units. Maximum system throughput is therefore $\mathcal{S}_{max} = \left\lceil \frac{C}{\mathcal{T}} \right\rceil$ for $C < M$. I-TDMA$_1^*$ and I-TDMA$_2^*$ have identical maximum throughput of $\frac{M}{\mathcal{T}}$ when $C = M$. In this case, I-TDMA$_2^*$ has an advantage over I-TDMA$_1^*$ because of its smaller cycle length.

Analytic and discrete-event simulation models to predict the performance of I-TDMA* were studied in [87, 92]. I-TDMA* was observed to have dramatic performance improvements over I-TDMA as observed in [87]. It attains the maximum theoretical throughput (C) with C multi-access channels, is collision free and has low implementational complexity. Unlike I-SA and I-SA*, I-TDMA and I-TDMA* were seen to be insensitive to propagation

S_i:	State i, $0 \leq i \leq 4$ for I-TDMA $0 \leq i \leq 4$ for I-SA
τ_i:	expected sojourn time in state S_i
V_i:	limiting probability of being in state S_i of embedded markov chain
P_i:	limiting probability of being in state S_i of semi-markov process
β:	probability of at least one packet being generated in a slot
θ_1:	probability of no packet being generated in $M/2$ slots
θ_2:	probability of no packet being generated in t_w slots
λ:	Poisson packet generation rate per node
P_s:	probability of successful packet transmission in I-SA
P_b:	probability of leaving backoff in I-SA
t_w:	Expected Full wait time in I-TDMA
$\mathcal{D}_{i-sa}$:	average delay per packet in I-SA
$\mathcal{D}_{i-tdma}$:	average delay per packet in I-TDMA
$\mathcal{S}_{i-sa}$:	System Throughput in I-SA
$\mathcal{S}_{i-tdma}$:	System Throughput in I-TDMA

Table 3.1: Notation used in analysis of the I-SA and I-TDMA.

delay.

3.3 Performance Analysis

A detailed performance analysis of the protocols was conducted using semi-markov analytic models and discrete-event simulation models. A semi-Markov model was developed in [95] for a multibus system. In particular, it developed the three state cluster which is inherent in the model developed below. A semi-Markov model is an approximation technique, useful because of the significant reduction in state space, and a comparison to simulation is used to ensure that the implied approximations are reasonable. This approach achieves a significant reduction in state complexity over standard markov models and improved accuracy over arrival blocking semi-markov models such as introduced in [95]. The semi-markov models allow modeling with arbitrary holding times [96].

The simulators are based on stochastic self-driven discrete event models, written in the C programming language with SimPack. SimPack is a C based library of routines that provides discrete-event and random variate facilities. Steady state transaction times and throughput were measured. Simulation convergence was obtained through the replication/deletion method [97], with a 95% confidence in a less than 5% variation from the mean. Later simulations were written using a process-oriented simulation package YACSIM [98]. The simulation model has been used to validate the analytic models. In certain cases, discrete-event simulation has been used to evaluate performance to ensure a consistent evaluation and avoid the simplification assumptions often required by analytic models.

The next sections describe analytic models for I-SA and I-TDMA. Table 3.1 describes the notation used in the following analysis. The system is described in terms of normalized time. Time is normalized to packet transmission time. The model for the single-level protocol makes use of the following assumptions:

$\mathcal{A}_1$: *i.i.d.* behavior of the nodes and channels. Channel traffic will not be uncorrelated in general, due to common transmit queues, but this approximation simplifies the transition probability derivations and is shown to be reasonable in the comparison to simulation in Section 3.3.4.

$\mathcal{A}_2$: Time is slotted on packet boundaries and packet generation per node follows a Poisson process.

$\mathcal{A}_3$: The traffic distribution is largely determined by reference of locality. The uniform reference model is used for performance evaluation. A packet generated a node m_i

is targeted to node m_j with probability $\dfrac{1}{M-1}$ for $i \neq j$ and $0 \leq i, j \leq M-1$; and with probability 0 for $i = j$.

$\mathcal{A}_4$: A transmitter has a capacity to hold one packet when processing another packet. This is not an inherent assumption of the protocol and the model can be extended to include generalized transmitter buffer capacities as in [92].

3.3.1 Semi-markov model for I-SA

State Definitions: The receiver and transmitter at each node can change state during every slot. The receiver is passive in that it does not initiate activity. It is idle unless a packet is directed along its home channel. The possible states then are: *idle* – this node is not currently receiving a data packet or a packet was transmitted along the channel but a collision occurred, and *receiving* – the node is currently receiving a packet.

The transmitter is the component that changes states most often. The possible states and their description are: *idle* - transmitter is idle as long as there is no packet generated at the node, *transmit* - transmitting the data packet, and *backoff* - the transmitter must wait a number of slots before attempting retransmission.

The receiver and transmitter at each node function simultaneously. Their functions are distinctly defined. From the above details of the states of the receiver and transmitter, it is seen that there is no conflict in their operation. The model need only represent the states of the transmitter to characterize the behavior of the protocol.

The model is based on finite transmitter buffers, queuing at most one packet while the transmitter is processing another packet. The model focuses on the behavior of a single node, which is taken to be typical of the behavior of each node in the system. The state diagram of the process depicting the behavior of one node is shown in Fig. 3.3, and the

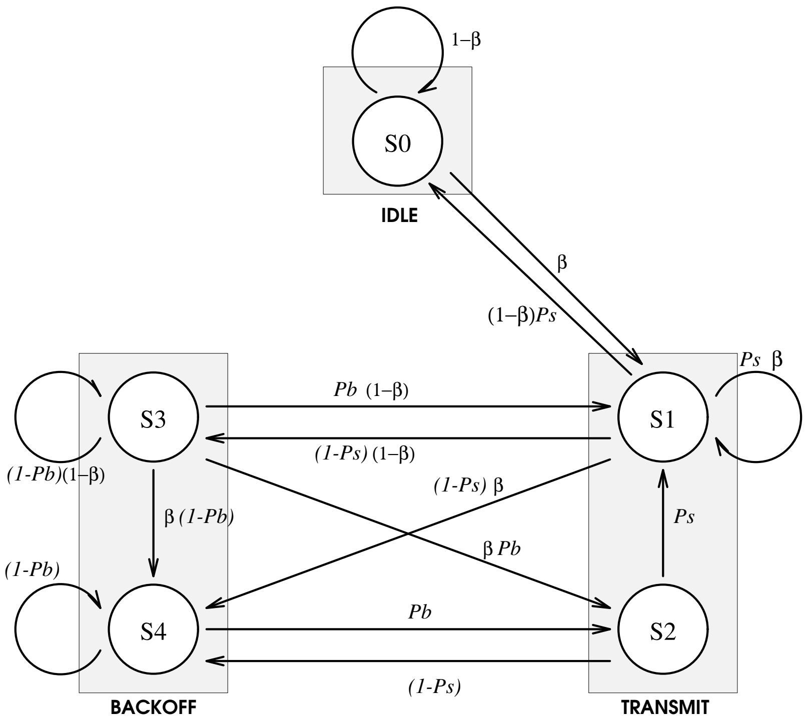

Figure 3.3: State diagram of the semi-markov process for analyzing the I-SA protocol.

states are defined as follows:

S_0: transmitter is idle;

S_1: transmitter is transmitting a packet, no additional packet is buffered;

S_2: transmitter is transmitting a packet, one packet is buffered;

S_3: transmitter in backoff, no additional packet is buffered;

S_4: transmitter in backoff, one packet is buffered.

Since the sojourn time at each state is unity, the semi-markov model is actually a discrete-time markov chain providing the Poisson arrival assumption is retained. A discrete-time markov chain is a special case of a general semi-markov model because of its *unit* state holding times [96]. A semi-markov model is more general and allows arbitrary state holding times.

Define $\beta = 1 - e^{-\lambda}$ as the probability of at least one packet arrival in one slot where

λ is the Poisson arrival rate per node; let P_s denote the probability of successful packet transmission and let P_b be the probability of leaving backoff state and attempting retransmission (Table 3.1). The transition probability matrix P is given below, where p_{ij} indicates the probability of transition from state S_i to S_j:

$$P = \begin{pmatrix} \overline{\beta} & \beta & 0 & 0 & 0 \\ P_s\overline{\beta} & P_s\beta & 0 & (1-P_s)\overline{\beta} & (1-P_s)\beta \\ 0 & P_s & 0 & 0 & (1-P_s) \\ 0 & P_b\overline{\beta} & P_b\beta & (1-P_b)\overline{\beta} & (1-P_b)\beta \\ 0 & 0 & P_b & 0 & (1-P_b) \end{pmatrix}$$

where $\overline{\beta} = 1 - \beta$.

States S_0 and S_4 have self-loops with deterministic sojourn times of 1. They can be replaced by states with a geometric sojourn time distribution with parameters $1/\beta$ and $1/P_b$ respectively, in which case the markov chain becomes a semi-markov model. The self-loops are retained because they aid in clarity of the model without any additional complexity.

Limiting Probabilities: The limiting probabilities of being in S_i of the embedded markov chain is denoted as V_i and can be obtained by solving the steady state equations. From the state diagram of Fig. 3.3 and the transition probabilities discussed above, the steady state equations are obtained. The limiting probability of being in state S_i of the semi-markov is denoted as P_i and is given by

$$P_i = \frac{V_i \tau_i}{\sum_{j=0}^{4} V_j \tau_j}$$

Since the sojourn times of all the states are 1, i.e. $\tau_i = 1, \forall i$, $P_i = V_i$. The above linear equations can be solved to yield the vector $\{P_i\}$ in terms of P_s. P_s in turn depends on $\{P_i\}$ as shown in [84]. The set of equations are solved iteratively to converge on P_s and $\{P_i\}$, which are then used to obtain the packet delay and the network throughput.

Probability of Successful Packet Transmission

The probability of successful packet transmission is an important parameter in I-SA. This factor is dependent on the probability that more than one packet is concurrently transmitted on the same channel. The probability that a node is in one of its transmitting states can be obtained from the model. States $S_i, \forall i : 1 \leq i \leq 2$ are the transmitting states. Let $\sigma = \sum_{i=1}^{i=2} P_i$ denote the probability that any node is in one of its transmitting states. The probability of successful packet transmission is based on two cases: the probability that the packet is directed to a node that shares its home channel with the source node; and the probability that the packet is directed to a node whose home channel is different from the home channel of the source node. Let A be the event that the home channel of the destination node is the home channel of the source node, B be the event that the home channel of the destination node is not the home channel of the source node.

1. Since M/C nodes share the same channel, the probability of event A occuring $P[A] = \dfrac{M-C}{C}\left[\dfrac{1}{M-1}\right]$. Successful packet transmission occurs if the remaining $\frac{M}{C} - 1$ nodes sharing the source node home channel and the other $\frac{M(C-1)}{C}$ nodes in the system do not transmit on the channel selected by the source node. The probability of success given event A occurs is: is given by:

$$\gamma_h = \frac{(M-C)}{C(M-1)}\left[1 - \frac{\sigma M}{C(M-1)}\right]^{\frac{M(C-1)}{C}}\left[1 - \frac{\sigma(M-C)}{C(M-1)}\right]^{\frac{M}{C}-1} \tag{3.1}$$

2. The probability of event B occuring $P[B] = \dfrac{M}{C(M-1)}$. The packet will be successfully transmitted if the M/C nodes which share the home channel and the other other $\frac{M(C-1)}{C} - 1$ do not transmit on the channel selected by the source node. The

probability of success given event B occurs is:

$$\gamma_o = \frac{M}{C(M-1)}\left[1-\frac{\sigma M}{C(M-1)}\right]^{\frac{M(C-1)}{C}-1}\left[1-\frac{\sigma(M-C)}{C(M-1)}\right]^{\frac{M}{C}} \tag{3.2}$$

The total probability of successful packet transmission is given by

$$\gamma = \gamma_h + (C-1)\gamma_o \tag{3.3}$$

If the assumption that a node cannot transmit to itself is removed this expression simplifies to: $\gamma = (1-\sigma/C)^{M-1}$.

Iterative Algorithm for I-SA

The probability of successful packet transmission (γ) depends on the limiting probabilities of being in the transmitting states in the semi-markov process (P_i). The limiting probabilities are dependent on γ and are computed iteratively using the following algorithm.

1. Choose an initial value for γ $(0 < \gamma < 1)$.
2. Compute the transition probabilities using the above value of γ.
3. An improved estimate of γ is computed using expression given by Eqns. (3.1)–(3.3).
4. Repeat steps (2)-(3) until γ has converged.

3.3.2 Semi-Markov model for I-TDMA

State Definitions: This section develops the model used for the analysis of I-TDMA protocol. The basic model is similar to the one developed in the previous section for I-SA. The possible states of the transmitter and their description are: *idle* – transmitter is idle as long as there is no packet generated at the node, *full wait* – transmitter waits for its place in the control cycle after a packet is generated at the node for transmission, and *transmit* – transmitting the data packet. As with I-SA, the model need only represent the behavior of the transmitter.

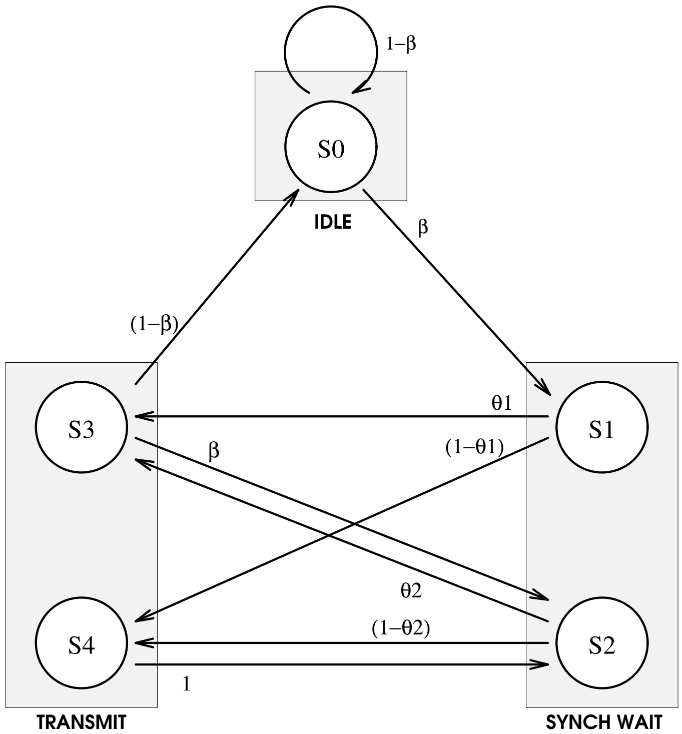

Figure 3.4: State diagram of the semi-markov process for analyzing the I-TDMA protocol.

The state diagram of the semi-markov process depicting the behavior of one node in the system implementing the I-TDMA protocol is shown in Fig. 3.4, and the states are defined as follows:

S_0: transmitter is idle;

S_1: transmitter is in full wait state – transition from *idle* state;

S_2: transmitter is in full wait state – transition from *transmit* state;

S_3: transmitter sending data packet – no additional packet is buffered;

S_4: transmitter sending data packet – one packet is buffered.

The expected sojourn times of the states are given by $\tau_0 = 1$, $\tau_1 = M/2$, $\tau_2 = t_w$ and $\tau_3 = \tau_4 = 1$. The transmitter can change from the idle state at the beginning of each slot, in the event of a packet arrival in the previous slot, so the sojourn time is $\tau_0 = 1$. The sojourn time in S_1 is the synchronization wait after the transmitter changes from the idle state. The expected sojourn time, τ_1 is therefore given by $M/2$. S_2 is also a synchronization state, where the transmitter waits for its turn in the I-TDMA allocation map when the previous state was a transmission state. The expected sojourn time for S_2, given by $\tau_2 = t_w$, is dependent on the number of nodes and channels in the system and the allocation scheme derived later. S_3 and S_4 are packet transmission states and their sojourn times are 1.

Define $\beta = 1 - e^{-\lambda}$ as the probability of at least one packet arrival in one slot; let $\theta_1 = e^{-\lambda M/2}$ be the probability of no packet arrival in $M/2$ slots and $\theta_2 = e^{-\lambda t_w}$ be the probability of no packet arrival in t_w slots (Table 3.1). The transition probability matrix is given below, where p_{ij} indicates the probability of transition from state S_i to S_j:

$$P = \begin{pmatrix} (1-\beta) & \beta & 0 & 0 & 0 \\ 0 & 0 & 0 & \theta_1 & (1-\theta_1) \\ 0 & 0 & 0 & \theta_2 & (1-\theta_2) \\ (1-\beta) & 0 & \beta & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

These transition probabilities are then used to compute the limiting probability of being in a state in the embedded markov chain of the semi-markov model. The limiting probabilities of being in the states of the semi-markov model are then computed to derive the average packet delay and network throughput.

Limiting Probabilities: The limiting probabilities of being in S_i in the embedded markov chain of the above model, denoted as V_i, can be obtained by solving the steady state equations. From the state diagram of Fig. 3.4 and the transition probabilities discussed above, the steady state equations are obtained. The limiting probability of being in state S_i of the semi-markov process [96] is now given by:

$$P_i = \frac{V_i \tau_i}{\sum_{j=0}^{4} V_j \tau_j}$$

Full Wait Time

The full wait time for a generated packet before it is actually transmitted depends on whether the transmitter is currently in the idle state or in the transmitting state. If the transmitter is in the idle state when the packet is generated, then the packet has to wait an average of $M/2$ slots before it is transmitted. If the transmitter is in the transmitting state when the packet is generated, the packet has to wait for a time that is dependent on the channel that the current packet is being transmitted and the channel for which the new packet is directed. This section derives the full wait for a packet in the buffer given that a packet is transmitted in the current slot. The full wait time is derived assuming that the home channel of a node is 0. This expression is valid for any other channel due to symmetry. This derivation takes into account the fact that a node will not transmit to itself.

Let i be the channel on which the current packet was transmitted, and j be the channel to which the next packet is directed. The delay can be represented by a matrix D_{ij} and is given below for $i, j = 0, \ldots, C-1$.

$$\begin{pmatrix} M-1 & 0 & 1 & 2 & \ldots & C-2 \\ M-2 & M-1 & 0 & 1 & \ldots & C-3 \\ \vdots & & & & & \vdots \\ M-C & M-C+1 & & \ldots & & M-1 \end{pmatrix}$$

The channel to which the next packet is directed is independent of the channel on which the current packet is being transmitted. This is valid because of the uniform reference model given in Section 3.3. The average wait for the next packet conditioned on the probability that the current channel is i is given by:

$$\begin{aligned} P[W_i \mid i] = & \\ & \frac{M}{C(M-1)}\left[\sum_{j=0}^{i}(M-1-i+j) + \sum_{j=i+1}^{C-1}(j-1-i)\right] \\ & - \frac{M-1-i}{M-1} \\ = & \frac{i}{M-1} + \frac{M(C-1)}{2(M-1)} + \frac{(M-C)M(i+1)}{C(M-1)} - 1 \end{aligned}$$

The probability that the current channel is i is given by:

$$P[i] = \begin{cases} \frac{M-C}{C(M-1)} & \text{if } i = 0 \\ \frac{M}{C(M-1)} & \text{if } i = 1, 2, \ldots, C-1 \end{cases}$$

The expected wait time for a packet directed to any channel is given by

$$\begin{aligned} t_w &= \sum_{i=0}^{C-1} P[W_i \mid i] P[i] \qquad (3.4) \\ &= \frac{M(M-C)((M(C+1))-2)}{2C(M-1)^2} \end{aligned}$$

$$+\frac{M^2(C-1)}{2(M-1)^2} - 1 \qquad (3.5)$$

If the assumption that a node does not transmit to itself is removed, the expression for t_w can be simplified to $t_w = \frac{M}{2}\left[\frac{C+1}{C}\right] - 1$.

Average Queue Occupancy

The average packet delay in I-TDMA and I-SA is derived using Little's law. The average number of packets present in a queue over a time interval is to be determined. Consider a queue of capacity B with i of these slots filled. Let λ be the Poisson arrival to the queue and T the number of time slots. Packets are assumed to arrive only at the beginning at a time slot. The expected number of packets in the queue at the end of time t is given by $N_t = \min[i + \lambda t, B]$. The average queue occupancy is given by:

$$\mathcal{F}[i, B, \lambda, T] = \frac{1}{T}\sum_{t=1}^{t=T} N_t \qquad (3.6)$$

The following section derives expressions for the average packet delay and network throughput from the models presented above.

3.3.3 Computation of Performance Metrics

The performance metrics being analyzed using the models described in the previous sections are the average packet delay ($\mathcal{D}_{i-sa}$ and $\mathcal{D}_{i-tdma}$), and total network throughput ($\mathcal{S}_{i-sa}$ and $\mathcal{S}_{i-tdma}$).

Network Throughput

The percent of time that a node spends transmitting data packets successfully gives the utilization of the data channels for that node. The network throughput equals the total channel utilization of all C channels since time is normalized to the data packet transmission time. Throughput is defined as the total number of packets successfully transmitted per slot across all data channels.

I-SA: A node transmits data on a data channel in states S_1 and S_2 of the model. The number of packets transmitted by a node is determined by the probability that the process is in a transmitting state and the probability of successful packet transmission. The total throughput is given by the following equation:

$$\mathcal{S}_{i-sa} = M(P_1 + P_2)P_s \tag{3.7}$$

I-TDMA: The data channels are utilized in states S_3 and S_4 of the process for the I-TDMA protocol. The total throughput is given by the following equation:

$$\mathcal{S}_{i-tdma} = M(P_3 + P_4) \tag{3.8}$$

Average Packet Delay

The packet delay is the time taken from the instant a packet is generated at the source node to the instant it is received at the destination node. This includes the waiting time, backoff time (with I-SA), synchronization time, packet transmission time, and delay due to retransmissions (with I-SA). The packet delay in I-TDMA protocol includes the full wait time and the packet transmission time.

The packet delay is obtained by applying Little's Law to a node in the system: $\mathcal{D} =$

$E[N]/\Gamma_n$, where $\mathcal{D}$ is the average packet delay at a node, $E[N]$ is the average number of packets (in buffer and in transmission) at a node, and Γ_n is the per node throughput ($\Gamma_n = \mathcal{S}/M$ where $\mathcal{S}$ is the system throughput).

I-SA: The expected sojourn times of the states in the model for I-SA are $\tau_i = 1$, for all $i = 0, \ldots, 4$. The average number of packets in the system, and average delay is then given by:

$$\begin{aligned} E[N] &= \sum_{i=0}^{4} E[N_i]P_i \\ \mathcal{D}_{i-sa} &= \frac{E[N]}{(P_1 + P_2)P_s} \end{aligned} \tag{3.9}$$

where $E[N_i]$ is the expected number of packets at the node when the process is in state S_i and P_i is the probability of being in state S_i. $E[N_i]$ can be obtained from the model as: $E[N_0] = 0$, $E[N_1] = 1$, $E[N_2] = 2$, $E[N_3] = 1$ and $E[N_4] = 2$.

I-TDMA: The packet delay for I-TDMA is also derived using Little's Law. The expected sojourn times of the states in the model are given by the vector $\tau = \{1, M/2, t_w, 1, 1\}$, and the average number of packets per node is given by:

$$\begin{aligned} E[N] &= \sum_{i=0}^{4} E[N_i]P_i \\ \mathcal{D}_{i-tdma} &= \frac{E[N]}{(P_3 + P_4)} \end{aligned} \tag{3.10}$$

The model is used to obtain the expected number of packets in the system $E[N_i]$ when the process is in state S_i as: $E[N_0] = 0$, $E[N_3] = 1$, $E[N_4] = 2$. The number of packets in the buffer in the above states is deterministic since the sojourn times are 1. When the sojourn time of any state in the semi-markov model is larger than 1, the number of packets buffered at a node during that state has to be derived based on the packet arrival process. When a node enters the full wait states S_1 and S_2, there is already one packet at the node. During the sojourn time of the state, more packets can be generated by the node of which exactly one is buffered and the remaining are discarded. $E[N_1]$ and $E[N_2]$ are the expected number of packets in the buffer at the end of time periods τ_1 and τ_2 respectively. The expected number of packets buffered at the end of time T, is given by Eqn. 3.6.

3.3.4 Validation of Analytic Models

Validation of the semi-markov models developed to analyze the I-SA and I-TDMA protocols is obtained through simulation. The simulators are based on stochastic self-driven discrete event models, written in the C programming language with SimPack. SimPack is a C based library of routines that provides discrete-event and random variate facilities. Steady state transaction times and throughput were measured. Simulation convergence was obtained through the replication/deletion method [97], with a 95% confidence in a less than 3% variation from the mean.

I-SA: Fig. 3.5 plots the comparison of the simulation results with the results predicted by the analytic model for the I-SA protocol. Lines correspond to the analytic model and points denote simulation. The graph in Fig. 3.5(a) depicts average packet delay and Fig. 3.5(b) illustrates the network throughput for varying system sizes and varying number of data channels: $M \in \{8, 16, 32\}$ and $C \in \{M, M/2, M/4\}$ with variations in packet generation rate. A value of $P_b = 0.05$ has been used as the probability of leaving the backoff state to attempt retransmission in simulation and analysis. It can be seen that the results obtained through simulation and analysis predict similar behavior of the protocol. The deviation of delay between the analytical and simulated results was less than 5%. The deviation of the

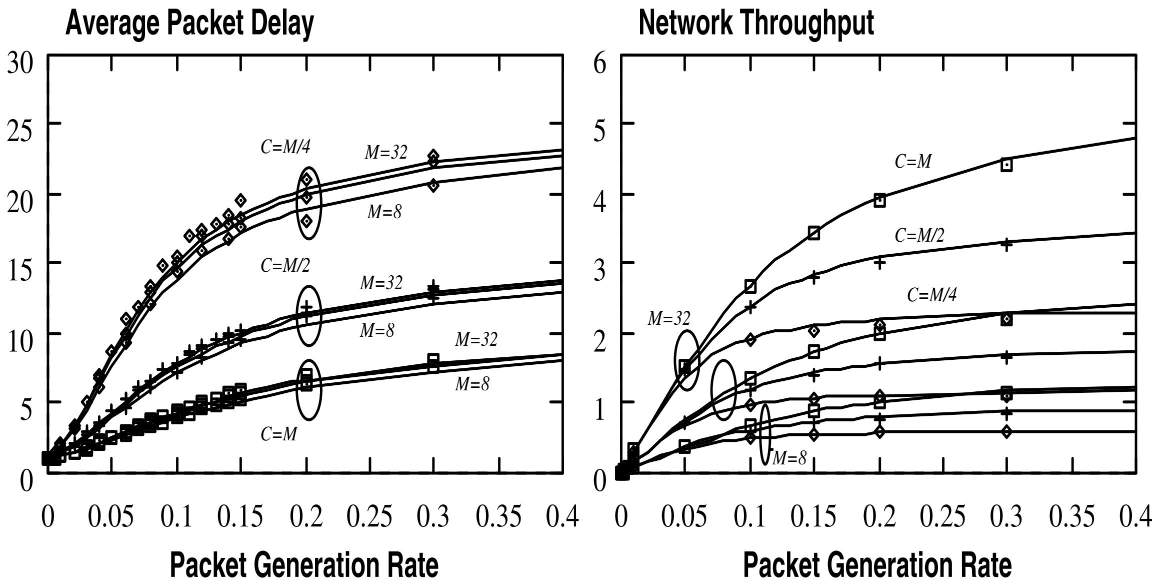

Figure 3.5: Validation of I-SA for $M \in \{8, 16, 32\}$ and $C \in \{M, M/2, M/4\}$: (a) Average Packet Delay, (b) Network Throughput. Points and lines represent simulation and analysis respectively.

total system throughput as computed by the model from that obtained through simulation was less than 3%.

Fig. 3.5(a) shows that the average packet delay decreases by 65% and the total system throughput increases by 100% when the number of channels is increased from $M/4$ to M at $\lambda = 0.3$ when $M = 32$. Average packet delay is reduced due to a reduction in collisions since the load per channel is decreased. The graphs show that the packet delay of I-SA is insensitive to increases in system size provided that the ratio of M/C is maintained. This shows that the system is scalable up to a point: the total system traffic increases as the number of nodes increases, but a level of performance can be maintained if the number of channels are increased proportionally to the increase in nodes. The maximum number of channels are determined by device and cost characteristics while the maximum number of nodes are determined by the optical power budget.

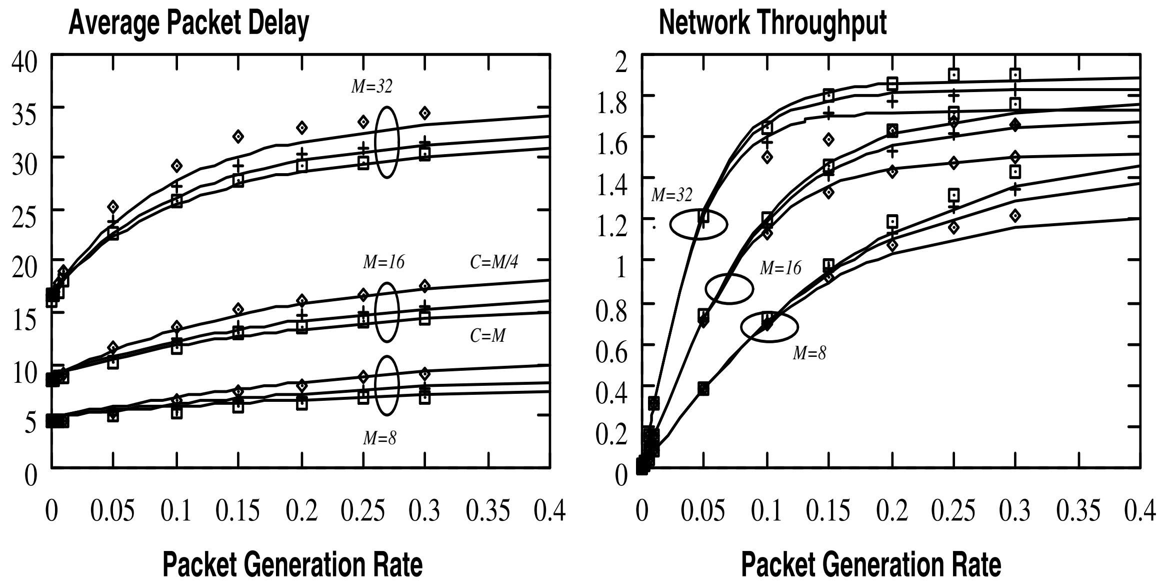

Figure 3.6: Validation of I-TDMA for $M \in \{8, 16, 32\}$ and $C \in \{M, M/2, M/4\}$: (a) Average Packet Delay, (b) Network Throughput. Points and lines represent simulation and analysis respectively.

I-TDMA: Fig. 3.6 shows the plots comparing the results obtained through analysis and simulation for the I-TDMA protocol. Lines correspond to the analytic model and points denote simulation. Fig. 3.6(a) plots the variations in the average packet delay and Fig. 3.6(b) plots the variations in the network throughput. The plots are for $M \in \{8, 16, 32\}$ with variations in the number of data channels $C \in \{M, M/2, M/4\}$. The deviation of the delay obtained through the analytic model from the simulation results was less than 8%, and the deviation in network throughput was less than 8%.

From Fig. 3.6(a) it can be seen that the minimum average packet delay is $M/2$, which is one half of the cycle length, when the packet generation rate is very low. A node must wait for its position within the round before it may transmit even if no other node in the system has a packet to be transmitted. This is the *synchronization delay* due to the nature of I-TDMA.

The cycle length is proportional to the number of nodes in the system, so it is expected that this protocol will be more sensitive to increases in M than I-SA. Fig. 3.6 verifies this expectation and shows that the performance in terms of delay and throughput cannot be maintained with increases in M even if the M/C ratio is held constant. For example, the delay increases by 395% when a system is expanded from $M = 8$ and $C = 8$ to $M = 32$ and $C = 8$ at $\lambda = 0.3$. However, the delay still increases by 346% if the ratio of M/C is maintained and a system is expanded from $M = 8$ and $C = 8$ to $M = 32$ and $C = 32$ at $\lambda = 0.3$.

From Fig. 3.6(b), the variation in network throughput with variations in M and C can be studied. As the number of channels increases, the throughput increases. However, the graph shows that this protocol cannot take advantage of an increase in channels since the number of nodes tends to dominate the resulting performance. For example, an increase of only 14% in network throughput is achieved for a system of $M = 32$ when the number of channels is increased from $C = 8$ to $C = 32$ at $\lambda = 0.3$.

3.3.5 Comparison of I-SA and I-TDMA

This section compares the relative behavior of the two protocols in terms of average packet delay and throughput with variations in the number of nodes, channels and arrival rate.

Variation in Data Channels

Fig. 3.7(a) compares the delay and throughput of the two protocols for a system of $M = 32$ nodes with varying channels $C \in \{2, 4, 8, 16, 32\}$.

As the number of data channels is increased from 2 to 32, the maximum network throughput of the I-TDMA protocol increases by a maximum of 25%. Fig. 3.7 shows I-SA is able to take greater advantage of an increase in channels. As C increases, the load

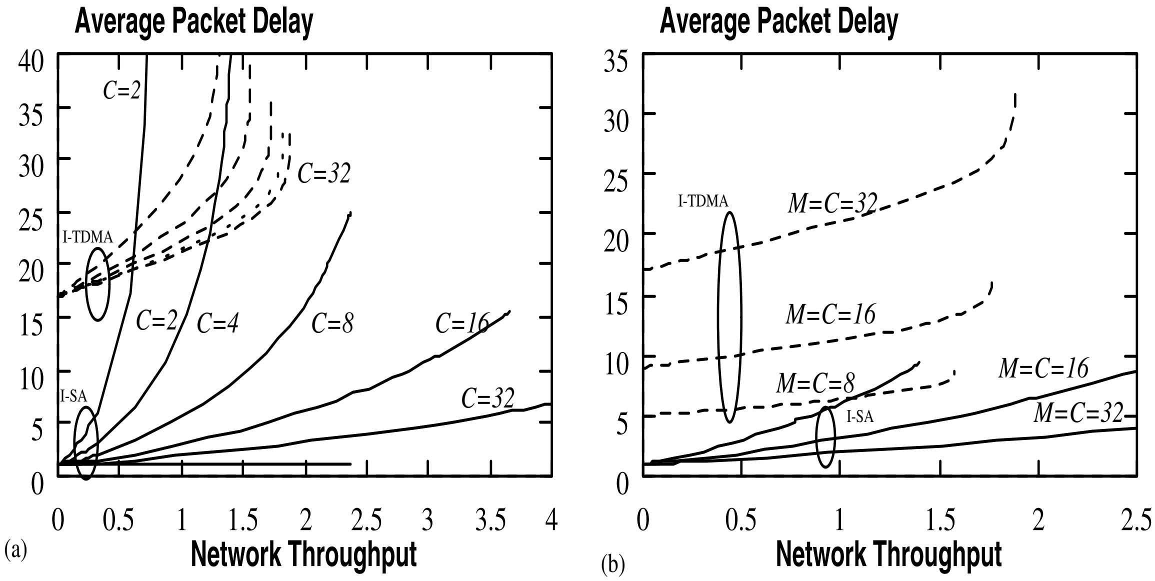

Figure 3.7: Comparison of delay versus network throughput for I-SA and I-TDMA: (a) $M = 32$ and $C \in \{2, 4, 8, 16, 32\}$, (b) $M = C$ for $C \in \{8, 16, 32\}$.

per channel decreases and fewer nodes share a channel thereby reducing collisions and increasing the network throughput. In I-TDMA the performance is dominated by the cycle time which is proportional to M and significant improvement in system performance is not obtained with an increase in the number of channels.

It can been observed from Fig. 3.7 that the choice of a suitable protocol for a system depends on the ratio M/C in addition to the expected traffic load. As expected, the random access protocol has superior performance for all configurations when the system is lightly loaded. As the traffic level increases, the choice in protocol is determined by the ratio. For a low ratio, $M/C < 8$, I-SA is shown to perform well even with heavy traffic. When the ratio increases so $M/C \geq 8$, where the number of nodes greatly exceed the number of data channels in the system, I-TDMA is shown to have better capacity.

Variation in System Size

Fig. 3.7(b) shows the comparison of the two protocols for a system with $M \in \{8, 16, 32\}$ and $C = M$. The graph illustrates that the I-SA protocol achieves superior performance in all cases other than $M = C = 8$. For the small system size of $M = C = 8$, the I-SA protocol has superior performance when the system is lightly loaded but the I-TDMA protocol is shown to have greater capacity before saturation.

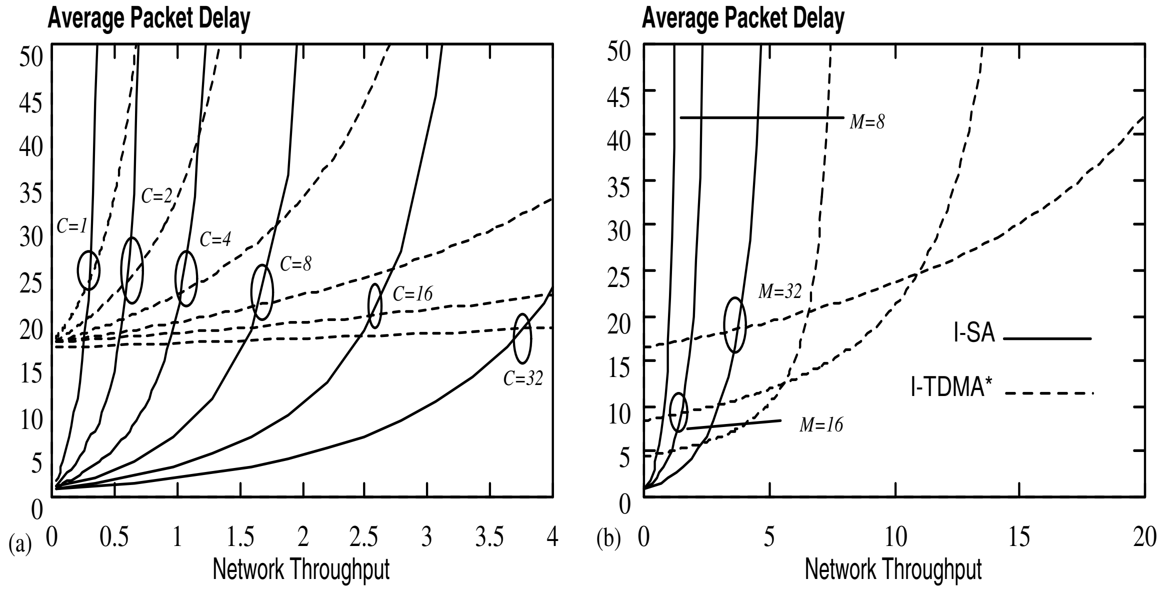

Figure 3.8: Comparison of I-SA and I-TDMA* for $B = 100$: (a) $M = 32$ and $C \in \{1, 2, 4, 8, 16, 32\}$, (b) $C = M$ and $M \in \{8, 16, 32\}$.

3.3.6 Comparison of I-SA and I-TDMA*

This section compares the relative behavior of I-SA and I-TDMA* in terms of average packet delay and throughput with variation in the number of nodes, channels and arrival rate. Define $\mathcal{S}_{max}$ as the maximum network throughput.

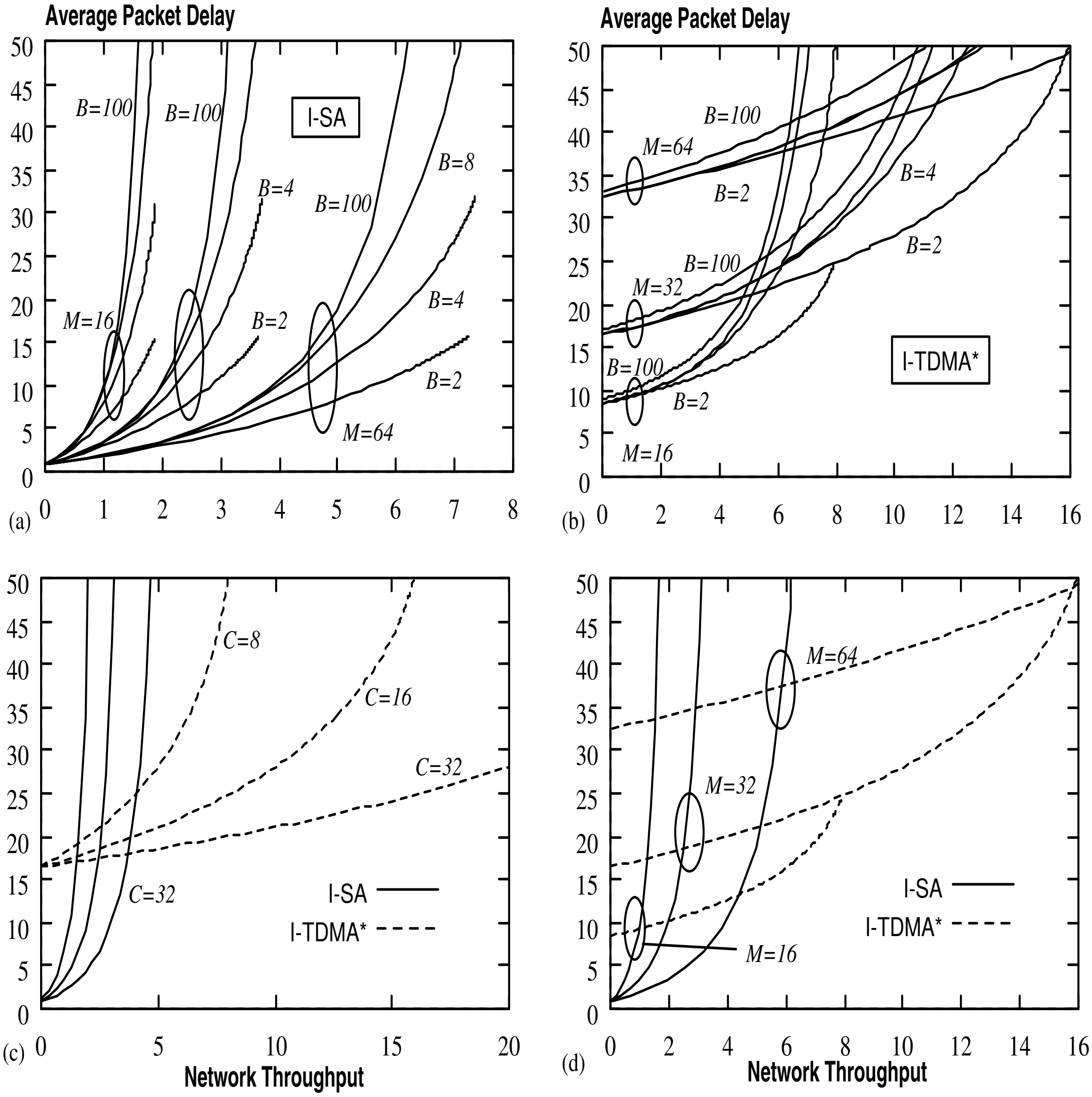

Figure 3.9: Impact of queue capacity: (a) I-SA for $M \in \{16, 32, 64\}$, $C = M/2$ and $B \in \{2, 4, 8, 100\}$, (b) I-TDMA* for $M \in \{16, 32, 64\}$, $C = M/2$ and $B \in \{2, 4, 8, 100\}$, (c) I-SA and I-TDMA* for $M = 32$, $C \in \{M/4, M/2, M\}$ and total queue capacity = $2C$, (d) Comparison of I-SA and I-TDMA* for $M \in \{16, 32, 64\}$, $C = M/2$ and total queue capacity = $2C$.

Variation in Data Channels

Fig. 3.8(a) compares the delay and throughput of the two protocols for a system of $M = 32$ nodes with varying channels $C \in \{1, 2, 4, 8, 16, 32\}$ and approximately infinite queue capacity ($B = 100$). The figure shows that both protocols are able to utilize an increase in channels to improve performance. The graphs compare the performance of I-SA and I-TDMA* to conventional Slotted Aloha and TDMA ($C = 1$). The maximum throughput using I-SA and I-TDMA* each increases by 100% when C is increased from 1 to 2.

As the number of data channels is increased from 2 to 32, the maximum network throughput of I-TDMA* increases from 2 to 8 – an increase of 1500%. Fig. 3.8(a) shows that I-SA is not able to take such an advantage of an increase in C. The maximum throughput increases by 633% when C is increased from 2 to 8. The improvement using I-SA is significant but lower in magnitude than that of I-TDMA*. The maximum theoretical throughput using Slotted Aloha is approximately $0.36C$. However, this is not observed in Fig. 3.8 because of the finite population model, a single transmitter and head-of-line effect due to a single queue. For a large M/C ratio, the maximum throughput of $0.36C$ is attained. For example, the maximum throughput is 1.44 for $C = 4$ and $M = 32$ which is close to the theoretical value. For $C = 8$ and $M = 32$, the maximum throughput attained using I-SA is 2.4 compared to the theoretical maximum of 2.9. In I-SA, the load per channel decreases as C increases and fewer nodes share a channel thereby reducing collisions and increasing the network throughput. In I-TDMA* the performance is dominated by the cycle time which is proportional to M. However, the head-of-line effect that restricted I-TDMA [84] from taking advantage of increase in C is eliminated in I-TDMA*. The main drawback is the minimum average delay of $M/2$ slots which limits this protocol in terms of scalability.

Variation in System Size

Fig. 3.8(b) compares the two protocols for $M \in \{8, 16, 32\}$ and $C = M$ with approximately infinite queue capacity ($B = 100$). The graphs show that packet delay in I-TDMA* is sensitive to M since cycle length is proportional to M. Packet delay is dependent on M with I-SA since additional nodes contribute more traffic to the system which increases total offered load and hence collisions. The impact of M on delay at lighter loads is absent in I-SA but prominent in I-TDMA*. I-SA maintains its delay characteristics so long as the ratio M/C is kept constant. I-TDMA* is able to take better advantage of the increased number of channels than does I-SA in terms of increased capacity. The maximum capacity increased consistently with increasing M for both protocols but I-TDMA* provided better performance at higher traffic. As expected, the random access protocol has superior performance for all configurations when the system is lightly loaded. As the traffic intensity increases, I-TDMA* is clearly the better choice due to stability and higher maximum throughput.

Impact of Queue Capacity

This section considers the impact of queue capacity on the performance of I-SA and I-TDMA*. I-SA has a single queue of variable capacity and I-TDMA* has C separate queues of variable capacity. The performance of the two protocols with equal total queue capacities is compared in this section.

I-SA: Fig. 3.9(a) shows the effect of queue capacity on the performance of I-SA for $M \in \{16, 32, 64\}$, $C = M/2$ and $B \in \{2, 4, 8, 100\}$. The plot for $B = 100$ is essentially identical to the plot for infinite queue capacity obtained through simulation so $B = 100$ is used to represent infinite queue capacity.

Finite queue capacity places an upper limit on the average packet delay of accepted packets. Network throughput decreases with reduced queue capacity because more packets are discarded. The reduction in network throughput is not observed under very light traffic and very heavy traffic. The effect of bounded delay with low queue capacity is seen in Fig. 3.9(a). The average delay for $C = 32$ and $M = 64$ is bounded at 15 slots for $B = 2$ and 66 slots for $B = 8$. The maximum bound on delay increases linearly with B because the queue is always full under heavy traffic. A packet has to wait for $B - 1$ packets ahead of it in the queue to complete transmission before it can be transmitted. Network throughput is reduced for lower values of B due to increased packet discarding. Under light traffic, very few packets need to be queued and the probability of discarding is low. Under heavy traffic, the node is limited by the processing capacity of the transmitter so increasing the queue capacity does not result in any difference in maximum throughput.

I-TDMA*: Fig. 3.9(b) shows the effect of queue capacity on the performance of I-TDMA* for $M \in \{16, 32, 64\}$, $C = M/2$ and $B \in \{2, 4, 8, 100\}$. The plot for $B = 100$ is essentially identical to the plot for infinite queue capacity obtained through simulation so $B = 100$ is used to represent infinite queue capacity.

Finite capacity queues result in bounded packet delay of accepted packets for I-TDMA* as in I-SA. Packets that arrive when the queue is full are discarded. The queue size is finite and has a maximum value of B. The amount of time spent transmitting each packet in the queue is fixed as one slot. This resulted in bounded packet delay for the accepted packets. This can be seen in Fig. 3.9(b) where delay for $C = 8$ and $M = 16$ is bounded at 25 slots for $B = 2$, and 55 slots for $B = 4$. Increasing B increases the maximum bound on delay because more packets are admitted which increases the average queue size. The performance using $B = 8$ is close to the performance using infinite capacity queues. Using higher values of B does not result in significant change in performance. For $B = 100$, packet arrival rates higher than λ_{max} where $\lambda_{max} = C/M$ results in unbounded delay and an unstable system. There is no such limit on input rate with finite queues because of the bounded queue size. The network throughput is identical at lower loads for all values of B since the queue capacity was not fully utilized. Similarly, at high loads, the system is limited by the number of channels in the system. The queues are always full under high loads and there is very little variation in network throughput. As before, the variation is at intermediate loads. The probability of packet discarding is higher for lower B which results in reduced throughput for lower B.

Equal Total Queue Capacity: This section compares I-SA and I-TDMA* with equal total queue capacity. The total queue capacity of I-TDMA* is CB where B is the capacity of a single channel queue. The single queue of I-SA has a capacity of CB. Fig. 3.9(c) compares the protocols for $M = 32$ and $C \in \{8, 16, 32\}$ with a total queue capacity of $2C$. Fig. 3.9(d) compares the delay and network throughput for $M \in \{16, 32, 64\}$ and $C = M/2$ with a total queue capacity of $2C$.

Fig. 3.9(c) shows that the delay using I-TDMA* is bounded at approximately 50 slots and the delay using I-SA is high under heavy traffic. The delay characteristics using a queue of capacity $2C$ for I-SA is close to that using a queue of unlimited capacity. It offers bounded delay and higher system capacity in the case of I-TDMA*. Fig. 3.9(d) shows similar results for varying system size – bounded delay for I-TDMA* and high delay under heavy traffic for I-SA. In both cases, the delay of I-SA is lower than I-TDMA* under light traffic as expected.

The impact on network throughput with varying channels and fixed system size is depicted in Fig. 3.9(c). The throughput attained by I-TDMA* is much higher than I-SA even though the per queue capacity is 2 for I-TDMA*. The maximum throughput of I-TDMA* is higher that I-SA by 233% for $C = 16$ and $M = 32$ and 370% for $C = 32$ and

$M = 32$. This shows the elimination of the head-of-line effects due to the single queue at a node. Fig. 3.9(d) presents identical results for different system sizes of $M \in \{16, 32, 64\}$ and $C = M/2$. The maximum throughput attained by I-TDMA* is higher than I-SA for all values of M. For example, maximum throughput of I-TDMA* is higher than I-SA by 320% for all values of M. This shows the throughput increase is maintained for all values of M as long as the M/C ratio is constant.

3.3.7 Comparison of I-TDMA and I-TDMA*

The performance of I-TDMA* is analyzed in this section. It was noted earlier that propagation delay results in a linear increase in delay which is inevitable and does not affect capacity. The following graphs have been plotted for $a = 0$.

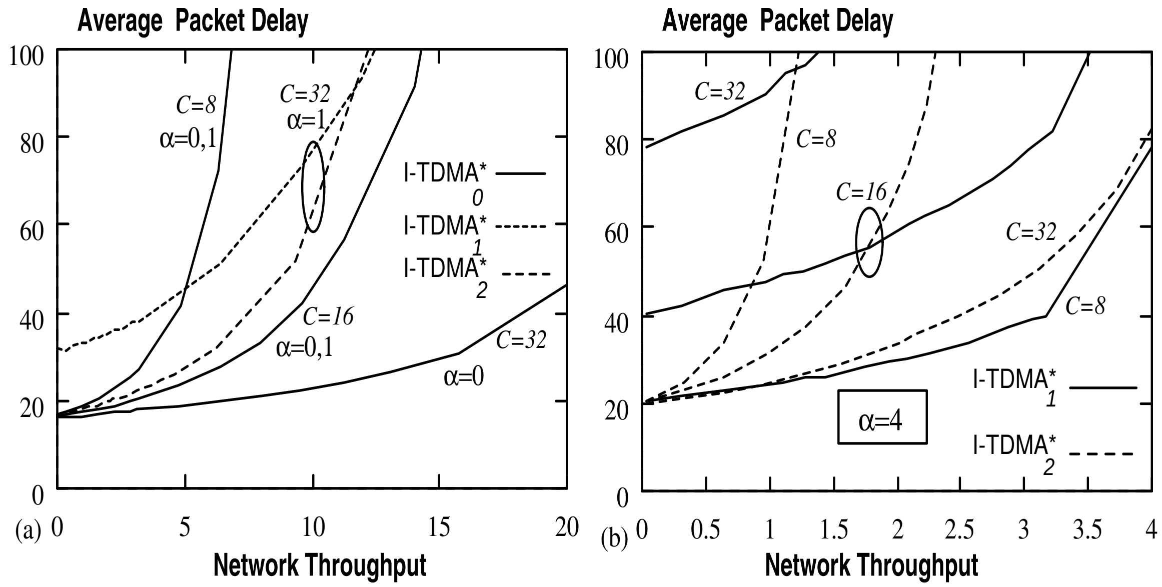

Figure 3.10: Performance of I-TDMA* for $M = 32$ and $C \in \{8, 16, 32\}$: (a) $\alpha \in \{0, 1\}$ (b) $\alpha = 4$.

Fig. 3.10 shows the performance impact of varying C for $M = 32$ and $\alpha \in \{0, 1, 2, 4\}$. In general, increasing the channels should improve performance. Fig. 3.10(a) shows the effect of increasing the number of channels for $\alpha \in \{0, 1\}$. For $\alpha = 0$, the relation $C\mathcal{T} \leq M$ holds for all values of C since $C \leq M$ and $\mathcal{L} = M$. $\mathcal{D}$ reduces with increasing C and in each case, $\mathcal{S}_{max} = C$. The decrease in $\mathcal{D}$ at $\mathcal{S} = 5$ is 50% when C is increased from 8 to 32 for $M = 32$ and $\alpha = 0$.

For $\alpha > 0$ the time spent switching between channels affects the cycle length and hence the performance. Increasing the number of channels also increases the total switching time in one cycle. The performance improves with increasing C until $C = C_{opt}$ where C_{opt} is the value of C that provides maximum performance and is given by $C_{opt} = \lfloor M/\mathcal{T} \rfloor$. $\mathcal{S}_{max} = C$ if $C \leq C_{opt}$. C_{opt} decreases with increasing α. If $C_{opt} < C < M$, then $\mathcal{S}_{max} = M/\mathcal{T}$ for I-TDMA$^*_1$ and $\mathcal{S}_{max} = \lceil C/\mathcal{T} \rceil$ for I-TDMA$^*_2$. For $C = M$, is $\mathcal{S}_{max} = M/\mathcal{T}$ for both schemes.

Fig. 3.10(a) shows the effect of increasing C for $M = 32$ and $\alpha = 1$. The performance improves with increasing C until $C = C_{opt}$ where $C_{opt} = 16$ for $M = 32$ and $\alpha = 1$. For $C > 16$, $\mathcal{D}$ of I-TDMA$^*_1$ increases with C due to the extended cycle and $\mathcal{S}_{max}$ is limited to 16. I-TDMA$^*_2$ results in lower delay under light traffic but $\mathcal{S}_{max}$ is limited to $\lceil C/2 \rceil$.

Fig. 3.10(b) shows the effect of increasing C for $\alpha \in \{2, 4\}$. $C_{opt} = 6$ for $\alpha = 4$. $\mathcal{D}$ is lower under light traffic for I-TDMA$^*_2$ but I-TDMA$^*_1$ has higher capacity. $\mathcal{S}_{max}$ with both schemes is 6 for $\alpha = 4$.

3.3.8 Comparison of I-SA* and I-TDMA*

Fig. 3.11 compares the two protocols for $M \in \{16, 32, 64\}$, $C = M/2$, $\mathsf{I} = M/2$, $\mathsf{A} = M/2$ and varying α. The time interval between packet reception and forced acknowledgment transmission is denoted by A. A forced acknowledgment is required under light traffic if an acknowledgment had not been piggybacked already. I is the source node timeout parameter for retransmission of an unacknowledged packet. Only I-TDMA$^*_1$ is plotted when

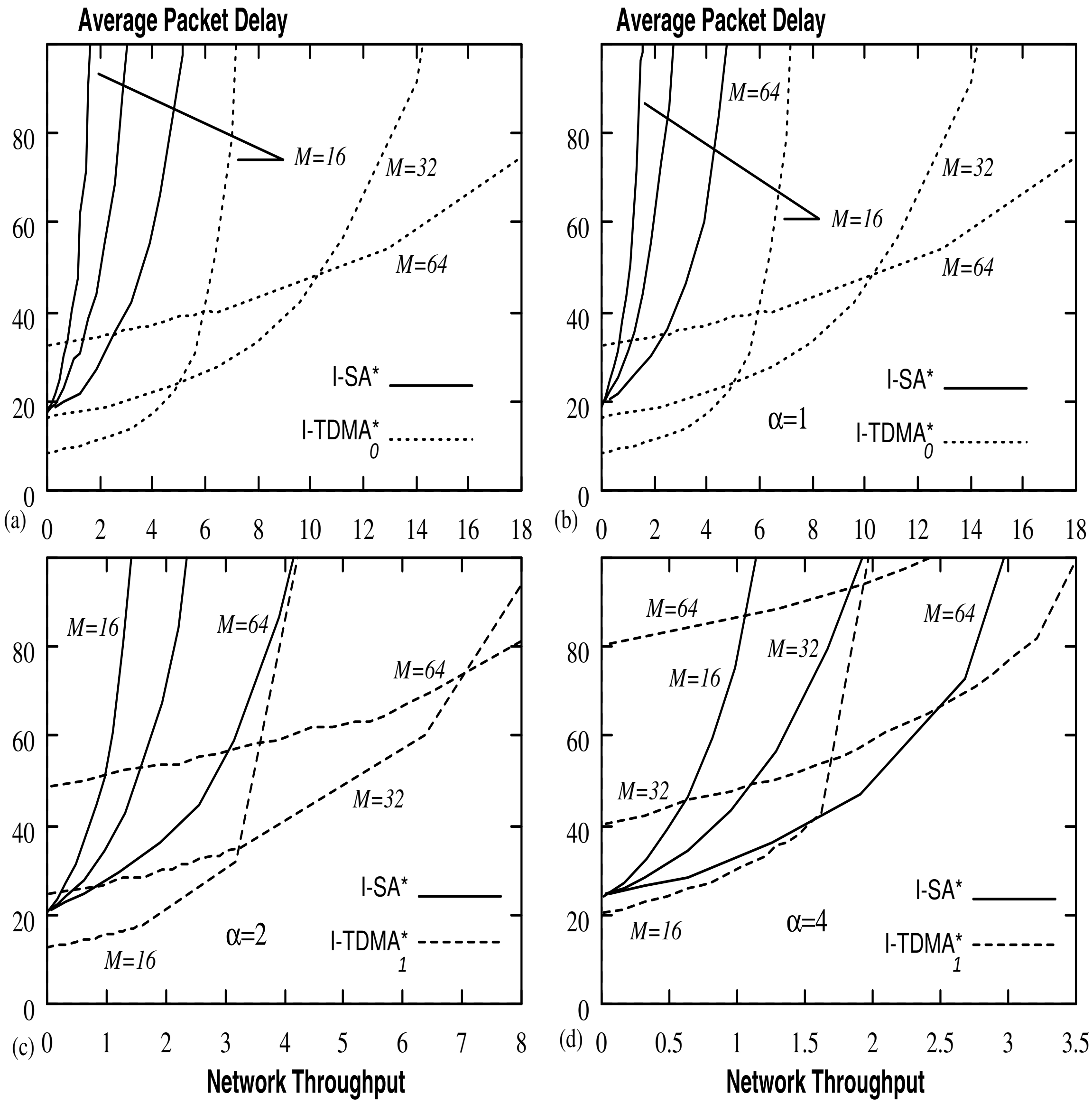

Figure 3.11: Comparison of I-SA* and I-TDMA* for $M \in \{16, 32, 64\}$, $C = M/2$, $\mathsf{I} = M/2$, $\mathsf{A} = M/2$: (a) $\alpha = 0$, (b) $\alpha = 1$, (c) $\alpha = 2$, (d) $\alpha = 4$.

$C\mathcal{T} > M$ since I-TDMA$_1^*$ was shown in the previous section to have better performance than I-TDMA$_2^*$ for $C < M$.

I-SA* can offer lower delay under light input traffic but the values herein chosen for I and A result in higher delay. Delay is not dependent on C or M under light traffic but is determined by I and A. I-TDMA* is sensitive to C, M and α since cycle length $\mathcal{L}$ depends on the allocation scheme and average packet delay is proportional to cycle length. As the load increases, I-SA* collapses because of increased collisions and is not able to withstand heavy traffic. On the other hand, I-TDMA* does not collapse under heavy traffic and provides higher throughput.

The maximum theoretical throughput per channel for I-SA* is not attained because of finite population and explicit acknowledgment requirements. I-TDMA* is more robust at higher loads and its saturation point is higher than that of I-SA* for all values of α irrespective of the allocation scheme used. If $C\mathcal{T} \leq M$, I-TDMA* offers the maximum theoretical throughput of C achieved with C multi-access channels. Both protocols are able to take advantage of an increase in the number of channels. However, the impact of increased switching overhead for large C is significant in both cases. The impact is higher in I-TDMA$_1^*$ where the cycle length is equal to $C\mathcal{T}$.

For $\alpha = 0$, $\mathcal{S}_{max}$ offered by I-TDMA* is higher than I-SA* by 330% for $M = 16$ and 430% for $M = 64$ where $C = M/2$ in both cases. The increase in maximum throughput diminishes with increasing α. For $\alpha = 2$, I-TDMA$_0^*$ cannot be used since $C\mathcal{T} > M$. I-TDMA$_1^*$ offers a maximum throughput of $M/\mathcal{T}$. The increase in $\mathcal{S}_{max}$ from I-SA* is 270% with I-TDMA$_1^*$ for $\alpha = 2$, $M \in \{16, 32, 64\}$ and $C = M/2$.

The principal advantage of I-TDMA* over I-SA* is its insensitivity to propagation delay. I-SA* is dependent on acknowledgments and its performance reduces significantly as a increases. I-TDMA* offers better throughput than I-SA* but its main drawback is the dependence of cycle length on M (C and α in I-TDMA$_1^*$) which results in increased delay. I-SA* has the capability to offer lower delay under light traffic. As traffic intensity increases, I-TDMA* is clearly the better choice due to stability and higher network throughput.

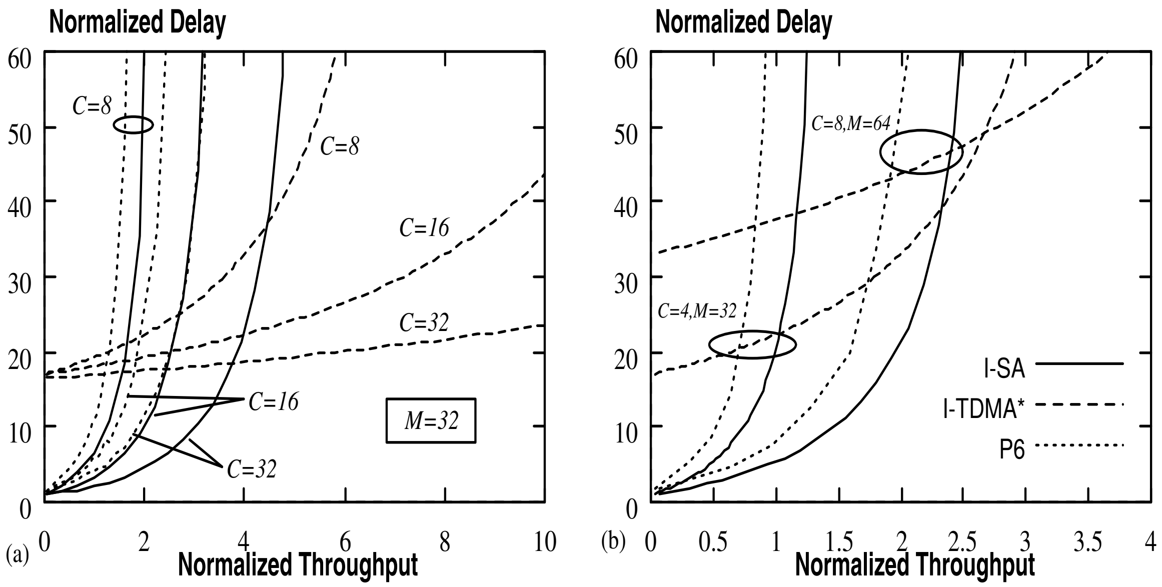

Figure 3.12: Comparison of I-SA and I-TDMA* to P6 – a reservation based protocol: (a) $M = 32, C \in \{8, 16, 32\}$ and $L = 32$, (b) $M \in \{32, 64\}, C = M/8$ and $L = 32$.

3.3.9 Comparison to reservation based protocols

This section compares protocol performance to reservation based protocols. About ten reservation protocols based on Slotted Aloha and Reservation Aloha were proposed in [11]. The protocols operate with one control channel and $C - 1$ data channels. The architecture is based on one tunable transmitter and one tunable receiver per node. The access mechanism is as follows. A node i transmits a control packet on the control channel if a data packet is generated at the node. An idle node listens to the control channel and tunes to the data channel indicated in the control packet if the control packet is successful and the packet is destined for it. At the end of control packet transmission, the data packet is transmitted. If a node attempts to transmit to a node currently receiving on another channel, the transmission is unsuccessful. This is termed receiver collision [76]. The analysis in [11] assumes infinite population and does not consider receiver collisions but are considered in the following analysis. Immediate feedback about the success of the transmitted packet is assumed and the packet is retransmitted till it is successful. Time is slotted on control packet boundaries and data packet length is assumed to be L time slots.

In this paper, we consider protocol 6 (P6) since it offers the best throughput-delay characteristics compared to other Slotted Aloha based protocols presented in [11]. Each channel with P6 has its own cycle of duration $L + 1$ slots. If a node has a data packet to transmit, it randomly chooses a data channel from $C - 1$ channels, waits for the beginning of the cycle on that channel, transmits the control packet followed by data packet irrespective of control packet success. Retransmissions are attempted until successful transmission of the data packet.

The advantages of using the architecture and protocols presented in this paper compared to that of P6 are summarized as follows:

Simpler low-cost architecture: Reduced architectural complexity and system cost – only one transmitter-receiver pair per node is required and fast-tunable receivers are not required.

Higher channel utilization: All channels are utilized for data transmission – the impact is higher for lower number of channels.

No receiver conflicts: No arbitration algorithm is required to resolve receiver conflicts since receiver conflicts are eliminated with the pre-allocation approach.

Extensible system: Easily reconfigurable system with change in number of nodes and channels.

Packet delay for P6 was defined as the number of control slots required to transmit a packet of length L slots. Since time is slotted on packet boundaries for I-SA and I-TDMA*, packet delay for P6 is normalized to the data packet length in the comparison to follow. Similarly, network throughput in P6 was defined as the number of packets transmitted per channel cycle so the throughput of P6 is also been normalized to packet transmission time. The delay is now defined as the number of data packet slots needed to transmit a data packet and system throughput is the number of packets successfully transmitted across all channels in one data packet slot.

The architecture proposed in this paper is more cost-effective because tunable receivers are not required. Also, the delay characteristics of I-SA are better than P6 because no control packet is transmitted. Each (re)transmission of a packet in P6 requires $L + 1$ time units whereas only L slots are required in I-SA. Also, the number of available data channels for I-SA is C instead of $C - 1$ as in P6. P6 suffers from an initial cycle synchronization delay per packet equal to $\frac{L+1}{2}$ slots and is dependent on packet size. The maximum theoretical throughput for infinite population achieved by I-SA is roughly $0.36C$ compared to $0.36(C - 1)$ attained by P6.

P6 is simulated using discrete event simulation. Convergence is obtained on delay with a confidence of 95% in a less than 5% variation from the mean. The retransmission probability per cycle for P6 and per data slot for I-SA is taken as 0.05. The protocols are implemented with finite population and also consider receiver conflicts. Fig. 3.12 compares the performance of the three protocols in terms of normalized throughput and delay. Fig. 3.12(a) compares the effect of increasing C and fixed M for $M = 32, C \in \{8, 16, 32\}$ and $L = 32$. The value of L is empirically chosen since it yielded the maximum performance for P6. Fig. 3.12(b) compares the protocols for $C = 4$ and $M = 32$ and $C = 8$ and $M = 64$.

The performance of I-SA is better than P6 in terms of lower packet delay and higher maximum throughput. Under light traffic, the average delay using I-SA is L units, which is the length of the data packet. The average delay using P6 is $\frac{L+1}{2} + L + 1$ which includes cycle synchronization and delay for both data and control packets. P6 suffers from destination conflicts which increases delay and decreases throughput. For example, the maximum throughput for $C = 8$ and $M = 64$ is equal to 2.4 for I-SA and 2.0 for P6 compared to the theoretical bound of 2.9 for I-SA and 2.5 for P6. Similarly, the maximum throughput for $C = 32$ and $M = 32$ is 5.5 for I-SA and 3.9 for P6. In short, I-SA is able to outperform P6 despite the lack of tunable receivers and proves to be a more cost-effective solution.

I-TDMA* is advantageous over P6 because of its higher maximum throughput, higher system capacity and stability under heavy traffic. P6 suffers from instability under heavy traffic like any Aloha based system. The packet delay is higher under light traffic using I-TDMA* because of its time multiplexed nature. The maximum theoretical throughput offered by I-TDMA* is C. This is verified in Fig. 3.12 where the delay using I-TDMA* is higher under light traffic but maximum throughput and system capacity is higher for I-TDMA*.

3.4 Summary

This chapter presented the pre-allocation protocols studied for a star coupled WDM network. Static and random access schemes were examined and extensively analyzed using analytic and simulation models. The results are summarized here. I-SA* was observed to have lower packet delay at lighter loads compared to I-TDMA*. However, the performance of I-TDMA* under heavy traffic is superior to that of I-SA* since its saturation point is significantly higher. In particular, I-TDMA* is capable of achieving a maximum throughput equal to the number of channels in the system. Both I-SA* and I-TDMA* take advantage of an increase in the number of channels in the system. I-TDMA* is sensitive to increases in system size and insensitive to propagation delay. I-SA* is less sensitive to system size increase and highly sensitive to propagation delay due to acknowledgments.

The choice between I-SA* and I-TDMA* lies in the delay/throughput requirements and traffic characteristics. I-SA* is suitable for a system with large number of bursty users where the aggregate traffic is low and propagation delay is small. I-TDMA* is more suitable for an environment where number of nodes is small, traffic is steady and high, and propagation delay is large.

This chapter focused on the star coupled network without amplification mainly because of its power budget and fault tolerance advantages. Another topology widely studied for local networks is the bus topology. Poor power budget characteristics of bus based networks led to star being the primary topology choice for optic networks. However, the advent of optical fiber amplifiers in particular erbium-doped fiber amplifiers (EDFA) [2] has renewed interest in bus topology. The next chapter examines different protocol choices for the bus and star WDM networks with photonic amplification.

Chapter 4

Star and Bus Topologies

The design of local area networks (LAN) and metropolitan area networks (MAN) is significantly impacted by the network topology. The network topology influences system characteristics such as system size, cost, fault tolerance and performance. The most commonly studied local network topologies have been the star and the bus. This chapter examines in detail the performance of the bus and star topologies with amplification with respect to fiber optic networks. The relative advantages of each topology in terms of system scalability, cost, fault tolerance and protocol design simplicity are examined.

Chapter 2 discussed some of the configurations for an optical multiple access network. Fig. 2.2 shows the topologies commonly studied for local area networks. In particular, this chapter considers the *star*, *dual bus* and *singly folded bus* topologies. The bus topology has been favored for local network interconnection since degradation due to propagation delay can be reduced by taking advantage of the physical ordering [53]. However, power budget considerations showed that the system size of the bus was severely limited while the star had excellent fanout and fault tolerance characteristics [5]. This was followed by extensive research towards development of star based networks.

The improvements of optical amplifiers [2,15] have the potential of extending the system size of bus based networks, motivating their reconsideration. The bus with amplification has the capacity to support hundreds of nodes [2, 16]. However, the star was still shown to retain its excellent fanout relative to the bus with optical amplifiers. In addition to providing implicit node ordering, the bus is characterized by lower fiber but higher coupler requirements [17], and requires amplifiers even for a reasonable network size. The star, on the other hand, is characterized by large fanout and high fault tolerance, but requires more fiber. The objective of this chapter is to evaluate the performance of bus and star topologies for fiber optic networks with optical amplifiers and examine whether the performance benefits of each topology justify their relative cost. The details of this analysis is available in [99].

Section 4.1 describes the topologies and their characteristics. Section 4.2 describes the access arbitration techniques used in the evaluation of the topologies. Section 4.3 compares the two topologies with respect to system design, protocol performance and protocol complexity. Discrete-event simulation has been used to evaluate performance to ensure a consistent evaluation and avoid the simplification assumptions often required by analytic models.

4.1 Topologies

Fig. 2.2(a) shows the singly folded unidirectional bus configuration (FUB). The FUB has two unidirectional sections – the *outbound* bus for data transmission and the *inbound* bus for data reception. This configuration has been used in protocols including D-Net [54]. Fig. 2.2(c) shows the doubly-folded unidirectional bus (DFUB) used in Expressnet [55]. Both FUB and DFUB configurations require a sense tap and a transmitter on the outbound bus and a receiver on the inbound bus. A dual-bus configuration (DUB) is shown in Fig. 2.2(b) with two independent buses where nodes communicate on the buses as specified by the access mechanism. This approach has been used by Fasnet [56] and DQDB (*IEEE 802.6)*. Each node has a sense tap, transmitter and a receiver on each bus doubling system cost. In a single channel system, the sense tap and receiver functions can be implemented using a single device [56]. In multi-channel systems where devices can be fixed or tunable, the sense tap and receiver may be implemented separately. The physical layout of the star is shown in Fig. 2.2(d). Access mechanisms studied for the multi-channel star network are summarized in [76].

The optical power budget (OPB) characteristics determine the number of nodes that can be connected in a network while maintaining a specified bit-error-rate. The number of nodes attached to a multiple access channel is bound by the saturation traffic and the optical power budget. The high bandwidth of fiber precludes the saturation limitation and the principal factor of optical fanout is the power budget. Power budget considerations for fiber optic networks [5] showed that the system size supported by the bus was very small whereas the star could support hundreds of nodes. The system sizes of bus and star networks with erbium doped optical amplifiers (EDFA) have been studied in [2, 16] and the star was also shown to have better fanout for multi-channel systems. A FUB network with an optimized number of amplifiers (Gain of 15 dB) and active reciprocal couplers was shown to have a fanout of over 400 nodes, whereas a star network with receiver pre-amplification was shown to support over 2000 nodes [2].

The following paragraphs discuss design considerations for both topologies, in terms of cost, couplers, propagation delay, and fault tolerance.

▹ System Cost: includes fiber, coupler, transceiver, and amplifier costs. The media

Topology	Fiber Cost	Couplers	Avg. Prop. Delay	Fault Tolerance
FUB	$2(M-1)$	$2M$	$(2M-1)/3$	$(3M, N_f, 1)$
DUB	$2(M-1)$	$2M$	$(M+1)/3$	$(4M, N_d, 1)$
Star	$0.765M^{\frac{3}{2}}$	1	$0.765\sqrt{M}$	$(2M, M, 1)$

Table 4.1: Topology comparison, where M is the number of nodes.

access protocol determines the transceiver requirements and is considered in Section 4.3.3. Table 4.1 shows that the star requires more fiber than the bus. The bus can also be realized using a centralized collector to which the nodes are connected in which case the fiber requirements for the bus are the same as the star.

- ▷ Couplers: Table 4.1 compares the coupler requirements for the topologies. The couplers used for the bus can be classified as passive, active and regenerative couplers. Passive directional couplers transfer a fixed percentage of the power, while active couplers modify their reciprocity when transmitting. Regenerative node couplers remove most of the power off the bus and re-inject power by retransmitting the received signal resulting in an active optical network [100]. This approach has better fault tolerance since the network is revived despite single coupler failures. The number of amplifiers required and the placement of amplifiers on the bus have been considered in [2]. The star coupler is usually passive when no amplification is provided. Receiver pre-amplification and built-in star coupler amplification have been compared in [2] where the former was shown to provide better power budget.

- ▷ Propagation Delay: between nodes is a significant concern, particularly in high speed networks. Table 4.1 compares the average propagation delay for the three topologies. The minimum delay between nodes is lowest for the bus and is equal to 1. Propagation delay aspects were compared in [17] and the star was shown to have lower average propagation delay.

- Fault tolerance: Fault tolerance of a topology is characterized by its ability to sustain network component and link failures. The ease with which a network fault can be overcome leading to normal network operation is another significant concern.

  A single link break at any section of FUB divides the network into two, bringing the network to a potential collapse. The DUB is actually not more fault tolerant than the FUB configuration since a node or link failure may affect only one bus, but both buses are needed to return an acknowledgment. Star-coupled networks have better fault tolerance since a single link failure localizes the impact to the node connected through the affected link.

  The network is constructed using passive optical links and passive couplers which are not often subject to damage. The crucial network components that can be used to compare fault tolerance are the active components, plus the places where connections are made (that require human intervention). In a network with passive couplers, the only active components are the amplifiers. The three topologies can be characterized by the 3-tuple (L, D, F) where L denotes the number of connectors, D the number of active components, and F a boolean indicating network failure due to one active component failure. Table 4.1 provides the fault tolerance comparison. N_f and N_d represent the optical number of amplifiers required for FUB and DUB respectively, where $0 \leq N_d \leq N_f < M$. The star is characterized by $(2M, M, 1)$ if an active star coupler [101] is used, or by $(2M, M, 0)$ if receiver pre-amplification is used.

4.2 Access Arbitration Mechanisms

The following section examines arbitration mechanisms for the topologies and describes the protocols used in the evaluation of the topologies. Demand Assignment Multiple Access (DAMA) schemes for the bus have been studied in [53]. Other approaches have since been proposed for bus-based networks. A TDM-based protocol referred to as AMTRAC has been studied in [102]. Fairnet is based on per node transmission probabilities and has been studied for the multi-channel domain [103]. The distributed queue dual bus (DQDB) is being considered as a metropolitan area network standard *(IEEE 802.6)*. A multi-channel extension of the DQDB protocol referred to Estimated Queue Extended Bus (EQEB) has been studied in [104].

In the following sections, we present the protocols used to evaluate the topologies. All use receiver pre-allocation: each node receives traffic on exactly one channel, referred to as its *home channel* [105]. A source node tunes its transmitter to the home channel of the destination node and transmits according to the access protocol. Each node has C queues to buffer packets based on destination channel. Network traffic is partitioned so that each node does not have to receive and process all transmitted packets, but only its home channel traffic. Let M and C denote the number of stations and channels, respectively. Node m_j is assigned c_k where $k = j$ mod C as its home channel based on an interleaved allocation policy. This allows each node to determine the home channel of a node in a decentralized fashion.

4.2.1 Folded Bus

Three protocols have been studied for the FUB configuration – a multi-channel extension of D-Net, Fairnet and an extended version of AMTRAC.

D-Net/C: D-Net [54] has been generalized to operate in a multi-channel WDM environment and is denoted as D-Net/C. A generator is located at the left end of the bus which launches a locomotive on a outbound bus channel when it senses an end-of-train (EOT) on the corresponding inbound bus channel. Packet transmission is simple: a backlogged node waits for the locomotive on the appropriate outbound bus channel and transmits immediately. The node simultaneously senses for upstream transmission and aborts upon detecting carrier. If collision is detected, nodes wait for an end-of-carrier (EOC) on the same outbound bus channel and re-attempt transmission. A node transmits once per channel round providing fair access to all nodes.

Fairnet: In this protocol, the head node generates fixed size data slots on all channels. Each node has a set of transmission probabilities p_{ic} where $0 \leq i \leq M-1$ and $0 \leq c \leq C-1$. p_{ic} represents the probability that node i transmits on channel c in a given slot. A backlogged node transmits if the BUSY bit of a slot on the destination channel is not set simultaneously setting the BUSY bit. The values of p_{ic} are determined from network traffic distribution and equity constraints [103].

B-TDMA: AMTRAC [102] is a protocol based on time multiplexed multi-channel access. Time is divided into equal slots of length Δ where Δ is the propagation delay between adjacent nodes. Each channel cycle is divided into $2M-2$ slots allocating each node one slot per cycle. A backlogged node begins transmission on a channel in its preassigned slot. B-TDMA is an extension of AMTRAC: (i) the slot is equal to a data packet, and (ii) the cycle length is determined based on the relation between M and Δ. Cycle length is M if M is not an integral multiple of $1+\Delta$, and is $M+1$ otherwise. The node allocation map of B-TDMA for a single channel for $M=8$ nodes and $\Delta=1$ is shown in Fig. 4.1.

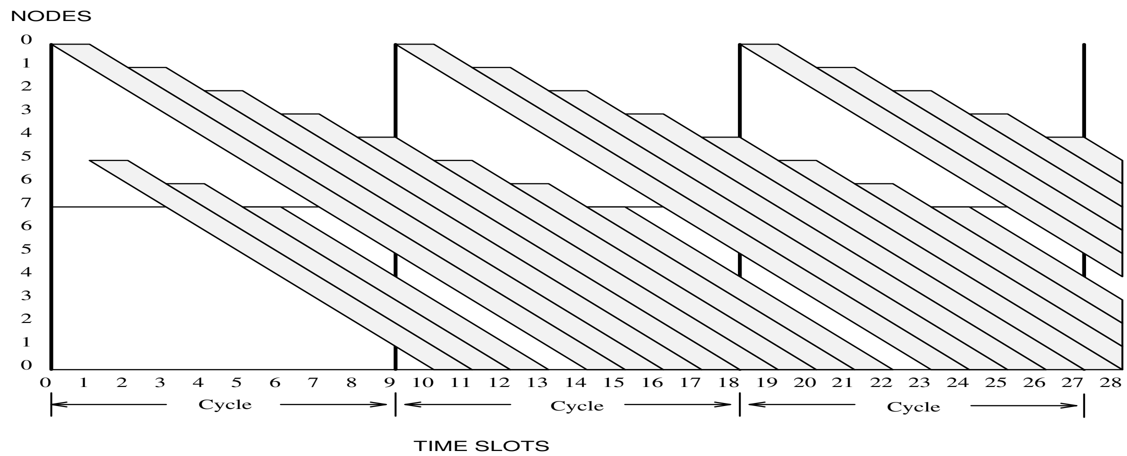

Figure 4.1: Allocation map for $M = 8$ nodes on one channel for B-TDMA. Internode propagation delay is assumed to be $\Delta = 1$ and cycle length is determined to be 9 slots.

4.2.2 Dual Bus

Fasnet [56] is a single channel media access scheme which uses a pair of buses transmitting data in opposite directions. DQDB, also based on the dual bus, has been proposed as the *IEEE 802.6* media access standard for MANs. In this chapter, we consider a multi-channel extension of DQDB based on EQEB presented in [104]. DQDB is based on the QPSX protocol [106] and uses a pair of counters per node per bus to implement the distributed queue. EQEB eliminates the per node counter and uses a global counter maintained by the head nodes to implement the queuing. The node uses estimated delays before placing the request and data on the buses [104]. This chapter extends the single-buffer per node system proposed by EQEB and requires C queues per node. This eliminates some head-of-line effects and helps in partitioning data traffic as seen later.

4.2.3 Star

Slotted Aloha provides access arbitration to each data channel with I-SA*. The transmitter chooses a non-empty queue and transmits a window of packets on the corresponding channel using Slotted Aloha. Acknowledgments are needed since collisions may occur on a channel. The source node transmits on other channels rather than wait for acknowledgment of the transmitted packets. As acknowledgments are received, other packets in the queue are transmitted. The receiver and transmitter behavior are described in [87]. In general, this is not a practical protocol for systems with large propagation delays. Alternative acknowledgment schemes have been examined in [93] that reduce but do not eliminate the performance degradation due to propagation delay.

Packet collisions occur with I-SA* requiring packet retransmission which may result in unbounded delay and system instability. I-TDMA* avoids the complexity of acknowledgments by time multiplexing access to data channels. Since I-TDMA* does not require acknowledgments, it is insensitive to propagation delay. Each node is allowed access to each channel once per channel cycle. A cycle is defined as the minimum time to allocate permission for all nodes to transmit on a channel. The minimum cycle length is M, the number of nodes. The node-to-channel allocation varies with transmitter switching latency, protocol processing latency, C and M [87]. The simplest allocation scheme is as follows: Node m_i is initially allocated channel c_0 at time $((i+1) \bmod M)$. Subsequent channels are allocated one unit apart until all channels are allocated for this cycle. The node remains idle for $M - C$ slots after which channel 0 is allocated again.

4.3 Performance Analysis

The following sections compare the topologies through protocol performance with varying system parameters such as number of nodes and channels, propagation delay and processing latency. The protocol design complexity and interaction between protocol and topology is then considered.

The performance metrics of interest are average packet delay and network throughput. The impact of varying packet generation rate (λ), unbalanced traffic, number of nodes (M), number of channels (C), processing latency (α) and propagation delay (P) is analyzed. The following paragraphs provide the basic definitions and the simulation model:

Processing Latency: Let T denote packet transmission time and κ denote the sum of protocol processing time and maximum transmitter tuning time between channels. Processing latency, denoted by α, is defined as

$$\alpha \triangleq \left\lceil \frac{\kappa}{T} \right\rceil$$

Propagation Delay: Let τ_b denote the propagation delay from one end of the bus to the other. The normalized propagation delay for the bus is defined to be:

$$P \triangleq \left\lceil \frac{\tau_b}{T} \right\rceil$$

For the star, we assume that the network may not be symmetric so all nodes may not be equidistant from the star. Let τ_i denote the propagation delay from node m_i to the star. Define τ as the maximum node-to-star propagation delay given by $\tau = \max\{\tau_i \mid 0 \leq i \leq M-1\}$, and normalized maximum propagation delay is denoted by

$$P \triangleq \left\lceil \frac{2\tau}{T} \right\rceil$$

Packet Delay: denoted by $\mathcal{D}$ is defined as the time elapsed between packet generation and reception of packet. Acknowledgments are required in I-SA* which makes it particularly sensitive to propagation delay, and $\mathcal{D}$ is defined as the time between the generation of the packet and the time its acknowledgment is received at the source node.

Network Throughput: denoted by $\mathcal{S}$ is defined as the number of packets that are successfully transmitted per slot across all channels. Since there are C channels in the system, the maximum network throughput is C. In EQEB, there are two independent buses and maximum throughput is $2C$. Maximum network throughput achieved by the protocol under extreme load is denoted by $\mathcal{S}_{max}$. Time, in general, is normalized to the packet transmission time.

Simulation Model: The simulators are based on stochastic self-driven discrete event models, written in the C programming language with a C based library of routines that provides discrete-event and random variate facilities. Steady state transaction times and throughput were measured. Simulation convergence was obtained through the replication/deletion method [97], with a 99% confidence in a less than 5% variation from the mean.

The simulation model makes use of the following assumptions:

1. All nodes are independent of each other.
2. Packet generation at each node follows a Poisson process with a rate of λ packets per unit time per node.
3. The traffic distribution can be characterized by two different models:

   Uniform Reference: A packet generated at node m_i is targeted to node m_j with probability $\mathcal{P}_{ij}$ where $\mathcal{P}_{ij}$ is defined as:

$$\mathcal{P}_{ij} = \begin{cases} \frac{1}{M-1} & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases}$$

Client/Server Model: One of the nodes designated as m_0 is the server and the other nodes are the clients. The server is equally likely to transmit to any of the $M-1$ clients. The traffic distribution for the client nodes ($1 \leq i \leq M-1$) is given by:

$$\mathcal{P}_{ij} = \begin{cases} 0 & \text{if } i = j \\ \frac{1}{M-1} & \text{if } i = 0 \text{ and } j \neq 0 \\ \beta & \text{if } i \neq 0 \text{ and } j = 0 \\ \frac{1-\beta}{M-2} & \text{if } i \neq 0, j \neq 0 \text{ and } i \neq j \end{cases}$$

4. C packet queues of variable capacity – one per channel. A queue capacity of 5 packets per channel has been used in the simulation analysis.

5. All packets are of fixed length, and time is normalized to the packet transmission time.

4.3.1 Performance Under Uniform Reference Model

The following sections consider the Uniform Reference Model where each node is equally likely to transmit a packet to all other node. Section 4.3.2 considers the client/server model.

D-Net/C: Fig. 4.2(a) shows the performance of D-Net/C for $M = 32$, $P \in \{1, 8, 16\}$ and varying C. The maximum throughput of D-Net/C for a C channel system is approximately:

$$S_{max}(\text{D-Net/C}) \approx \frac{1}{\frac{1}{C} + \frac{2P}{M}}$$

This assumes that M/C nodes are backlogged on a channel during a cycle. The delay from packet arrival at head of queue until transmission is bounded and is given by $M + 2P$ units, if each node transmits exactly one packet in a round. One advantage of this protocol is

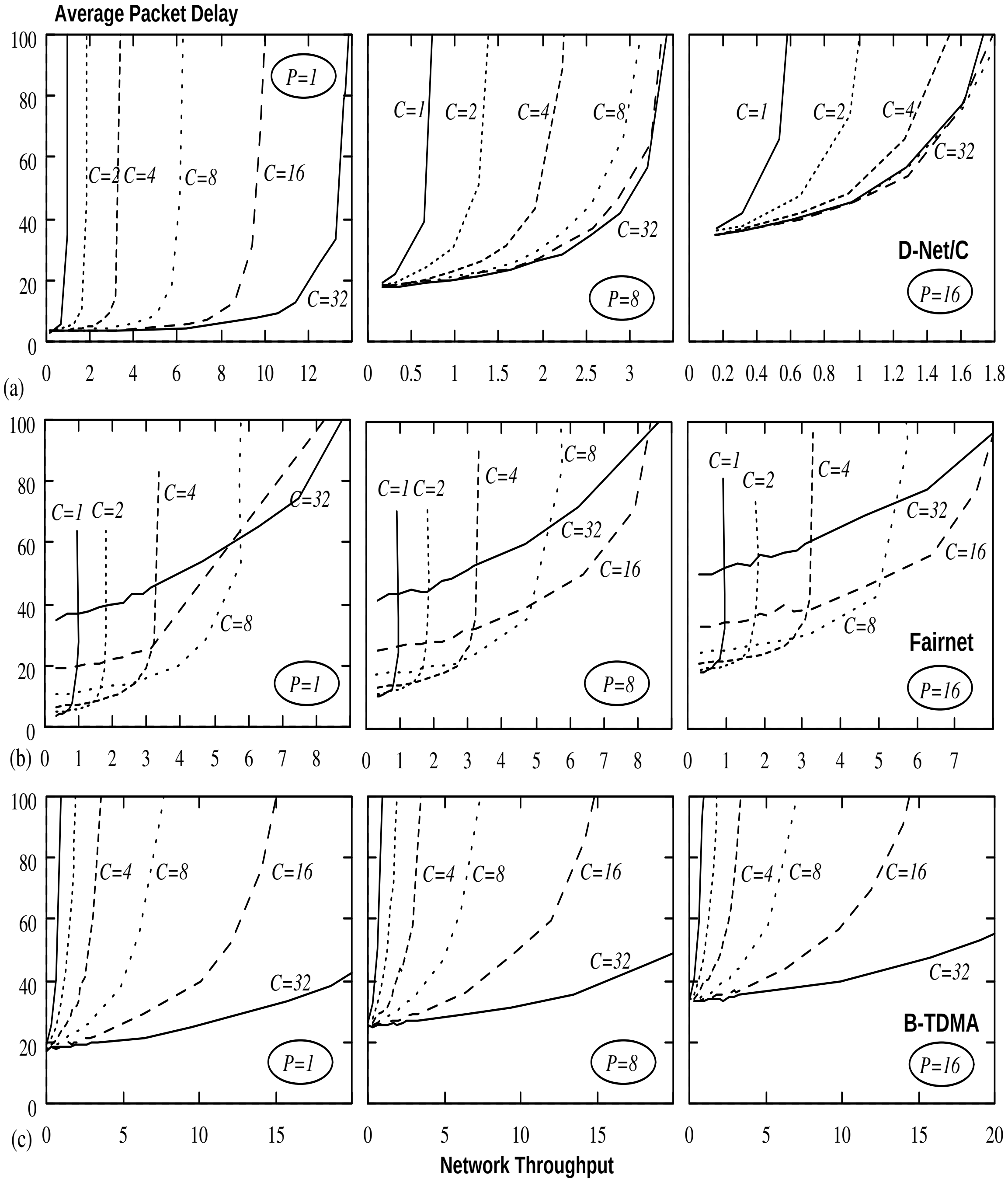

Figure 4.2: Performance of D-Net/C, Fairnet and B-TDMA for $M = 32$, and varying C and propagation delay: (a) D-Net/C, (b) Fairnet, and (c) B-TDMA.

that it is asynchronous and packets need not be of fixed length since EOC is used to trigger transmission.

For $P = 1$, the propagation delay between adjacent nodes is roughly 0.03. The delay is seen to be roughly the same for all loads until saturation is attained. For $C = 1$, $\mathcal{S}_{max}$ approaches 1 corresponding to maximum channel utilization. Maximum throughput increases with increasing C but does not equal C. This is due to the following reason: a node has to wait for end-of-carrier (EOC) from the upstream node before it can transmit. After tuning to a channel, there is an average delay of $M/2$ before a node can transmit its packet. This delay affects maximum network throughput and the channels are not fully utilized at all times. Also, P is still the predominant factor in determining maximum throughput of the system and increasing C does not completely eliminate propagation delay impact. $\mathcal{S}_{max}$ at $P = 1$ is approximately 10 for $C = 16$ and $M = 32$, and 15 for $C = 32$ and $M = 32$. Maximum throughput for $C = 16$ and $M = 32$ decreases by 75% when P is increased from 1 to 8.

Fairnet: Fig. 4.2(b) shows the performance of Fairnet for $M = 32$, $P \in \{1, 8, 16\}$ and varying C. The transmission probabilities are based on equal average packet delay criterion as derived in [103]. For $C = 1$, $P = 1$ and $M = 32$, maximum system throughput is almost equal to 1. $\mathcal{S}_{max}$ increases with C but does not approach C because of the increased switching overhead. The maximum throughput with 32 nodes and 32 channels is approximately 12. Packet delay under light loads increases with increasing C. This is explained as follows: under uniform traffic and light load, $p_{ic} \approx \frac{1}{C}$ for all nodes and all channels. A packet waits on the average C slots after generation since its transmission channel is selected with geometric probability $\frac{1}{C}$. One advantage of this protocol is the lack of cycle which significantly diminishes the impact of propagation delay. This is due to the continuous stream of slots from the head node and lack of a cycle. The decrease in

$\mathcal{S}_{max}$ is less than 1% as P is increased from 1 to 16 for $C = 32$ and $M = 32$. Another advantage of Fairnet is its ability to tune the transmission probabilities based on a priori traffic distribution.

B-TDMA: Fig. 4.2(c) shows the performance of B-TDMA for $M = 32$, $P \in \{1, 8, 16\}$ and varying C. The impact of propagation delay is de-emphasized in this approach by eliminating the dependency of start of cycle on propagation delay. The head node starts a cycle on each channel every M slots. The time division multiplexed nature of B-TDMA results in an average delay of $M/2$ after tuning to a channel, similar to the start of round wait with D-Net/C. Nodes wait for EOC in D-Net/C, whereas B-TDMA uses system wide synchronization to establish transmission start times. Increasing number of channels results in increased throughput and reduced delay as seen in Fig. 4.2(c). It is possible to attain maximum theoretical throughput of C with this approach. The delay from packet arrival to head of queue until transmission is bounded by a maximum of M units. Increasing propagation delay results in increased packet delay but does not affect maximum throughput. The maximum throughput, $\mathcal{S}_{max}$, is equal to C irrespective of propagation delay. This high throughput is achieved at the expense of higher delay under light traffic.

EQEB: Fig. 4.3(a) shows the performance of EQEB for $M = 32$, $P \in \{1, 8, 16\}$ and varying C. The maximum throughput for a single channel DQDB is 2. Maximum throughput achieved by EQEB increases with C but does not attain a maximum of $2C$. One possible reason is the finite population of $M = 32$ due to which the number of nodes backlogged on a channel given by is small. The increase in $\mathcal{S}_{max}$ is 64% when C is increased from 8 to 32 for $M = 32$ and $P = 1$. This shows the limitation due to the increased switching overhead. A maximum per channel utilization of 0.8 can be achieved

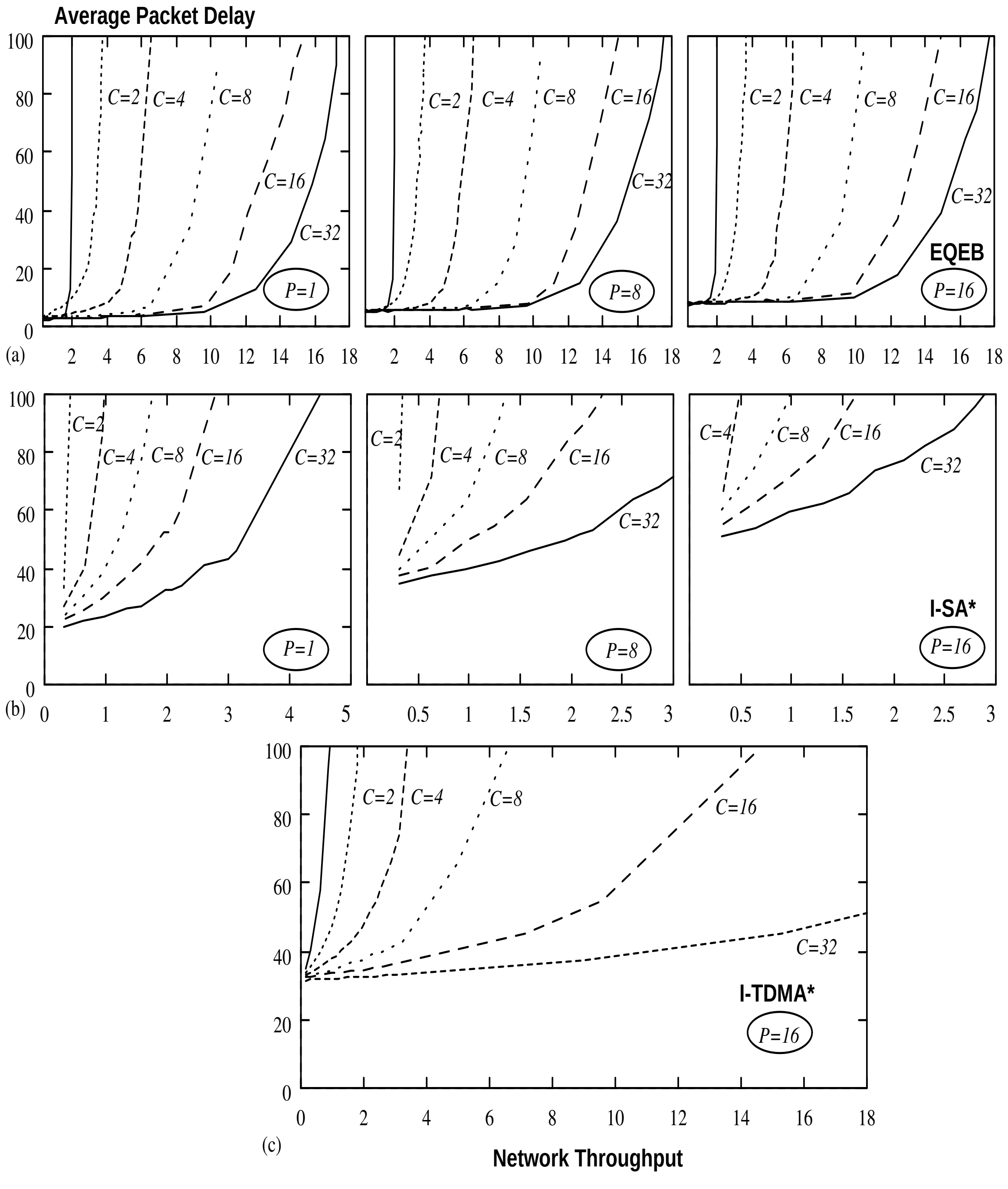

Figure 4.3: Performance of EQEB, I-SA* and I-TDMA* for $M = 32$ and varying C and propagation delay: (a) EQEB, (b) I-SA*, and (c) I-TDMA*.

using EQEB for large network sizes [104]. The delay characteristics are found to be low even for high loads. $\mathcal{S}_{max}$ is insensitive to propagation delay since slots are continuously generated by the head nodes. One disadvantage of using a DQDB scheme is the unfairness which can be addressed using bandwidth balancing techniques [107].

I-SA*: Fig. 4.3(b) shows the performance of I-SA* for $M = 32$, $P \in \{1, 8, 16\}$ and varying C. Increasing the number of channels improves performance because of the reduced collisions per channel. The maximum throughout using Slotted Aloha on C channels is often specified as $0.36C$ but this was not achieved due to the acknowledgment requirement and a finite population. The increase in $\mathcal{S}_{max}$ is approximately 300% when C is increased from 8 to 32 for $M = 32$ and $P = 1$. Figs. 4.3(b) also depicts degradation in performance with increasing propagation delay. Since I-SA* is acknowledgment based, it is extremely sensitive to propagation delay. The decrease in throughput for $M = 32$ and $C = 16$ is 30% when P is increased from 1 to 8.

I-TDMA*: Fig. 4.3(c) shows the performance of I-TDMA* for $P = 16$, $M = 32$ and varying C. I-TDMA* is independent of propagation delay except the linear increase in delay. The time multiplexed nature of I-TDMA* results in an average delay of $M/2$ even under light loads, which limits its scalability. The protocol is characterized by stability under heavy load and attains the maximum theoretical throughput of C.

The above discussion assumed that zero protocol processing and transmitter tuning time. Processing latency (α) has a significant impact on protocol performance. A node may switch between channels for consecutive packet transmissions. All the protocols except EQEB incur one tuning period per packet. Packet transmission in EQEB involves transmitters on both the buses (for request and data transmission) which increases the impact of tuning latency. The latency can be overlapped to a certain extent through the estimated

Protocol	Delay	% Increase in Delay				Thpt.	% Decrease in Thpt.			
	$\alpha = 0$	$\alpha = 1$	$\alpha = 2$	$\alpha = 4$	$\alpha = 8$	$\alpha = 0$	$\alpha = 1$	$\alpha = 2$	$\alpha = 4$	$\alpha = 8$
D-Net/C	24.4	45.5	81.1	197.1	434.4	9.1	14.3	25.3	46.2	64.8
Fairnet	55.5	17.8	35.9	87.6	193.7	5.7	10.5	21.1	36.9	54.4
B-TDMA	20.1	0.0	55.2	181.6	471.2	9.9	0.0	18.2	33.3	64.7
EQEB	9.9	57.7	140	328.6	600.1	12.7	11.0	21.3	37.4	58.1
I-SA*	99.4	12.6	19.3	41.8	91.7	3.9	7.7	12.8	25.6	41.0
I-TDMA*	20.0	0.0	55.2	181.6	471.2	9.9	0.0	18.2	33.3	64.7

Table 4.2: Comparison of protocols to study impact of processing latency for $C = 16$, $M = 32$, $P = 1$, $\lambda = 0.5$, and $\alpha \in \{0, 1, 2, 4, 8\}$. The table shows the actual value of delay and throughput for $\alpha = 0$ and the percentage variation for other values of α.

delays [104]. System performance degrades with increasing α for all the protocols. The severity of performance degradation is shown in Table 4.2 for $M = 32$, $C = 16$, $P = 1$, $\alpha \in \{0, 1, 2, 4, 8\}$. EQEB is seen to be most sensitive to α because of the double tuning time. It is possible to completely eliminate the impact of α in I-TDMA* and B-TDMA if $C(\alpha + 1) \leq M$ as shown in [87]

4.3.2 Performance under Client/Server Model

Fig. 4.4 compares the performance of the protocols under the client/server model for $C = 16$, $M = 32$, $P = 16$ and $\beta \in \{0.1, 0.5, 0.9\}$ where β represents the fraction of traffic directed towards the server.

Fig. 4.4(a) presents the performance for $\beta = 0.1$ which is close to the performance under the uniform reference model where $\beta \approx 0.03$. Increasing β affects other nodes sharing a home channel with the server. This impact can be reduced by isolating the server home channel at system startup. I-SA* is significantly affected by the increased traffic on the server channel due to the increased collisions. Providing per channel queues eliminates head-of-line effects for packets destined for other channels [92]. In D-Net/C, EQEB and

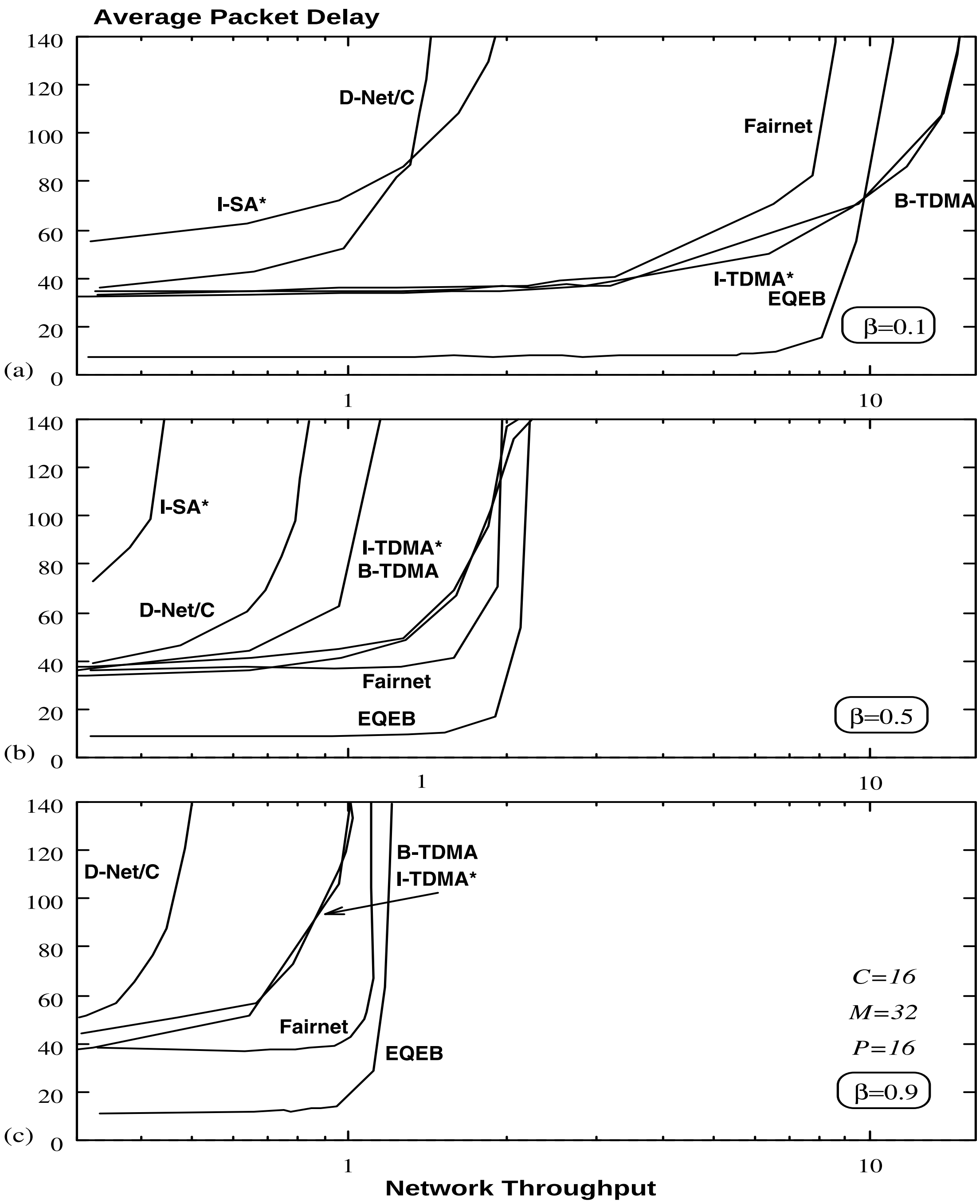

Figure 4.4: Performance comparison of protocols with client server model for $C = 16$, $M = 32$, $P = 16$ and $\beta \in \{0.1, 0.5, 0.9\}$.

I-SA*, the node examines the queues at the beginning of a slot in a cyclic order and selects the first non-empty queue. All channel queues are examined fairly and the server traffic is transmitted immediately if other queues are empty. The server channel saturates earlier than other channels, but that does not affect traffic on other channels if separate queues are used. In a similar way, these three protocols are adaptive to traffic patterns other than uniform reference and client/server models.

Fairnet provides better access to server channel queue since transmission probabilities are based on a priori knowledge of traffic distribution. It is not easily adaptive to statistical fluctuations in traffic. I-TDMA* and B-TDMA are fair in allowing equal access to all channels. However, both are not optimal since the channel allocation policy is static irrespective of traffic distribution. This leads to poor utilization under a client-server traffic since a node may remain idle in a slot despite a backlogged server channel queue. Figs. 4.4(b)-(c) compares performance impact for $\beta = 0.5$ and $\beta = 0.9$, respectively, showing that Fairnet provides the best delay characteristics under heavy and unbalanced loads.

4.3.3 Discussion

In this section, the protocol characteristics with respect to synchronization requirements, sensitivity to propagation delay and processing latency, network scalability and fault tolerance are examined.

Transceiver Cost: An objective of protocol design is to reduce the number of transmitters and receivers required per node. In particular, to reduce the number of tunable components, and relax the design constraints (to reduce cost) of the tunable devices by reducing parameters such as the required switching speed and the number of channels formed. The

Protocol	Topology	TT	FR	Tun. Tap
D-Net/C	FUB	M	M	M
Fairnet	FUB	M	M	M
B-TDMA	FUB	M	M	M
EQEB	DUB	2M	2M	2M
I-SA*	Star	M	M	0
I-TDMA*	Star	M	M	0

Table 4.3: Transceiver requirements of the protocols.

protocols defined in the previous section are based on receiver channel pre-allocation. Each node is required to have a minimum of a fast tunable transmitter and a fixed receiver. In addition, bus based systems require a sense tap to sense activity on the bus. The sense tap is required to be tunable to sense traffic on all channels. The sense tap need not have the full capabilities of a tunable receiver and is less expensive. In addition, a slot/locomotive generator for all channels is required at the head node for the implementation of all the bus based schemes. Table 4.3 summarizes the system transceiver requirements for a system with M nodes and C channels imposed by the protocols – per node requirements for data and special head node generator requirements for slot/locomotive generation. A laser array eliminates the requirement of a special locomotive generator. This requires that all channels be synchronized with respect to start of slots.

Synchronization: Network synchronization is required since no common clock is maintained. D-Net/C does not require frame level synchronization since nodes transmit a packet after sensing EOC from an upstream node. However, a preamble is required. EQEB, B-TDMA and Fairnet require strict frame synchronization since data is transmitted in fixed length slots. In general, frame synchronization is easier to implement on the bus using the head nodes. I-SA* and I-TDMA* have similar frame synchronization requirements. The absence of a central synchronization node and the different propagation delays from each node to the star have to be accounted for in providing synchronization.

Propagation Delay: The start of each round in D-Net/C is initiated after the locomotive generator detects end-of-train (EOT) on the inbound channel. The cycle length is dependent on the number of backlogged nodes and the end-to-end propagation delay, and the throughput of D-Net/C is to inversely proportional to propagation delay. Increasing the channels reduces the load per channel but does not reduce performance impact due to propagation delay. Fairnet, EQEB and B-TDMA de-emphasize the impact of propagation delay since slots are constantly generated by the head node. I-SA* is sensitive to propagation delay because of its dependency on acknowledgments. Each packet requires at least $2P + 2$ slots which includes propagation delay and transmission time for data and acknowledgment. I-TDMA* is not sensitive to propagation delay because of its collision-less nature and lack of acknowledgments.

Scalability: Adding/deleting a node results in a modified channel cycle for I-TDMA* and B-TDMA. All nodes have to recompute the channel cycle to accommodate the change in system size. D-Net/C, EQEB and I-SA* require all nodes to record the addition/deletion, whereas EQEB also requires recording the new node's location. Fairnet is the most sensitive to change in network size since all nodes have to recompute transmission probabilities during which the network is inactive. The impact of an increase in M on the performance of D/Net-C, I-TDMA* and B-TDMA is identical since each channel cycle is directly proportional to M. In EQEB, increasing M increases the size of the distributed queue and hence access delay. I-SA* is scalable in the sense that performance can be maintained so long as the ratio of M/C is maintained, but only in the case of low propagation delay [87].

Increasing C reduces the traffic per channel and improves performance for all protocols as seen from Figs. 4.2-4.3. However, when tuning latency is high, increasing the number of channels tends to offset the performance improvement due to more increased channel switching overhead. In I-TDMA* and B-TDMA, cycle length is determined by the relation between M, C and α and allocation schemes vary accordingly [87]. Fairnet requires recomputation of transmission probabilities for all nodes with variations in C.

Fault tolerance: The topological factors of fault tolerance have been studied earlier in Section 4.1. The media access aspects of fault tolerance are studied here. The bus based protocols require a head node for generating either the locomotive or the fixed slots. If the head node fails in Fairnet, B-TDMA and EQEB, the network operation on that bus is suspended temporarily. However, the adjacent nodes can always assume head node functionalities if they sense that the head node is not functioning. Since the head node requires C transmitters and C receivers to generate slots on all channels, adjacent nodes that assume the role of head node need to be provided with the additional transmitters/receivers. A laser array transmitter can eliminate the need for specialized locomotive generators at each node. I-SA* and I-TDMA* have excellent fault tolerance since failure of a single node does not affect the protocol operation of the rest of the network.

Protocol Complexity: Protocol complexity is a critical feature in protocol design for high speed networks. In I-SA*, the transmitter and receiver require processing in each slot to keep track of unacknowledged packets which increases processing overhead per packet [87]. In I-TDMA*, nodes need not maintain acknowledgment information but the complexity of network synchronization to support time division multiplexing is a drawback. The bus based protocols require each node to sense carrier for protocol operation. The ratio of carrier sense time to packet transmission time is significant at high transmission speeds, adversely affecting performance. In EQEB, Fairnet and B-TDMA, nodes must examine

Protocol	Delay	% Increase in Delay			Thpt.	% Decrease in Thpt.		
	$P = 1$	$P = 4$	$P = 8$	$P = 16$	$P = 1$	$P = 4$	$P = 8$	$P = 16$
D-Net/C	24.4	225.4	512.7	1098.4	9.1	48.3	68.1	81.3
Fairnet	55.5	6.1	11.2	24.1	5.7	0.0	0.0	0.0
B-TDMA	20.1	17.4	34.8	73.6	9.9	0.0	0.0	0.0
EQEB	9.9	5.1	21.2	51.2	12.9	0.0	0.0	0.0
I-SA*	99.4	10.1	14.3	28.7	3.9	8.0	10.4	18.1
I-TDMA*	20.0	17.4	34.8	73.6	9.9	0.0	0.0	0.0

Table 4.4: Performance of protocols for $C = 16$, $M = 32$, $\alpha = 0$, $\lambda = 0.5$, and $P \in \{1, 4, 8, 16\}$. The actual delay and throughput are presented for $P = 1$ and percentage increase/decrease for other values of P.

each slot for status bits before transmitting. There is also the tighter requirement that a node transmit in the current slot sensed. The protocols themselves have been designed so that there is minimal processing overhead per packet except for carrier sensing and header processing. Fairnet may incur extra overhead if the nodes adaptively modify their transmission probabilities [103].

Case Study: Consider a network with 32 nodes and 16 channels. The star can easily support this system size without any amplification for a power margin of 40 dB, whereas both DUB and FUB require optical amplifiers. For low propagation delay of $P = 1$, D-Net/C and Fairnet achieve a per channel efficiency of 60% and B-TDMA an efficiency of 100%. EQEB is able to achieve higher throughput because of the dual bus but the per channel efficiency is 40%. The minimum delay is higher for B-TDMA because of time multiplexing and the delay is higher for Fairnet since channel queues are selected probabilistically. I-SA* has low delay under light traffic but collapses under heavy traffic. The channel efficiency is roughly 32%. I-TDMA* offers higher delay under light traffic, is stable under heavy traffic and achieves maximum per channel efficiency of 100%, but its main drawback is poor scalability with increasing M.

Table 4.4 summarizes the protocol performance for $\lambda = 0.5$. For higher propagation delay of $P = 16$, D-Net/C suffers reduction in maximum throughput of up to 80%. I-SA* is sensitive to increase in P but the throughput drop is less than 20% for $P = 16$. Fairnet, EQEB, B-TDMA, and I-TDMA* are insensitive to propagation delay with respect to maximum throughput.

The advantages of the star over the bus for the single channel photonic system are extendable to a multi-channel optical network with amplification. The fault tolerance and fanout characteristics of the star are better than that of the bus but the bus still retains the advantages of implicit node ordering and lower fiber cost. However, protocol implementation on a bus requires more transceiver components (Table 4.3). The time division multiplexed schemes (B-TDMA and I-TDMA*) were shown to offer high throughput under heavy traffic but were sensitive to system size. The balance still remains tilted in the favor of star networks because of better fanout, lower amplification cost, better fault tolerance, improved protocol performance and implementational simplicity; its disadvantages are the higher fiber cost and complex network synchronization.

4.4 Summary

The design of photonic local area networks is crucially dependent on the network interconnection topology. The bus topology is no longer considered impractical for optical networks due to improvements in optical amplifiers. This chapter examined the bus and star topologies with optical amplification with respect to fanout, access arbitration, scalability, cost and fault tolerance. Access arbitration schemes for the multiple multi-access channels were evaluated for both the topologies in terms of performance, scalability, complexity and fault tolerance. The star was shown to have better fault tolerance, fanout, protocol performance and implementational simplicity at the expense of increased fiber cost and complex network synchronization.

The next chapter presents a scalable, all-optical, hierarchical parallel architecture based on an optical interprocessor communication network. Access protocols for this architecture supporting a Distributed Shared Memory (DSM) organization are studied in detail.

Chapter 5

Protocols for DSM Architecture

In this chapter, a hierarchical multi-processor architecture based on an optical interprocessor communication network introduced in [18] is presented. Optically interconnected distributed shared memory (OIDSM) architectures have been studied in [108, 109]. This chapter studies access protocols for this hierarchical architecture in support of the DSM organization.

Section 5.1 describes the hierarchical architecture. A prototype based on this architecture, referred to as *Lightning*, is currently under implementation. Section 5.2 provides the classification of parallel computer architectures according to memory organization. Section 5.3 describes the access protocols for this architecture. Section 5.4 describes a semi-Markov model for a hybrid reservation/pre-allocation protocol. Section 5.5 compares the performance of the hybrid protocol to a pure TDM based pre-allocation protocol. Section 5.6 discusses synchronization issues for implementation of the protocols. The advantages and disadvantages of the two protocols are discussed.

5.1 Lightning Architecture

This section describes the architecture of the DSM optically interconnected multi-processor system. The objective of this hierarchical architecture is to achieve scalability yet avoid the requirement of multiple wavelength tunable devices per node. Furthermore, single-hop communication is achieved: a packet remains in the optical form from source to destination and does not require intermediate routing. This all-optical characteristics is achieved for both data transport *and* control. Path establishment via electronic intervention is not needed.

The wavelength multiplexed hierarchical structure features wavelength channel re-use at each level, allowing scalability to very large system sizes. It employs a wavelength partitioner (denoted as Λ-x) between different levels of the hierarchy without electronic intervention.

An advantage of this structure is its ability to dynamically vary the bandwidth provided to different levels of the hierarchy. The system adapts to the users instead of the users adapting to the system. Bandwidth reallocation is invisible to the users, and is initiated by the system to balance the per channel intensities. The hierarchy resembles a general m_i-ary tree, where an i-level node branches to one parent and m_i children. A single cluster is illustrated in Figure 5.1. The *processors* (0-level nodes) are located at the leaves and the internal nodes are wavelength/spatial routing switches (Λ-x). The result is a system with a unity degree per processor yet unity diameter: any two nodes can communicate without any intermediate routing (this includes no electronic control intervention). This is very significant since the intermediate routing and control processing latencies may be much larger than the packet transmission time in an optical environment. An additional advantage is the independence of the number of channels in the network and the number of nodes which preserves the scalability of the architecture.

A mixed radix system [110] is used to represent the node numbering. Let M (the total

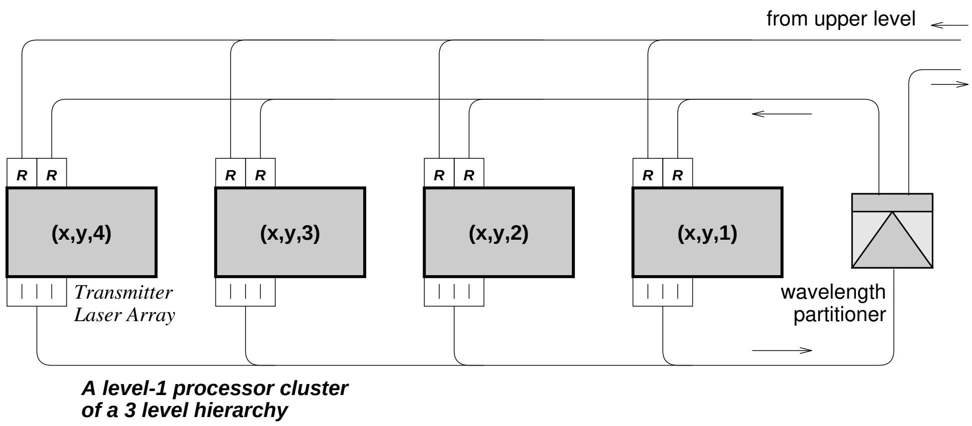

Figure 5.1: Example of Level-1 interconnection.

number of processors) be a decimal integer represented as a product of r factors:

$$M = \prod_{i=1}^{r} m_i \tag{5.1}$$

The processor identifier P, $1 \leq P \leq M$, can be represented as an r-tuple

$$P = (p_r, p_{r-1}, \ldots, p_1) \tag{5.2}$$

where $1 \leq p_i \leq m_i$ for all $1 \leq i \leq r$ and r denotes the number of hierarchical levels.

The term *processor* is used to denote a node at the leaves of the tree (a 0-level node). An actual system might place m_0 processors at each 0-level node, perhaps interconnected through a shared metal bus to amortize the cost of the communication subsystem over multiple processors depending on the cost, performance and typical communication requirements of the individual processors.

Note that the node numbering in Figure 5.2 is generalized to provide numbering for the Λ-x switches. This enables control scalability as described below. The rest of this section describes the Λ-x component and defines the LIGHTNING topology.

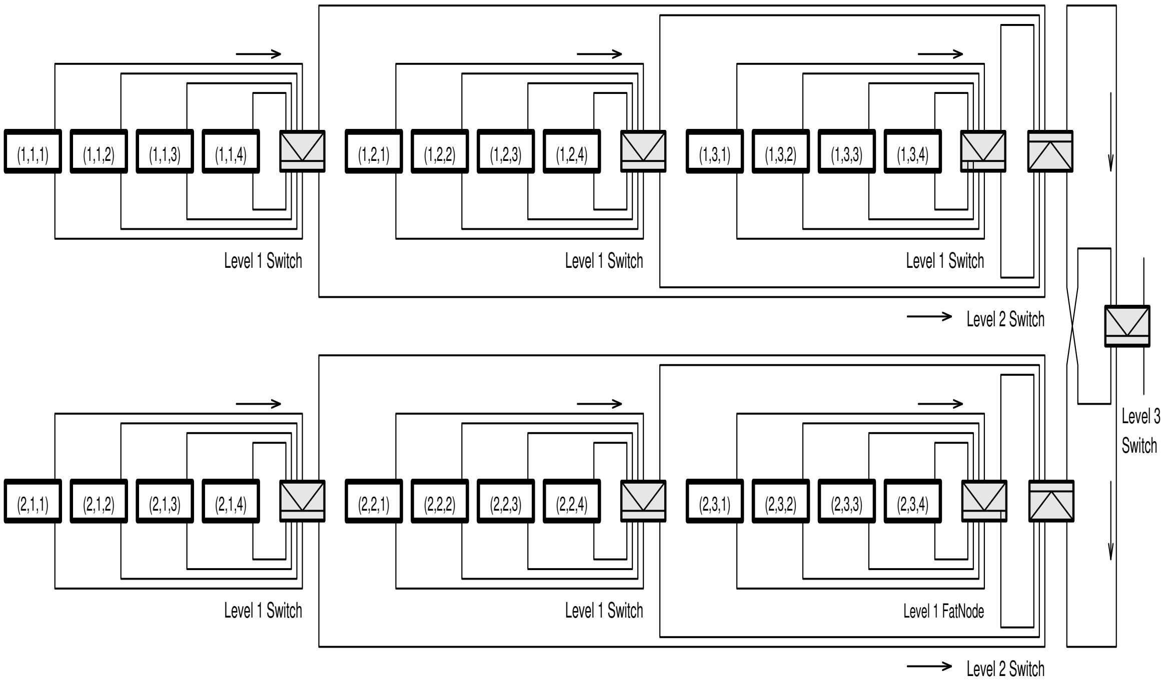

Figure 5.2: Example of 24 node 3 level structure.

Λ-x: An i-level Λ-x is used as a wavelength-selective space-division switch to select a subset of wavelengths to switch. A subset of the wavelengths is retained within the local cluster and the remaining channels are transmitted to the next level partitioner for higher level communication.

Spatial re-use of wavelength channels occurs at each of the r levels in this hierarchy. Let C denote the total number of wavelength channels and $\mathbf{C} = \{\lambda_1, \ldots, \lambda_i, \ldots, \lambda_C\}$ denote the set of wavelength channels.

An i-level Λ-x is configured to a *crossover channel* denoted as X_i, $\lambda_{X_i} \in \mathbf{C}$, where channels $\{\lambda_{X_i+1}, \ldots, \lambda_C\}$ are passed to the upper level, and channels $\{\lambda_1, \ldots, \lambda_{X_i}\}$ are contained locally for local intra-cluster communication. Connection patterns of channels to nodes at the cluster are governed by the allocation strategy of the access protocol. Essentially, this mapping is used to retain the local i-level traffic

$\{\lambda_1, \ldots, \lambda_{X_i}\}$, and forward the global traffic $\{\lambda_{X_i+1}, \ldots, \lambda_C\}$ to the $(i+1)$-level Λ-x switch.

Topology: Fig. 5.2 illustrates a 2 level structure with $M = 4 \times 4$. An i-level Λ-x, $1 \leq i \leq r$, partitions the traffic it receives according to its space-wavelength configuration. There are two major modes of operation:

▷ *Symmetric Organization*

All equal level Λ-x switches are configured to the same partition point. The partition points are defined as $\mathbf{X} = \{X_1, X_2, \ldots, X_r\}$ where channels $\lambda_{X_{i-1}+1}$ through λ_{X_i} are used to support i-level communication ($X_r = C$ and $X_0 = 0$). The configuration remains fixed at the partition point and is not altered until the system is reconfigured to adapt to a change in the locality characteristics.

The partition points $\mathbf{X}$ are shifted during reconfiguration which allows the system to adapt to a shift in traffic locality characteristics and dynamically reconfigure the amount of communication bandwidth provided at each level of the hierarchy.

The total number of channels achieved due to the combination of spatial and wavelength multiplexing for an r-level LIGHTNING is

$$\mathcal{C} = \sum_{j=1}^{r} (X_j - X_{j-1}) \prod_{k=j+1}^{r} m_k \tag{5.3}$$

For example, with $M = 16$ as in Fig. 5.2 with $C = 4$ and $\mathbf{X} = \{3, 4\}$, a total of $\mathcal{C} = 13$ separate channels are effectively created achieving a maximum network capacity of about 2.5Gbps when the individual channel data rate is 200Mbps.

▷ *Asymmetric Organization*

The objective is to allow peer Λ-x switches to have flexible control of the partition-point of the channel allocation.

This would allow a level-i cluster to alter the amount of bandwidth made available to its nodes independent of what other peer level-i clusters desire. Furthermore, an additional advantage is that a level-i cluster could completely isolate itself from the rest of the network. This might be attractive for security and high-demand situations where local level-i traffic could be contained and all bandwidth allocated for local communication.

Support of this objective impacts the decentralized control strategy, and will be mainly useful during the second stage of the project. Network control is composed of media access control, and reconfiguration control. Access protocols are described in Section 5.3. We have split the reconfiguration control into two portions: at the network interface and at the operating system. The network interface will be used for high speed traffic load monitoring, while operating system level routines will be used to decide when and to control reconfiguration. Therefore, since the reconfiguration strategies will be implemented at the operating system level we will not preclude stage 2 if we optimize the design at stage 1 for our DSMM application since they can easily be modified. The home channel allocation scheme of the media access protocol is impacted with decoupled clusters and must be generalized to support the required flexibility.

5.2 Parallel Computer Memory Organization

Parallel computer may be classified according to memory organization as *shared memory*, *distributed memory*, and *distributed shared memory* systems.

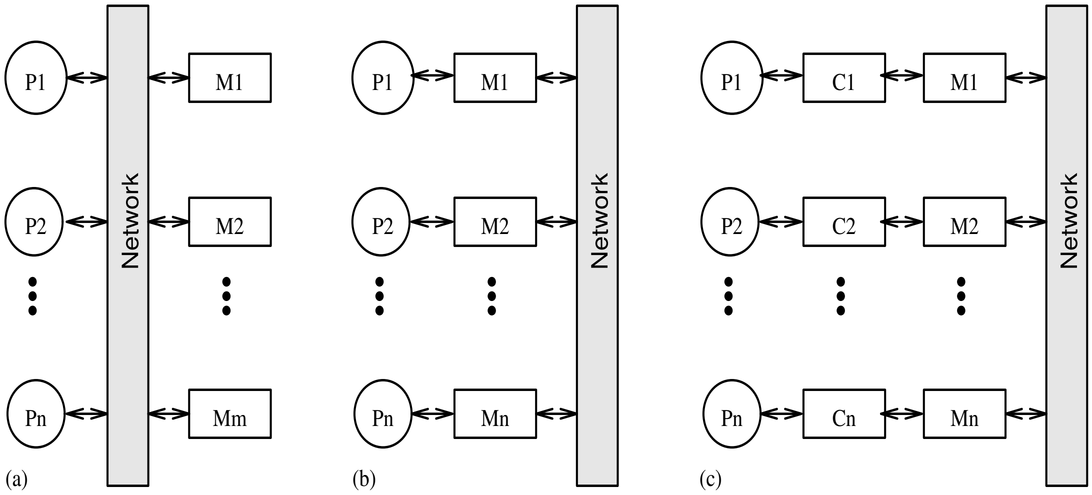

Figure 5.3: Parallel Computer memory organization: (a) shared memory, (b) distributed memory, and (c) distributed shared memory.

5.2.1 Shared Memory

Shared memory systems require a single global address space to be shared by all the processors. A typical configuration of such a system with N processors and M memory modules is shown in Fig. 5.3(a). The communication network can be packet or circuit switched. Task execution on this class of systems involve code and data references by the processors to the shared memory modules. Contentions occur at both the communication network and memory modules. Each processor may maintain a local cache to reduce frequent network access. Increased hardware complexity to maintain coherence between the cache and main memory may be used to reduce contention. The principal advantages of this approach are ease of programming and good performance with small systems. Limitations due to the memory contention and communication bandwidth inhibit support of large system sizes.

5.2.2 Distributed Memory

Each processor in this class has its own memory and its private address space as shown in Fig. 5.3(b). The processors are typically interconnected through a packet switched communication network. Application development for distributed memory systems face difficulties: the message passing nature of sharing data and the relatively small size of the individual memory. The message passing mechanism involves packet formation, routing and decoding. This complexity, together with the relative programming difficulty, has hampered distributed memory systems from becoming a more dominant approach to parallel architecture. Distributed memory systems have a strong advantage in their ability to support a large system size, systems which can extend into the massively parallel region.

5.2.3 Distributed Shared Memory

Each processor in a DSM system is associated with a portion of the system address space, which can be accessed by the other processors through message passing [19]. A general configuration for a DSM is shown in Fig. 5.3(c). Each processor typically executes from its associated memory as in distributed systems. However, all processors have access to the entire address space as with shared memory systems. The required blocks are brought into the address space allotted to the processor, and used for process execution. This configuration is very attractive in that it combines the topological advantages of distributed systems with the programming advantages of shared memory systems. The application base of the resulting system is wider than for both of the two approaches individually, moving the system in the direction of general-purpose parallel computing.

Distributed shared memory systems are constructed as a distributed memory system, but appear to the programmer as a shared memory system. The shared memory appearance is obtained by mapping the memory modules located at the processor into a global address space. Each processor is associated with its own memory which can also be accessed by other processors through message passing. In this case, message passing is handled by the system and makes the memory access transparent to the programmer.

Message passing involves transmission and reception of request/response packets. For example, in order to support the view of shared memory in a distributed memory system, the following steps are involved. A node identifies the target node that contains the target memory block. This is followed by transmission of a request packet to the target node. The requested block is transmitted by the target node. This might lead to contention problems at the network and the global memory module.

An *optically interconnected distributed shared memory* system has been proposed and analyzed in [111]. The combination of the concepts of shared memory, distributed memory and optical interconnects result in a system with the advantages of all three: ease of programming, scalable systems, and fast low-cost performance. The network is longer the principal constraint and has changed the perspective of various issues such as processor migration and load balancing.

This section briefly introduced the different memory organizations for parallel and distributed systems. The following section provides the protocol definitions for the *Lightning* architecture described in Section 5.1.

5.3 Protocol Definition

This section describes the hybrid access protocol. It is defined for a single level system, and then generalized to the multi-level architecture defined in Section 5.1.

The protocol considered here is based on channels pre-allocated for data reception where each node receives on a specific channel referred to as its *home channel*. Each processor is assumed to possess a tunable transmitter constructed from a laser-array and a receiver tuned to its home channel.

The traffic characteristics of the system play a crucial role in the access protocol design. Define two classes of traffic:

Class A: small amounts of data, such as control information generated by the cache control mechanism and the operating system.

Examples are memory block requests, invalidations, cache level acknowledgments, application level low-latency messages, operating system control information, bandwidth reconfiguration (for the multi-level network) and other network management packets.

Class B: large amounts of data, such as a memory block.

This class of traffic generates a reservation control packet for media access if a reservation based protocol is used. In this case, Class A control information can be piggybacked on a Class B reservation packet.

Memory block length is expressed in terms of control packet length: let L denote the ratio of memory block packet length to control packet length.

Often only a control packet needs to be sent to transmit control signals. Block transfers occur only a fraction of the time. This provides the motivation to develop a protocol that can reduce communication latency and transmit unfragmented memory block packets. Define γ as the fraction of Class B packets generated in the system. The distribution of packet types is determined both by the media access protocol and the cache coherency protocol. Cache coherency protocols for multiprocessor systems can be broadly classified as *snoopy* and *directory based* [21]. The write-invalidate and write-update *consistency commands* generate varying levels of cache control traffic. Snoopy cache protocols use some form of broadcast mechanism (for fast invalidations, etc.), and directory based schemes store information on where copies of blocks reside and usually require explicit invalidations. The value of γ depends largely on the amount of data sharing which is determined by the address references generated. Snoopy schemes generate less control packets since all nodes can read the invalidate commands. Directory based schemes generate more node-to-node traffic and are characterized by many more cache control packets.

The following sections describe the time multiplexed protocol I-TDMA [92] and the proposed hybrid protocol FatMAC. Each node requires a tunable transmitter and a fixed receiver subsystem. Each node receives traffic on a pre-determined channel. A source node tunes its transmitter to the home channel of the destination node and transmits according to the access protocol. A node receives and processes all traffic along its home channel. In a multi-level system, each node has one receiver per level tuned to the home channel for that level. A source node can compute the home channel of the destination node in a decentralized fashion through a simple computation based on the channel allocation policy. Let N_i and C_i respectively denote the number of stations and channels within a level i cluster. Node m_k is assigned λ_j as its home channel based on the allocation policy, where $j \in \{0, 1, 2, \ldots, C_i - 1\}$ and $0 \leq k \leq N_i - 1$. An interleaved channel allocation scheme assigns channel λ_j to station m_k where $j = k \bmod C_i$.

5.3.1 Single-Level Protocols

This section provides the protocol definition for a single-level network such as a passive star coupled network.

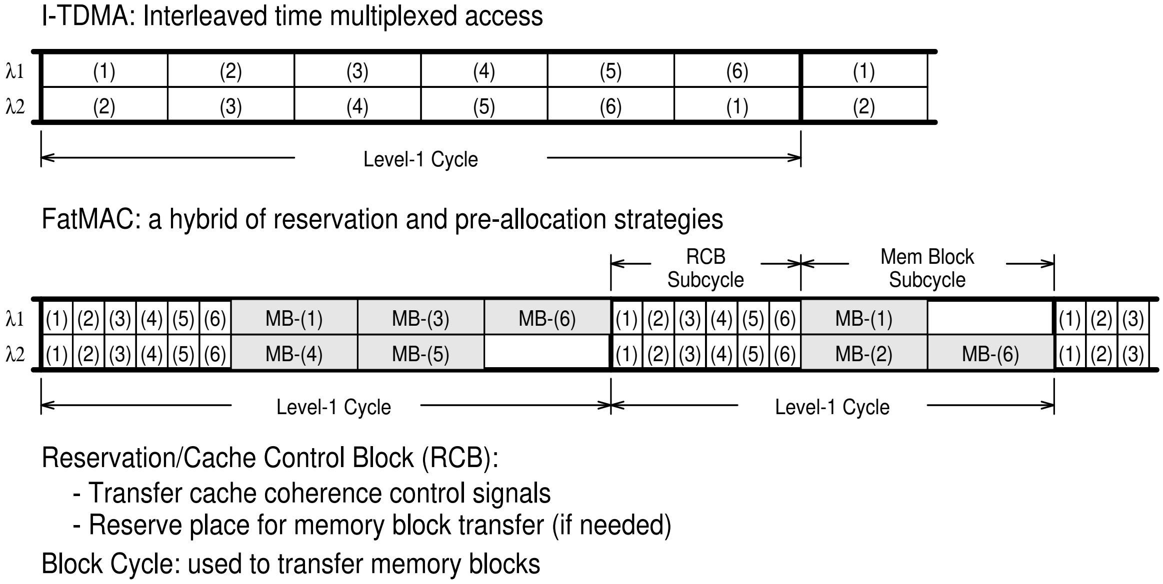

Figure 5.4: Assignment map for single level I-TDMA and FatMAC protocols for $M = 6$ and $C = 2$: (a) I-TDMA – one slot equals memory block packet, and (b) FatMAC – one slot equals control packet.

I-TDMA

This is a multi-channel extension to basic TDMA. Time is slotted on each channel where each slot corresponds to a memory-block packet transmission time. Every node in the system has a chance to transmit on each channel per cycle. Determining the slot which is assigned to a particular source-destination pair is simple and decentralized. Fig. 5.4 shows an example of a source node/channel allocation map for a single level system with $M = 6$ and $C = 3$. A node is assigned a total of C slots per cycle and remains idle for the remaining $M - C$ slots. However, as shown in the figure, the channels are fully allocated. I-TDMA has one queue per channel to partition traffic based on destination channel. A node can transmit up to C packets per cycle. Cache request packets may be required to transmitted in order, depending on the required level of consistency and partitioning packets into separate queues may result in out-of-order transmission. One way to ensure ordered delivery is to provide a separate queue for request packets. The request packets may be piggybacked on outgoing Class A or Class B packets.

Although I-TDMA can achieve the maximum theoretical throughput, its latency characteristics limit scalability. Transmitting 8 Kbyte packets at 200 Mbits/s requires 328 μs. If there are 16 nodes in the cluster, the average delay under light loads is very high (approximately 2.6 ms). The multi-level version of the protocol helps to reduce the limitations caused by the large cycle synchronization delays, but does not provide the adaptability required for the target application. Another possible extension to I-TDMA is to generalize the number of slots per frame where slots are allocated by a resource scheduler based on demand such as was proposed with *universal time-slots* [112]. This would avoid the static bandwidth allocation of I-TDMA, providing a more dynamic environment, but creates a connection-style environment where link setup needs to be negotiated. This strategy works well with voice or video, since the connection duration is usually orders of magnitude longer than the setup time, but not when the link is only needed on a transient basis.

In the proposed system where there are many more Class A packets than Class B packets, the delay suffered by short packets is high. There is also reduction in throughput efficiency since slots are not fully utilized when short packets are transmitted. I-TDMA is also unable to support broadcast. Broadcasting is very desirable, and significant system-level performance improvement is possible in support of cache coherency mechanisms such as the snoopy scheme.

The FatMAC protocol described in the next section has been designed to eliminate some of the disadvantages and accommodate broadcasting and variable packet sizes.

FatMAC:

This protocol has the following main goals: (a) minimize the latency especially under light loads; (b) minimize utilization loss and improve the system throughput for small values of γ; (c) provide a broadcast mechanism which might be required by the applications or by cache coherency schemes; and (d) support variable sized packets without segmentation.

FatMAC combines reservation access and pre-allocated receivers. The satellite access protocol studied in [113] used a time multiplexed reservation cycle followed by a data cycle, overlapping the long satellite propagation delays through interleaving reservation and data cycles. FatMAC generalizes this two-phase approach to a multi-channel environment. Typical WDM based reservation access protocols allocate one of the channels as a dedicated control channel. This is not attractive in the target environment since the number of channels available is usually small. FatMAC is a hybrid approach using reservation access but without a reserved control channel.

Transmission in FatMAC is organized into cycles where each cycle consists of a control phase and a data phase. The control phase operates in a broadcast environment. A control packet sent during the Reservation/Control Block (RCB) phase of a cycle may perform up to two functions: transmit a block reservation and/or include a Class A packet waiting to be transmitted through a small multi-purpose payload. A reservation specifies the destination, the data channel and the packet size (if variable sized packets are supported). Access during the control phase is TDM-based mainly to preserve a collisionless environment but other access strategies are possible. FatMAC exploits the orientation of the transmitter. Since it is constructed as a laser array, all channels can be simultaneously driven to achieve broadcast.

A control packet may contain a cache control signal such as an invalidation and not need to reserve a data slot. This allows the average cycle length to be significantly reduced. A fast broadcast facility is achieved through this multi-purpose control packet. Broadcast is achieved by modulating all lasers (allocated to level-i) of the array with the the same information. The operation with a single tunable laser would be similar, except that it would explicitly transmit on each channel in a pipelined fashion and the overall reservation interval would be extended by $\mathcal{C}_i - 1$ control slots.

Time is slotted in FatMAC on control packet boundaries. In I-TDMA, time is slotted on memory block boundaries as seen from Fig. 5.4. FatMAC requires one queue per transmitter since a node can transmit only one packet set per cycle. Figure 5.4 shows the allocation maps for single level FatMAC for $M = 6$ nodes and $C = 2$ channels. The transmitter behavior can be described as follows:

1. When a packet is ready for transmission, the node waits for its turn on the reservation phase (cycle synchronization delay).

   ▷ Access during the reservation phase is based on TDM to support collisionless transmission.

   ▷ Access to the reservation phase could be based on other strategies, such as Slotted Aloha, to further reduce packet latency. With SA, acknowledgments are not needed for the reservation phases but the sensitivity of the system to propagation delay is increased.

2. The control packet is broadcast to all nodes during the reservation phase. This takes advantage of the structure of the transmitter which is an array of distinct wavelength lasers rather than a single tunable laser.

   The control packet contains a class A packet and/or a block reservation if a block transfer is scheduled to follow the control packet.

3. The node waits until the end of the current control phase before it attempts to transmit the corresponding memory block if any.

4. If the node has a data block to transmit, a node calculates its offset within the data phase and transmits the memory block on the channel it specified in its control packet. Variable size blocks could be supported, but it is assumed in the analysis to follow that all nodes in a system use the same block size. Furthermore, it simplifies the network interface design if the block size remains constant.

   - ▷ Nodes that reserved access on the same channel transmit memory blocks in the order in which they transmitted their control packets.
   - ▷ The service mechanism in this case is first-come first-serve though a general scheduling algorithm may be used.
   - ▷ A reversing service during the control phase could improve fairness.

The length of each data cycle is the maximum data cycle length of the individual channels within the channel set (the channels allocated to that level). Each node receives all control packets and knows the start of the next cycle. In the event that a node misses a control packet (through buffer overflow or corrupted transmission), start of a new cycle is detected by the synchronization sequence.

Note that the average cycle length of FatMAC can be shorter compared to I-TDMA especially under light loads. This is due to the fact that FatMAC is slotted on control packet boundaries whereas I-TDMA is slotted on memory block boundaries. The cycle length in I-TDMA is ML slots independent of traffic characteristics. The minimum cycle length in FatMAC is M slots where all packets in the cycle are control packets. The data cycle length depends on the number of memory block packets transmitted per cycle.

I-TDMA may be slotted on control packet boundaries. This reduces the cycle length to M slots and improves delay performance for control packets. Memory block packets will be segmented across different cycles increasing the delay to about LM. For the rest of this

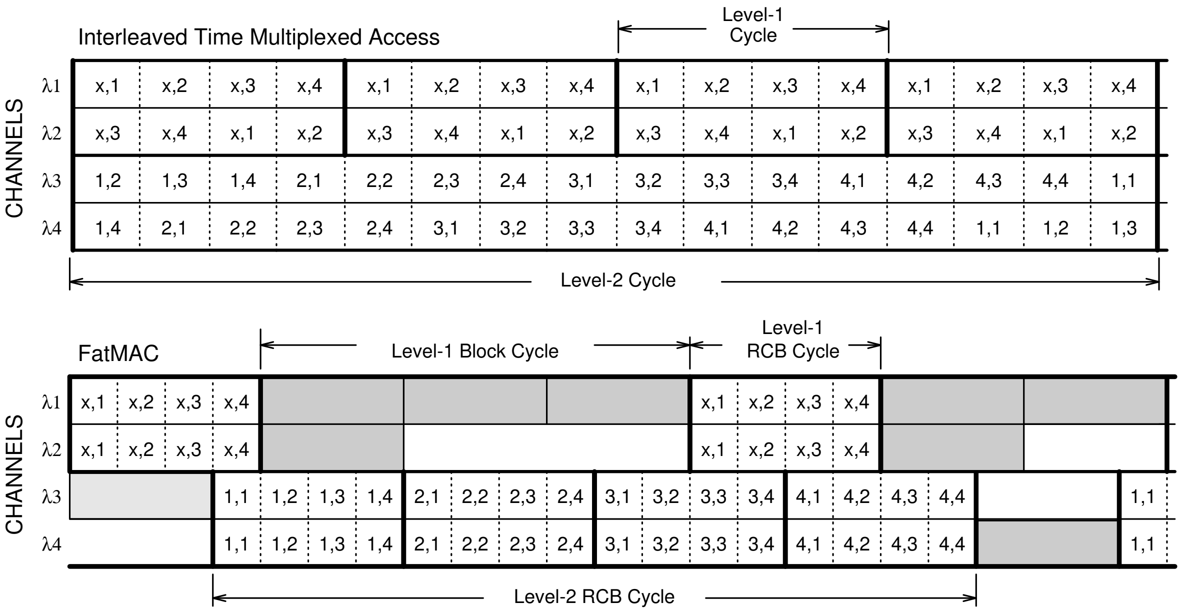

Figure 5.5: Assignment map for two level I-TDMA and FatMAC protocols for $M = 4 \times 4$ and $C = (2, 2)$: (a) I-TDMA, and (b) FatMAC.

paper, we will study I-TDMA slotted on memory block packet boundaries.

This approach trades channel utilization for reduced latency. There are advantages with this strategy: (a) collisionless transfer, (b) a significant reduction in latency under light loads when compared with I-TDMA, and (c) improved channel utilization for situations when γ is small since most I-TDMA slots will only be partially used. However, FatMAC does not match the maximum utilization achieved by I-TDMA.

Both protocols are collision free and do not require acknowledgment support under normal circumstances. However, there is always the possibility of corrupted or dropped packets, or loss of slot synchronization. The media access layer is free of providing acknowledgments which may be handled by higher level protocols. In the proposed system, transaction-level acknowledgments are handled by the cache coherence protocol.

5.3.2 Multi-Level protocols

This section describes the multi-level extensions of I-TDMA and FatMAC.

I-TDMA: Fig. 5.5(a) shows the extension of I-TDMA to a multi-level system. Source-destination pairs are determined based on their addresses and the current time slot as explained in [18]. Each node computes its assigned channel in each slot using an identical distributed algorithm ensuring conflict-free and fair access to the nodes. The cycle length in I-TDMA is fixed and is dependent only on the number of nodes in the system. High utilization can be achieved as with the single-level system at the expense of high latencies under light loads. As discussed earlier, another possible extension to I-TDMA is to generalize the number of slots per frame where slots are allocated by a resource scheduler based on demand such as was proposed with *universal time-slots* [112].

FatMAC: Fig. 5.5(b) shows the extension of single-level FatMAC to a multi-level system. A *cluster* is defined as a set of nodes and channels operating at one level of the hierarchy. For example, consider a 2-level 4×4 system. There are 4 clusters of 4 nodes each at level 1 and one cluster comprising 16 nodes at level 2. Each node is thus associated with one cluster at each level. Each of these clusters is associated with cycles composed of reservation and data phases as shown in the figure. Each node is allocated one control slot in each of its clusters' control cycles. At any instant, a node transmits on only one cluster cycle and participates in a cycle on at most one level.

Figure 5.5(b) shows a snapshot of a $M = 4 \times 4$ system with 4 channels. The cycle lengths are seen to be different for each level. The figure shows the loss of utilization when channels are only partially utilized within a cluster during the memory block transfer phase of a cycle. Since the node-channel allocation for FatMAC is not fixed as in I-TDMA, a node may be assigned simultaneous reservation slots on different levels. However, the cluster which a node selects at any time is uniquely determined by the current packet chosen. The downside is that the node may miss its reserved control slot.

Each receiver of a node is passive since it receives all packets transmitted on its home channel. It updates cycle length information with the associated cluster if a control packet is received. A data packet is forwarded to the memory subsystem if it is the intended destination. A node is constantly aware of the data cycle status in all its associated clusters. In the event that a node fails to receive a control packet, it can be resynchronized by a synchronization sequence at the start of the next cycle.

The performance of FatMAC and comparison to I-TDMA is investigated in the following section.

5.4 Analytic Performance Model and Validation

The performance of a simplified version of the protocol is analyzed through semi-Markov modeling. A semi-Markov model was developed in [95] for a multibus system. In particular, it developed the three state cluster which is inherent in the model developed below. A semi-Markov model is an approximation technique, useful because of the significant reduction in state space, and a comparison to simulation is used to ensure that the implied approximations are reasonable.

The analytic model enables prediction of channel utilization and average packet latency with varying system characteristics. Table 5.1 describes the notation used in the following analysis. The system is described in terms of normalized time. Time is normalized to control packet transmission time. A preliminary version of this model is presented in [114]. The model for the single-level protocol makes use of the following assumptions:

$\mathcal{A}_1$: *i.i.d.* behavior of the nodes and channels. Channel traffic will not be uncorrelated

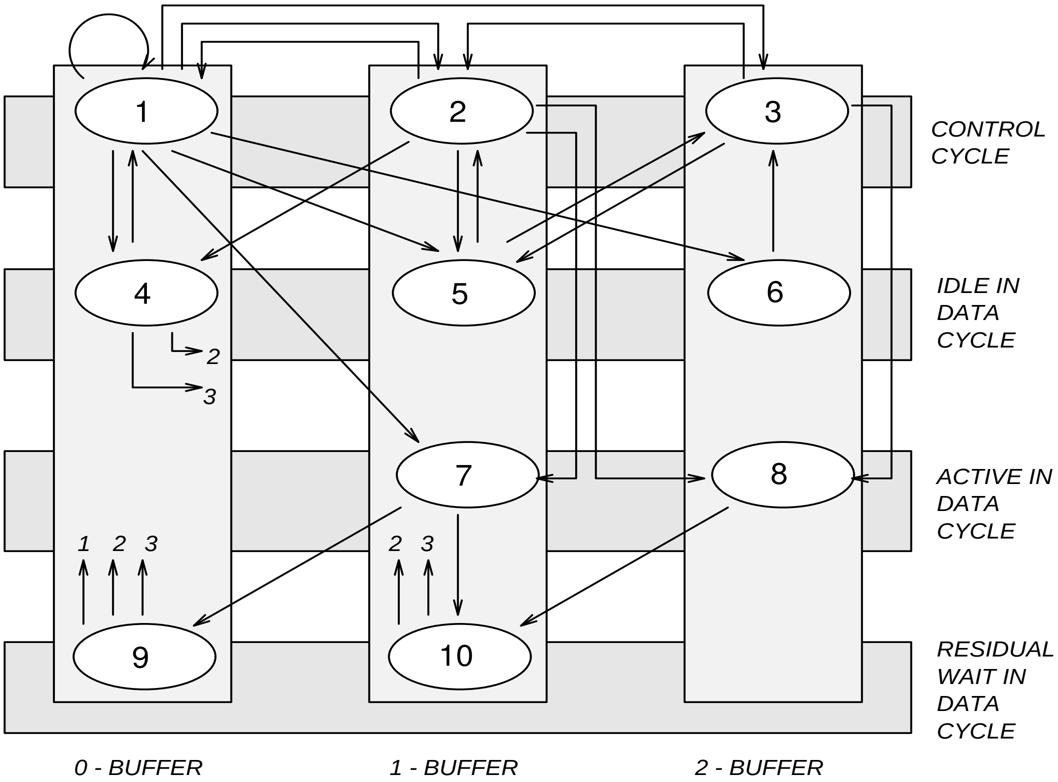

Figure 5.6: Semi-Markov analytic model for FatMAC protocol. Each transmitter has capacity to hold one packet when processing another packet. Table 2 lists the transition probabilities.

in general, due to common transmit queues, but this approximation simplifies the transition probability derivations and is shown to be reasonable in the comparison to simulation in Section 5.4.4.

$\mathcal{A}_2$: Time is slotted on control packet boundaries and packet generation per slot follows a Poisson process with rate λ.

$\mathcal{A}_3$: The traffic distribution is largely determined by reference of locality. Typically, reference locality would be exhibited within a cluster (level). Since the analytic model considers only nodes within a single level, the uniform reference model is used for performance evaluation. A packet generated a node m_i is targeted to node m_j with probability $\dfrac{1}{M-1}$ for $i \neq j$ and $0 \leq i, j \leq M-1$; and with probability 0 for $i = j$.

$\mathcal{A}_4$: A transmitter has a capacity to hold one packet when processing another packet. This is not an inherent assumption of the protocol and the model can be extended to include generalized transmitter buffer capacities as in [92].

$\mathcal{A}_5$: All memory block packets are of fixed length L times the length of a control packet.

The performance of the protocol is modeled by studying the behavior of the transmitter which changes states most often. The receiver is fixed at its home channel and does not actively affect protocol behavior. The following section describes the possible states of the transmitter and the transition probabilities between states.

5.4.1 State Definitions

Fig. 5.6 shows the semi-Markov process depicting the behavior of the transmitter. To summarize the steps involved in the transmission of a packet: A node generates a packet and waits for its turn in the next reservation cycle. There is a memory block transmission in the corresponding data cycle only if it is a class B packet. The states of the transmitter depict the behavior of the node as it sequences through these steps. The states represent one of the two cycles of the system: control or data. The state descriptions are as follows:

S_1, S_2, S_3 – These states represent the transmitter during the control cycle. The transmitter has zero, one and two packets queued on entering states S_1, S_2 and S_3 respectively.

S_4, S_5, S_6 – These states represent the transmitter being idle during the current data cycle. The transmitter has zero, one and two packets queued on entering states S_4, S_5 and S_6 respectively.

S_7, S_8 – These states represent block transmission during the data cycle. The transmitter has sent a memory block reservation in the preceding control cycle. The transmitter has zero and one packet queued on entering states S_7 and S_8 respectively.

S_9, S_{10} – These states are entered after completion of packet transmission in the data phase. This is the residual wait in data cycle until start of the next control cycle. The transmitter has zero and one packet queued on entering states S_9 and S_{10} respectively.

The probability of a transition from state S_i to state S_j is denoted as $p[i,j]$. The following derives the transition probabilities between the states in the process. The transitions between states not mentioned below do not occur. Fig. 5.6 does not label all state transitions for the sake of clarity.

Let τ_i denote the *average sojourn time* of S_i. A packet can be generated in any control slot of a cycle. Packet generation is assumed to be a Poisson process with a rate λ packets per slot per node. The probability that at least one packet is generated in a control slot is denoted as β where $\beta = 1 - e^{-\lambda}$. Let $\eta_i = e^{-\lambda\tau_i}$ represent the probability that no packet is generated during τ_i; $\theta_i = \lambda\tau_i e^{-\lambda\tau_i}$ the probability that exactly one packet is generated during τ_i; $\rho_i = 1 - \eta_i - \theta_i$ the probability at least two packets are generated during τ_i.

The states of the transmitter can be broadly classified as *control cycle* and *data cycle* states. The states are differentiated based on the number of backlogged packets in the transmitter queue.

Control cycle states: States S_1, S_2 and S_3 represent the transmitter during the control phase. The node has zero, one and two backlogged packets respectively on entering states S_1, S_2 and S_3. The sojourn time of the all three states is M, the control cycle length.

When a packet is generated during S_1, the node might miss its turn in the current control cycle or it may transmit in the current cycle. A rotating order of reservation slot assignment is assumed. Let ξ represent the probability the node does not miss its turn in the control cycle. It is derived as follows. The conditional probability of packet arriving in slot i given that at least one packet occurs in M slots is given by $\frac{(1-\beta)^i\beta}{(1-\eta_1)}$ where $0 \leq i \leq M-1$. This represents the fact that a packet is generated in the slot i but not in any preceding slot. If a node arrives in slot i, the probability that any of the following $M-i$ slots is its turn in the cycle is $\frac{M-i}{M}$. This is used to derive the following equation:

$$\xi = \frac{\beta}{M(1-\eta_1)} \sum_{i=0}^{M-1} (1-\beta)^i (M-i) \tag{5.4}$$

Let σ denote the probability that a node will have a reservation to transmit in the current control cycle. This packet may be backlogged or generated during the control cycle. The probability that no *other* node in the system has sent a reservation in the current control cycle is denoted by $\mu = (1-\sigma)^{M-1}$.

The transitions from state S_1 are:

1. The node remains in state S_1 if (i) generated exactly one class A packet which was sent in the same cycle and no other node sent a reservation, or (ii) the node did not generate any packet and no other transmitted a reservation in the current cycle, so $p[1,1] = \mu(\eta_1 + \theta_1\xi(1-\gamma)$.

2. The node moves to state S_2 if (i) the node generated two packets where the first is a class A packet and was sent in the same control cycle, and (ii) no other node has sent a reservation in the control cycle. The probability is $p[1,2] = \rho_1\xi(1-\gamma)\mu$.

3. The node moves to state S_3 if (i) the node generated two packets and the node missed its turn in the control cycle, and (ii) no other node has sent a reservation in the control cycle. The probability is $p[1,3] = \rho_1(1-\xi)\mu$.

4. The node moves to state S_4 if (i) at least one other node has sent a reservation in the

control cycle, and (ii) the node either generates no packet or generates one or more packets where the first packet is class A and sent in the same cycle. The probability is $p[1,4] = (1-\mu)[\theta_1\xi(1-\gamma)+\eta_1]$.

5. States S_5 and S_6 represent the data cycle phase where the node generated at least one packet, missed its turn in the preceding control phase and some other node has sent a reservation in the control cycle. The node moves to S_5 or S_6 depending upon the number of packets generated where $p[1,5] = \theta_1(1-\xi)(1-\mu)$ and $p[1,6] = \rho_1(1-\xi)(1-\mu)$.

6. The node moves to state S_7 if exactly one class B packet was generated and its reservation was sent in the same cycle so $p[1,7] = \theta_1\xi\gamma$.

7. The transmitter moves from S_1 to S_8 if at least two packets were generated and the first packet was class B packet whose reservation was sent in the same cycle so $p[1,8] = \rho_1\xi\gamma$.

The transitions from state S_2 are:

1. The process moves to state S_1 if the current packet was class A, no new packet was generated and no other node has sent a reservation during τ_2, so $p[2,1] = \eta_2(1-\gamma)\mu$.

2. Node moves to state S_2 if current packet was class A, at least one new packet was generated and at least one node placed a reservation, $p[2,2] = (1-\eta_2)(1-\gamma)\mu$.

3. Node moves to state S_4 for block transmission if current packet was class A, no packet was generated, and at least one node placed a reservation, so $p[2,4] = \eta_2(1-\gamma)(1-\mu)$.

4. Node moves to state S_5 for block transmission if current packet was class A, at least one packet was generated and at least one node placed a reservation, $p[2,5] = (1-\eta_2)(1-\gamma)(1-\mu)$.

5. Node moves to state S_7 for block transmission if current packet was class B and no packet was generated, $p[2,7] = \eta_2\gamma$.

6. Node moves to state S_8 for block transmission if current packet was class B and at least one packet was generated, $p[2,8] = (1-\eta_2)\gamma$.

The transitions from state S_3 are:

1. The node moves to state S_2 if the current packet was class A and no other node placed a reservation during control cycle, so $p[3,2] = (1-\gamma)\mu$.

2. The node moves to state S_5 if current packet was class A and at least one other node placed a reservation during control cycle, so $p[3,5] = (1-\gamma)(1-\mu)$.

3. The node moves to state S_8 if current packet was class B with $p[3,8] = \gamma$.

Data cycle states: The remaining states $S_i, i \in \{4,\ldots,10\}$ represent the transmitter during the data cycle. States S_4, S_5, S_6 represent the transmitter being idle in the current data cycle. The sojourn time of these states is C_L, the average data cycle length given by Eqn. 5.7. This represents the node idle in the current data cycle and is entered if it has no backlogged packet. The process moves from state S_4 to state S_1 if no new packet was generated and $p[4,1] = \eta_4$; to state S_2 if one new packet is generated so $p[4,2] = \theta_4$; and to state S_3 if more than one packet is generated where $p[4,3] = \rho_4$. The process moves from state S_5 to state S_2 with probability $p[5,2] = \eta_5$ and to state S_3 with probability $p[5,3] = 1-\eta_5$. The node always moves to state S_3 from S_6 so $p[6,3] = 1$.

States S_7 and S_8 represent the data transmission phase and the sojourn times are given by $\tau_7 = \tau_8 = D$. The expression for average delay D is given by Eqn. 5.8. The node moves from state S_7 to state S_9 where $p[7,9] = \eta_7$. The node moves from state S_7 to state S_{10} if at least one packet was generated so $p[7, 10] = 1 - \eta_7$. The node always moves from state S_8 to state S_{10}, so $p[8, 10] = 1$.

The sojourn time of states S_9 and S_{10} is the residual wait in data cycle and $\tau_9 = \tau_{10} = C_L - D$. The node moves from state S_9 to state S_1 if no packet was generated and $p[9, 1] = \eta_9$. It moves to state S_2 if exactly one packet was generated and $p[9, 2] = \theta_9$. If more than one packet was generated it moves to state S_3 with $p[9, 3] = \rho_9$. The node moves from state S_{10} to state S_2 if no packet was generated so $p[10, 2] = \eta_{10}$, and to state S_3 otherwise so $p[10, 3] = 1 - \eta_{10}$. The transition probabilities are summarized in Table 5.2.

The transition probabilities are used to obtain the limiting probabilities of the embedded Markov chain from which the steady state probabilities of the semi-Markov process are derived.

5.4.2 Steady state probabilities

The limiting probabilities of the embedded Markov chain are obtained using the following equation:

$$\mathbf{V} = \mathbf{VP} \tag{5.5}$$

where **V** represents the limiting probability vector and **P** the transition probability matrix. The steady state probabilities of the semi-Markov process are obtained using the equation [96]:

$$P_i = \frac{V_i \tau_i}{\sum_{j=1}^{10} V_j \tau_j} \tag{5.6}$$

The transition probabilities and sojourn times of the states are dependent on the expected data cycle length (C_L) and expected delay within a cycle (D). The expected cycle length is dependent on the number of nodes that are active in a cycle. Given M nodes, C channels and σ the probability that a node has a packet to transmit in a cycle, the expected

M:	number of nodes
C:	number of data channels
L:	ratio of length of data packet to length of control packet
r:	number of levels in hierarchy
γ:	fraction of packets that are memory blocks
p_i:	fraction of traffic generated for level i
S_i:	denotes State i, $1 \leq i \leq 10$
τ_i:	sojourn time in state S_i
V_i:	limiting probability of being in state S_i in the embedded Markov chain
P_i:	limiting probability of being in state S_i in the semi-Markov process
β:	probability of at least one packet being generated in a control slot
λ:	Poisson packet generation rate at each node
T:	average packet delay
Γ:	channel utilization

Table 5.1: Notation used in analysis of FatMAC.

Control Cycle			Data Cycle		
Transition		Probability	Transition		Probability
$p[1,1]$	=	$\mu[\eta_1 + \theta_1\xi(1-\gamma)]$	$p[4,1]$	=	η_4
$p[1,2]$	=	$\mu[\theta_1(1-\xi) + \rho_1\xi(1-\gamma)]$	$p[4,2]$	=	θ_4
$p[1,3]$	=	$\mu\rho_1(1-\xi)$	$p[4,3]$	=	ρ_4
$p[1,4]$	=	$(1-\mu)[\theta_1\xi(1-\gamma) + \eta_1]$	$p[5,2]$	=	η_5
$p[1,5]$	=	$(1-\mu)[\theta_1(1-\xi) + \rho_1\xi(1-\gamma)]$	$p[5,3]$	=	$1-\eta_5$
$p[1,6]$	=	$(1-\mu)\rho_1(1-\xi)$	$p[6,3]$	=	1
$p[1,7]$	=	$\theta_1\xi\gamma$	$p[7,9]$	=	η_7
$p[1,8]$	=	$\rho_1\xi\gamma$	$p[7,10]$	=	$1-\eta_7$
$p[2,1]$	=	$\mu\eta_2(1-\gamma)$	$p[8,10]$	=	1
$p[2,2]$	=	$\mu(1-\eta_2)(1-\gamma)$	$p[9,1]$	=	η_9
$p[2,4]$	=	$(1-\mu)\eta_2(1-\gamma)$	$p[9,2]$	=	θ_9
$p[2,5]$	=	$(1-\mu)(1-\eta_2)(1-\gamma)$	$p[9,3]$	=	ρ_9
$p[2,7]$	=	$\eta_2\gamma$	$p[10,2]$	=	η_{10}
$p[2,8]$	=	$(1-\eta_2)\gamma$	$p[10,3]$	=	$1-\eta_{10}$
$p[3,2]$	=	$\mu(1-\gamma)\mu$			
$p[3,5]$	=	$(1-\mu)(1-\gamma)$			
$p[3,8]$	=	γ			

Table 5.2: State transition probabilities.

cycle length (C_L) and average delay per packet within a cycle (D) is derived from hashing theory [115–117]. Let k be the number of reservations for packets of length L in a cycle with C channels. Each packet is uniformly destined to any of the channels. Define $\vartheta = k/C$. The expected data cycle length is derived in [117] and given by:

$$\begin{aligned} C_l(k) &= L\sum_{b=0}^{k}\left[1-\left(e^{-\vartheta}\sum_{n=0}^{b}\frac{\vartheta^n}{n!}\right)^C\right] \\ C_L &= \frac{1}{1-(1-\sigma)^M}\sum_{k=1}^{M}\begin{pmatrix} M \\ k \end{pmatrix}\sigma^k(1-\sigma)^{M-k}C_l(k) \end{aligned} \tag{5.7}$$

and the average delay per packet within a cycle after transmitting the control packet is given by

$$\begin{aligned} D_l(k) &= L\left[1+\frac{k-1}{2C}\right] \\ D &= \frac{1}{1-(1-\sigma)^M}\sum_{k=1}^{M}\begin{pmatrix} M \\ k \end{pmatrix}\sigma_k(1-\sigma)^{M-k}D_l(k) \end{aligned} \tag{5.8}$$

The expected cycle length and packet delay are derived from the semi-Markov transition probabilities which are dependent on C_L and D. This leads to an iterative solution with the following steps:

1. Choose an initial value of σ the probability of a node reserving a memory block in a cycle $(0 < \sigma \leq 1)$.

2. Estimate C_L and D.

3. Compute the transition probabilities and recompute σ, C_L and D using expressions

derived in Section 5.4.3. The value of σ is computed using the expression

$$\sigma = \gamma \frac{P_1(1-\eta_1)\xi + P_2 + P_3}{P_1 + P_2 + P_3}$$

4. Repeat until the value of σ converges.

The following section derives the average packet delay and network throughput based on the transmission probabilities in this section.

5.4.3 Performance metrics

The performance metrics of interest are network channel utilization and average packet delay. Network channel utilization (Γ) is defined as the average number of slots utilized across all channels per control slot. Average packet delay (T) is the time between packet generation at the source and packet reception at the destination.

Channel utilization: Channel utilization (or throughput) within a cycle is determined by the number of nodes active in the cycle. The probability of leaving a state is given by $\Psi_i = P_i/\tau_i$ [96]. A class A packet is sent during states S_1, S_2 and S_3. A class B packet is sent during states S_7 and states S_8. The class A and class B throughput, and overall utilization are given by:

$$\Gamma_n^A = (1-\gamma)[\Psi_1(1-\eta_1)\xi + \Psi_2 + \Psi_3] \quad (5.9)$$

$$\Gamma_n^B = \Psi_7 + \Psi_8 \quad (5.10)$$

$$\Gamma = M[\Gamma_n^A + L\Gamma_n^B] \quad (5.11)$$

Average packet delay: Packet delay is defined as the time between packet generation and packet reception. The average packet delay (T) can be obtained by listing the possible paths a packet can traverse using Little's law.

$$
\begin{aligned}
T^A &= \frac{1}{\Gamma_n^A}\sum_{i=1}^{10} E^A[N_i]P_i && (5.12)\\
T^B &= \frac{L}{\Gamma_n^B}\sum_{i=1}^{10} E^B[N_i]P_i && (5.13)\\
T &= \gamma T^B + (1-\gamma)T^A && (5.14)
\end{aligned}
$$

where $E[N_i]$ denotes the expected number of packets queued per node in state S_i and Γ_n denotes per node throughput. The superscripts indicate the packet type. The values of $E[N_i]$ are obtained from the model. Table 5.3 summarizes these values. $\mathcal{F}$ gives the average number of packets in the queue when the node is resident in state S_j and is dependent on τ_j. I represents the buffer size on entering the state and B the maximum buffer size:

$$\mathcal{F}[I, B, \lambda, \tau] = \frac{1}{\tau}\sum_{k=0}^{\tau} \min[I + \lambda k, B]$$

5.4.4 Model Validation

This section validates the analytic model through discrete-event simulation. Simulation results have been obtained using the stochastic self-driven discrete-event models, written in C with YACSIM [98]. YACSIM is a C based library of routines that provides discrete-event and random variate facilities. Steady state transaction times and throughput were measured. Simulation convergence was obtained through the replication/deletion method [97], with a 95% confidence in a less than 5% variation from the mean.

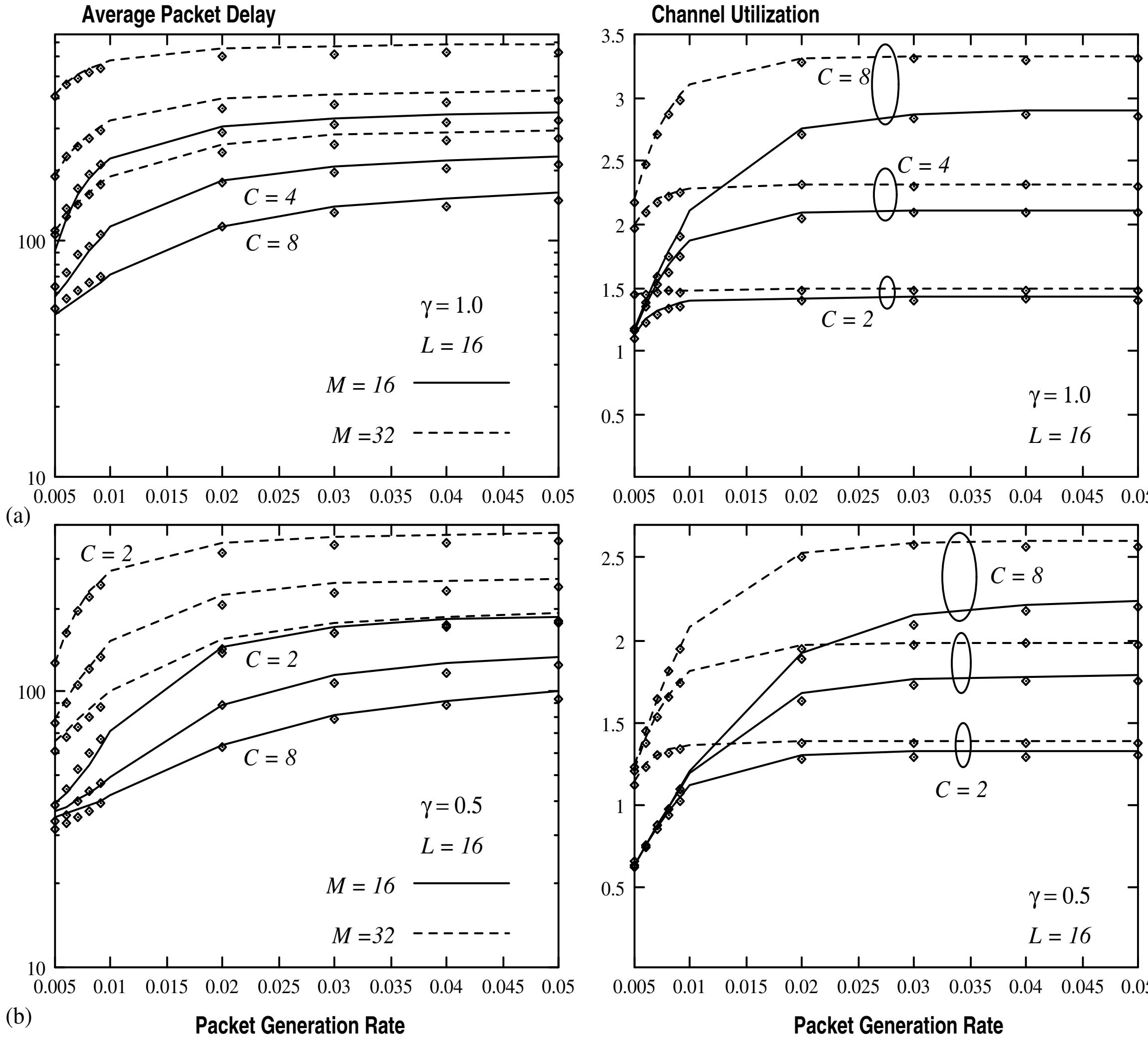

Figure 5.7: Validation of analytic model for FatMAC protocol through simulation for $M \in \{16, 32\}$, $C \in \{2, 4, 8\}$, $\gamma \in \{0.5, 1.0\}$ and $L = 16$. Points and lines represent simulation and analysis respectively.

State	$E^A[N_i]$	$E^B[N_i]$
1	$(1-\gamma)\mathcal{F}[0,2,\lambda,\tau_1]$	$\gamma\mathcal{F}[0,2,\lambda,\tau_1]$
2	$(1-\gamma)\mathcal{F}[1,2,\lambda,\tau_2]$	$\gamma\mathcal{F}[1,2,\lambda,\tau_2]$
3	$2(1-\gamma)$	2γ
4	$(1-\gamma)\mathcal{F}[0,2,\lambda,\tau_4]$	$\gamma\mathcal{F}[0,2,\lambda,\tau_4]$
5	$(1-\gamma)\mathcal{F}[1,2,\lambda,\tau_5]$	$\gamma\mathcal{F}[1,2,\lambda,\tau_5]$
6	$2(1-\gamma)$	2γ
7	$(1-\gamma)(\mathcal{F}[1,2,\lambda,\tau_7]-1)$	$1+\gamma(\mathcal{F}[1,2,\lambda,\tau_7]-1)$
8	$(1-\gamma)$	$1+\gamma$
9	$(1-\gamma)\mathcal{F}[0,2,\lambda,\tau_9]$	$\gamma\mathcal{F}[0,2,\lambda,\tau_9]$
10	$(1-\gamma)\mathcal{F}[1,2,\lambda,\tau_{10}]$	$\gamma\mathcal{F}[1,2,\lambda,\tau_{10}]$

Table 5.3: Expected number of packets in each state used in Little's law to derive average packet delay.

Fig. 5.7 compares the performance predicted by the analytic and simulation models for $M \in \{16, 32\}$, $C \in \{2, 4, 8\}$, $\gamma \in \{0.5, 1.0\}$ and $L = 16$. The deviation of network channel utilization predicted by the analytic model was less than 5% and the deviation of average packet delay was less than 8%. Fig. 5.7(a) shows that the performance of the protocol improves with increasing C. This is due to the reduced traffic per channel which decreases the expected cycle length reducing latency and increasing utilization. The increase in maximum utilization is 100% when C is increased from 2 to 8 for $M = 16$. The increase in utilization with increasing C will be higher for a larger M/C ratio . Fig. 5.7(b) shows that the validity of the model for a system with $\gamma = 0.5$. The graph plots the overall traffic delay and channel utilization for both control and block packets. The model also enables prediction of control packet and data packet performance separately as explained in Section 5.4.3.

The following section compares the performance of FatMAC and I-TDMA for varying system parameters.

5.5 Performance Analysis

This section compares the performance of FatMAC and I-TDMA for single-level and multi-level systems. The performance of I-TDMA has been studied through simulation and analytic models in [18]. The system parameters of interest are C, M, and the traffic generation rate. Other parameters include the number of levels (r), the fraction of class B packets (γ) and traffic fraction requiring level-i communication (p_i) where $1 \leq i \leq r$.

The ratio γ has a significant impact on system performance and in identifying the performance advantages of either protocol. A simulation study of snooping-based and directory-based cache coherence scheme was used to estimate γ. The *FFT* and *SPEECH* multiprocessor traces [118] were used with traces from the SPLASH benchmark suite [119] to estimate γ.

Snooping-based coherence schemes, which generally require a broadcast facility for requests and invalidations, used with the *mp3d* and *water* SPLASH benchmarks had a value of $\gamma \approx 0.3$. Directory-based coherence schemes, studied with the FFT and SPEECH traces, usually use explicit invalidations and the measured value of γ ranged between 0.06 and 0.03, depending on the amount of sharing. This shows that traffic is mostly composed of control packets in the studied applications even with a snooping-based coherence strategy. This led to the primary motivation for the development of FatMAC since a low value of γ results in throughput loss and high latency for I-TDMA. In the study below, we examine protocol performance for $\gamma \in \{0.1, 0.5, 0.9\}$.

5.5.1 Single-Level system performance

This section compares the performance of FatMAC and I-TDMA for a single level system. The reduction in latency and loss of maximum throughput of FatMAC are quantitatively

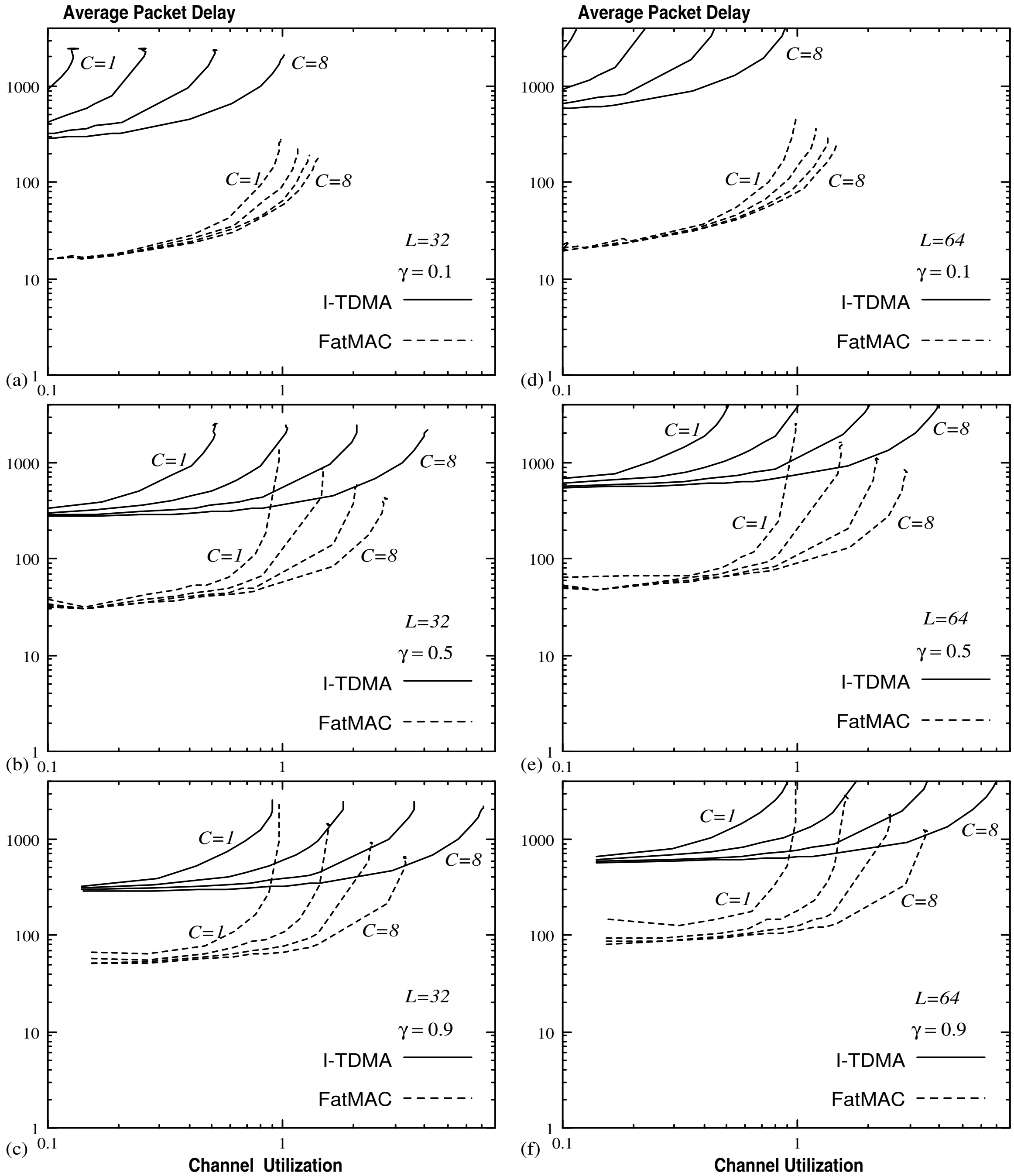

Figure 5.8: Comparison of single-level protocol performance for $M = 16$ and $C \in \{1, 2, 4, 8\}$: (a) - (c) $L = 32$: (a) $\gamma = 0.1$, (b) $\gamma = 0.5$, (c) $\gamma = 0.9$; (d) - (f) $L = 64$: (d) $\gamma = 0.1$, (e) $\gamma = 0.5$, (f) $\gamma = 0.9$.

analyzed in this section.

Variations in C

Fig. 5.8 compares the performance of single level I-TDMA and FatMAC for $M = 16$, $L \in \{32, 64\}$, $\gamma \in \{0.1, 0.5, 0.9\}$ and $C \in \{1, 2, 4, 8\}$. It is observed for all cases that FatMAC has lower latency under light loads whereas I-TDMA has higher maximum utilization under extreme loads. For example, the delay offered by FatMAC for $M = 16$, $C = 4$ and $\gamma = 0.1$ at utilization of 0.2 is 3% of that offered by I-TDMA. This represents a significant reduction in delay mainly due to the large cycle length of I-TDMA and the fact that most traffic is control packets. The reduction in delay for a similar system at throughput of 1.5 is 80% for $\gamma = 0.9$. In general, the latency advantage of FatMAC increases with increasing system size, packet size, and decreasing γ. When the fraction of Class B packets increases, the expected cycle length of FatMAC increases reducing the magnitude of latency reduction. For low γ, the utilization of FatMAC is higher than I-TDMA for systems with small M/C ratios. Under heavy loads and for large γ the maximum utilization of I-TDMA is higher.

The maximum throughput attained by FatMAC is less than that of I-TDMA since a node can transmit only one packet per data cycle in FatMAC compared to C packets per cycle in I-TDMA. The cycle lengths for FatMAC is approximately $M + ML/C$. This assumes that the packets generated by the M nodes are distributed evenly over C channels. The cycle length of I-TDMA is given by ML. Comparing the two cycle lengths yields:

$$\mathcal{L}^{Fat} = \left[\frac{1}{L} + \frac{1}{C}\right] \mathcal{L}^{Td} \tag{5.15}$$

For $L \gg C$, which is the case of our target environment, the second term in the above summation will be dominant. The cycle length of FatMAC will be $1/C$ times that of

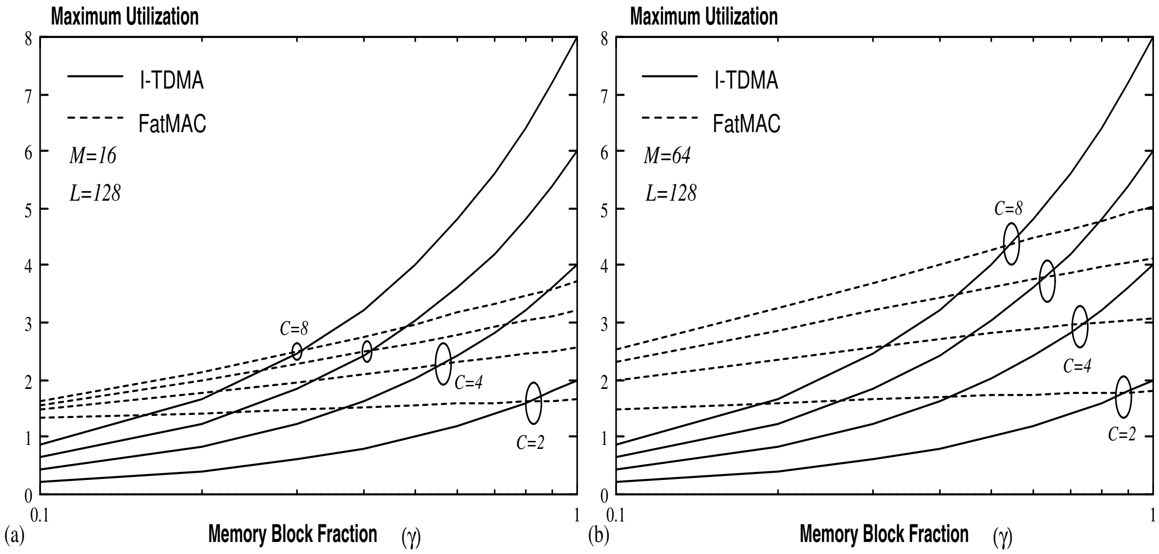

Figure 5.9: Comparison of maximum throughput for I-TDMA and FatMAC for $L = 128$ with varying γ and C: (a) $M = 16$, and (b) $M = 64$.

I-TDMA. The maximum utilization in this case will be roughly the same for both I-TDMA and FatMAC.

Fig. 5.9 studies the maximum utilization achieved by the two protocols for $M \in \{16, 64\}$, $L = 128$, and varying γ and C. The graphs show that the maximum utilization of FatMAC can be greater than that of I-TDMA depending on γ. Fig. 5.9(a) shows that FatMAC can achieve higher maximum throughput than I-TDMA for $\gamma \leq 0.8$ for $C = 2$ and $M = 16$. The cutoff value of γ where the throughput of I-TDMA exceeds that of FatMAC decreases with increasing the number of channels. The performance advantage of FatMAC improves with increasing M due to the longer cycle lengths of I-TDMA.

As the ratio L increases, the throughput advantage of I-TDMA decreases. This is due to the longer data cycles which reduces the relative overhead due to the control cycle. Maximum utilization Γ_{max} with C channels for FatMAC and I-TDMA is given as:

$$\Gamma_{max}^{Fat} = C\left[\frac{1+(L-1)\gamma}{C+L}\right] \tag{5.16}$$

$$\Gamma_{max}^{Td} = C\left[\frac{1+(L-1)\gamma}{L}\right] \tag{5.17}$$

$$\Gamma_{max}^{Td} = \left[1+\frac{C}{L}\right]\Gamma_{max}^{Fat} \tag{5.18}$$

The equation shows that the throughput advantage of I-TDMA under heavy loads is independent of M and γ. However, the advantage decreases with increasing data packet size. The graphs also show that the performance improvement in terms of maximum throughput with increasing C is higher for I-TDMA. The throughput advantage of I-TDMA is higher as C increases. The system under consideration contains a small number of channels and in the case where $L \gg C$ which is the case of our target environment, $\Gamma_{max}^{Fat} \to C$.

Though I-TDMA can theoretically achieve the maximum utilization of C, it depends on the rate at which the system interface can generate outgoing packets. The network interface has the capability to send consecutive packets on the channels up to C packets in a I-TDMA cycle. The system interface may not be able to sustain such high packet rates in a realistic situation. Consider a system with 16 nodes, 4 channels and transmission speed of 2 Gbit/s. The cycle length with 8 Kbyte packets is $525\mu s$. If the system interface is able to generate 7500 memory blocks per second that is uniformly targeted to all the 16 nodes, the maximum throughput can be achieved. The protocol is thus limited by the maximum traffic the system interface can generate.

The above paragraphs quantified the reduction in latency achieved by FatMAC particularly under light loads at the expense of bandwidth. In a high speed environment, bandwidth tradeoff for lower delay can be reasonably justified. The following section examines the protocols with respect to scalability in system size.

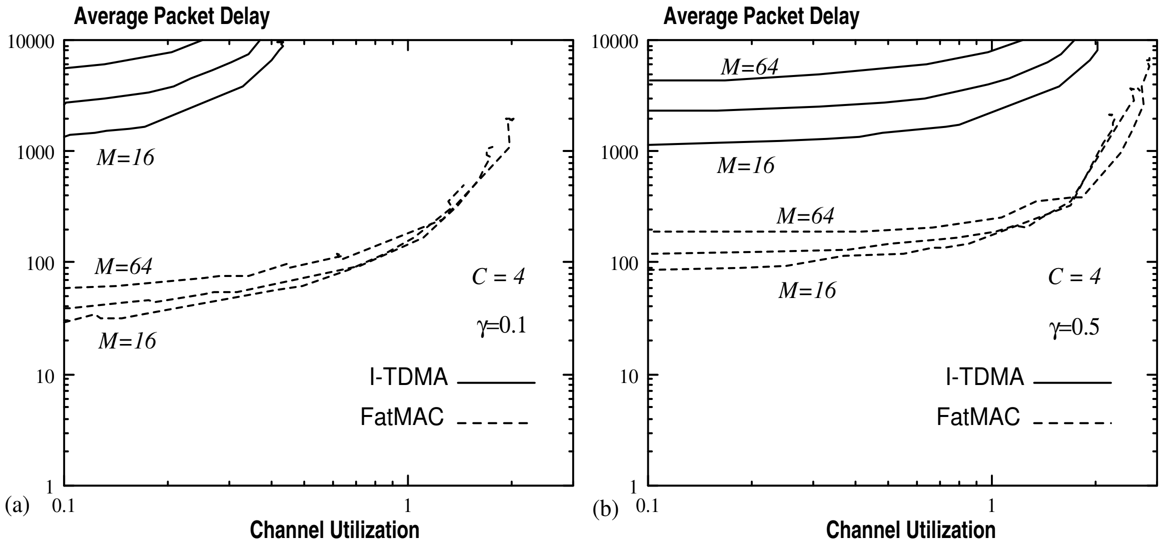

Figure 5.10: Comparison of single-level protocol performance for $M \in \{16, 32, 64\}$, $C = 4$ and $L = 128$: (a) $\gamma = 0.1$, and (b) $\gamma = 0.5$.

Variations in M

Fig. 5.10 compares the performance of the two protocols with respect to system size scalability. The graph shows the performance for $M \in \{16, 32, 64\}$, $C = 4$, $\gamma \in \{0.1, 0.5\}$ and $L = 128$. Increasing M increases the average latency for both protocols since the cycle length is proportional to M. Under light loads, the increase in cycle length is proportional to memory packet and control packet length for I-TDMA and FatMAC respectively. This contributes to the lower latency of FatMAC. For small values of γ, the delay reduction gained with FatMAC is higher since most of the slots in I-TDMA are under-utilized. The increase in delay under light loads when M is increased from 16 to 64 is 200% and 300% for FatMAC and I-TDMA respectively. However, the delay obtained with FatMAC for $M = 64$, $C = 8$ and $\gamma = 0.1$ is less than that of I-TDMA. For higher values of γ, there is little variation in the delay of I-TDMA since the cycle length is independent of γ. The cycle length increases with γ for FatMAC and hence delay is higher. The reduction in latency holds for higher values of γ but the magnitude is diminished as seen from Figs. 5.10(a)-(b).

The following section compares the performance of the protocols for two level systems.

5.5.2 Two-Level system performance

Fig. 5.11 compares protocol performance for a 2 level system with $M = 4\times4$, $L \in \{32, 64\}$, and $\gamma \in \{0.1, 0.5, 0.9\}$. The number of channels is varied between $\{(1,3),(2,2),(3,1)\}$. The probability that a packet is destined to the local cluster (p_1) is 0.9 and the global cluster (p_2) is 0.1. The lower latency offered by FatMAC under light loads is observed in all cases. The reduction in latency for $C = 2 \times 2$, $M = 4 \times 4$, $L = 64$, and $\gamma = 0.1$ is 94% at system utilization of 0.2. The magnitude of latency reduction decreases with increasing γ as in the single level case. For small values of γ, the synchronization delay of I-TDMA is of the order of ML whereas the latency of FatMAC is of the order of M; as γ increases, the delay of FatMAC increases due to longer data cycles and hence the performance advantage decreases. It is also observed that maximum throughput is higher for FatMAC at lower γ and higher for I-TDMA at higher γ. This again illustrates the tradeoff of bandwidth to minimize latency. The level 2 traffic among nodes is quite low and does not benefit from an increase in channels beyond 2 at that level. The number of channels allocated to each level can be varied in accordance with p_i to maintain uniform traffic intensity across levels.

Increasing the number of channels for level-1 communication improves performance depending on the value of γ. Increasing C from 2 to 3 for level-1 communication among 4 nodes does not reduce latency significantly particularly for $\gamma = 0.1$. The number of class B packets is small and there is little reduction in average cycle length by increasing C. For low values of γ, there is minimal improvement in increasing the number of channels for the first level since most packets are control packets and data cycle lengths are small.

Fig. 5.12 compares the performance of the protocols with reduced locality for $p_1 = 0.7$

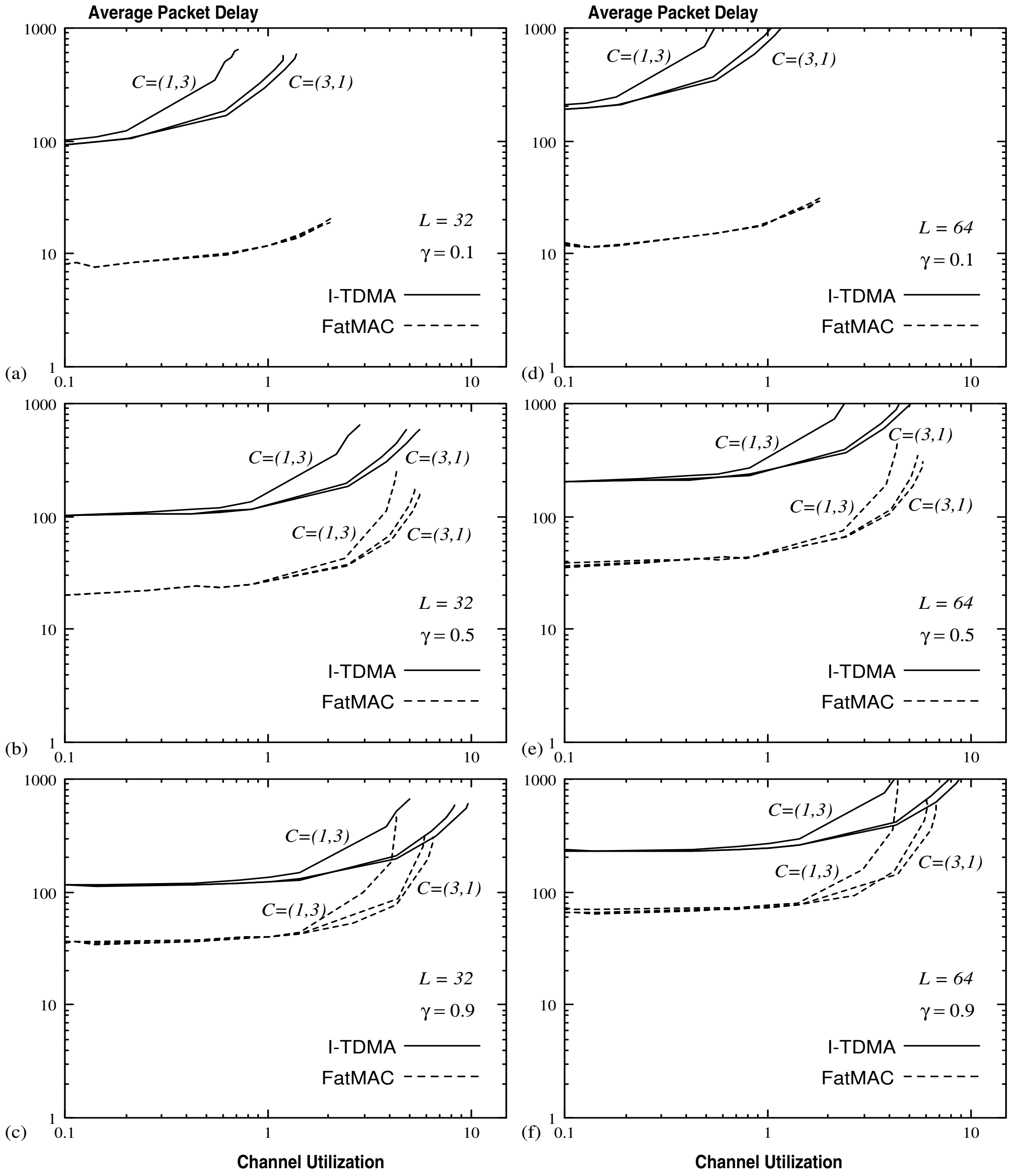

Figure 5.11: Comparison of two-level protocol performance for $p_1 = 0.9$ and $p_2 = 0.1$. (a)–(c) $M = 4 \times 4$, $L = 32$, $C \in \{(1,3),(2,2),(3,1)\}$, (a) $\gamma = 0.1$, (b) $\gamma = 0.5$, (c) $\gamma = 0.9$; (d)–(f) $M = 4 \times 4$, $L = 64$, $C \in \{(1,3),(2,2),(3,1)\}$, (d) $\gamma = 0.1$, (e) $\gamma = 0.5$, (f) $\gamma = 0.9$.

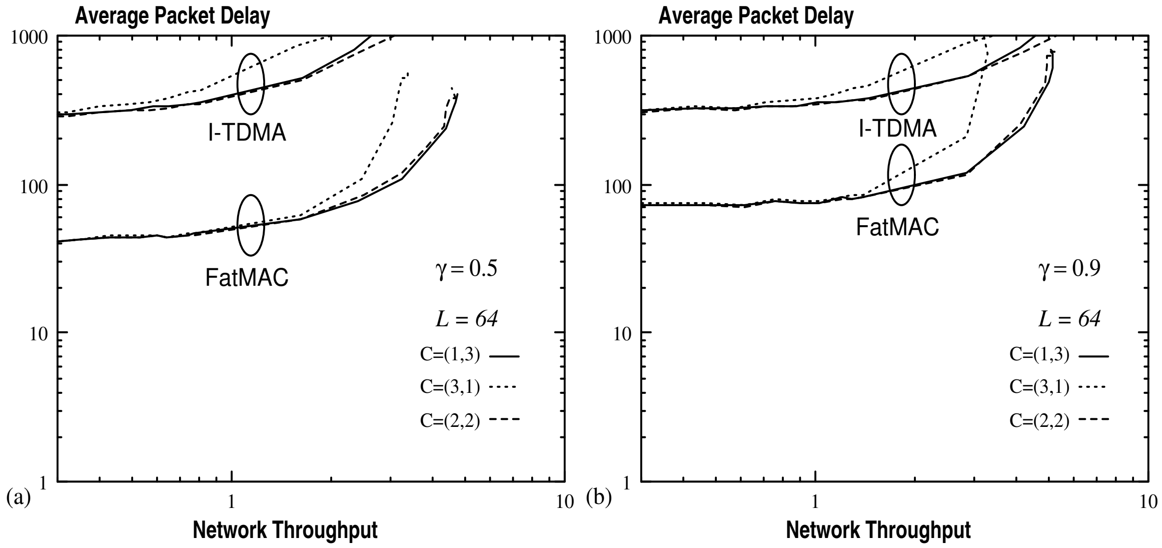

Figure 5.12: Comparison of two-level protocol performance for $M = 4 \times 4$, $L = 64$, $C \in \{(1,3),(2,2),(3,1)\}$, $p_1 = 0.7$ and $p_2 = 0.3$: (a) $\gamma = 0.5$, (b) $\gamma = 0.9$.

and $p_1 = 0.3$. The figure is plotted for a system with $M = 4 \times 4$, $L = 64$, $C = 4$, $\gamma \in \{0.5, 0.9\}$ and varying channel allocation to levels. The results for $\gamma = 0.1$ were identical to that shown for $p_1 = 0.9$. There was little variation in performance with varying channel allocation for $\gamma = 0.1$. The general trend of low latency with FatMAC is observed in all the cases. When the number of channels used in the first level (C_1) is increased from 1 to 2, there is improvement in performance for both protocols. When C_1 is increased from to 3, the performance degrades. This is due to the fact that only one channel is available for level 2 traffic creating a bottleneck. The maximum throughput decreases for the case of $C_1 = 3$ and $C_1 = 2$ for both protocols.

For a multi-level system, the node does not transmit a control packet on another cluster until it is finished with transmitting the current packet. This can lead to reduced utilization since a node misses its allocated control slot on a different level. The protocol can be modified to detect such a situation and efficiently utilize the bandwidth but results in added complexity and extra processing latency. The interaction of traffic between the two levels will have impact on system performance. This is due to contention between the different levels for the single transmitter at the node. An expensive solution to overcome this contention will be to provide a transmitter per level.

The next section examines the use of Slotted Aloha on the control cycle to implement packet scheduling.

5.5.3 Slotted Aloha for control cycle

The definition for FatMAC has been based on using TDM for the reservation cycle to provide a collision free protocol. The disadvantage of TDM based systems is the long synchronization delay especially with large number of nodes. The objective of this section is to study latency reduction through alternate strategies for the reservation cycle. Slotted Aloha (SA) is the simplest random access scheme that provides low latency under light loads. SA based reservation access schemes have been widely studied for satellite communication systems [80]. The following section presents the various approaches using SA for satellite communication. This is followed by examining the applicability of these approaches to the WDM environment. An alternate approach using TDM scheduling based on scheduling theory has been studied in [120, 121].

SA based satellite protocols

The SA based protocols studied earlier assume a single broadcast channel. Some of the protocols use separate subchannels for data and control transmission. The different multiple access mechanisms for broadcast channels in general have been compared and analyzed in [78–80]. The protocols of particular interest are reservation based approaches which use SA.

Transmission in satellite frames is usually organized in frames. A frame consists of a fixed number of packet slots and each frame has enough slots so that the frame length exceeds the round trip propagation delay [122]. When a source is able to successfully transmit in a slot, it automatically reserves access on corresponding slots in the next frame. A node may transmit and end-of-transmission in its last packet allowing other nodes to contend for the slot in the next frame. A variation of this scheme is to allow each source own a data slot within the frame [123]. When a source is not using its slot, other nodes can capture it by contention. The owner node can reclaim a captured slot by sending a packet in its allocated slot. This approach is stable and provides fair allocation. In each data slot, a node can send both a data packet and a piggybacked reservation in the data header.

Reservation access with a separate channel for reservations based on Slotted Aloha was proposed in [124]. This protocol was modified to include a fixed-frame structure and has been studied in [125, 126]. The frame consists of a reservation subframe and a data subframe each consisting of more than one slot. Contention based priority oriented demand assignment (CPODA) [113] uses SA for the reservation subframe of each data frame. Each fixed size frame consists of a centralized assignment subframe, a distributed assignment subframe and control subframe. Stability and fairness mechanisms have been added to the protocol. This is done by varying the size of the control subframe according to current scheduling requirements. If no reservations are outstanding, the control subframe occupies the entire frame. When the system is heavily scheduled, only higher priority traffic are allowed to compete in the contention subframe. Fixed PODA, CPODA and Fixed TDMA have been extensively studied in [127] with respect to the Atlantic Packet Satellite Experiment.

SRUC (Split Reservation upon collision) is an adaptive protocol which combines SA and TDMA Reservation [128]. Each frame consists of L large slots: each large slot consists of a data slot followed by V minislots. The protocol switches from SA to TDMA-Reservation based on the channel traffic. The data slot can be in either contention mode or reserved mode. It normally operates in contention mode – a node sends a packet using SA in the data slot and a reservation in one of the following minislots. In the event of a collision in the data slot, the system switches to reserved data slots.

The protocol proposed in [129] uses a fixed frame and a separate reservation channel. The data frame is based on reservations in the previous data frame. Unscheduled data frame slots in the current frame are used in a contention mode in an attempt to reduce delay. This protocol has been modified to include a separate narrowband reservation subchannel using SA in [130]. Two distributed reservation control protocols are studied in the paper for transmission of fixed length datagram packets.

A combination of contention and reservation based transmission is studied in [131]. Each frame is divided into reserved slots followed by Aloha slots. Each slot consists of a set of reservation minislots and one data subslot. A node places a reservation on the reservation minislot in contention mode. It also attempts transmission in the Aloha slots. If this is successful, the reservation is ignored. A modification of this protocol consists of Aloha slots followed by reserved slots protocol is studied in [132]. The protocol is studied for traffic composed of short and long messages. A short message is sent in an Aloha slot as are reservations for long messages. This model corresponds to the SA based protocol proposed in the next section. The main difference is the fixed frame structure of [132].

The satellite internetworking environment described in [133] combines TDM-based and SA-based access. The network is based on a satellite switch that has on-board processing to connect uplinks and downlinks. Each uplink frame consists of different types of slots: request, interactive data slots, and reserved slots for data services. Nodes with connection oriented traffic use the request slot using SA prior to data transmission. A successful request packet allocates slots in subsequent frames to the requesting node. Nodes with connectionless traffic use the interactive data slots using SA. The onboard processor examines the uplink frames and transmits the status of collided slots on the next downlink frame. This approach assumes an active processor to examine the status slots and is not suited to the passive star WDM environment.

The above paragraphs summarized earlier reservation approaches using SA and in particular for satellite systems. Propagation delay was a major concern in the design of the protocols due to the earth-to-satellite distances. Another common feature observed in satellite protocols is the concept of a fixed size frame containing data slots. In the high speed network under consideration, propagation delay is still a major concern even for the short network distances. Also, the concept of a fixed frame will not be used here. The remainder of this section presents a Slotted Aloha (SA) based reservation protocol for the WDM environment.

SA Reservation for WDM

In this section, the proposed approach for reservation access with SA is presented. Transmission is organized into cycles and each cycle consists of a reservation phase followed by a data phase. The access during reservation is simple: a node transmits its control packet at the beginning of a slot. The node listens for the successful transmission of its control packet. If it does not receive its packet successfully within twice propagation delay from node to partitioner, collision has occurred. If the packet is successfully transmitted, all nodes update the data cycle information. If the packet was lost due to collision, the node follows a backoff policy before attempting retransmission of the control packet.

The main issue to be resolved with this approach is the termination of the control cycle end and the start of the data cycle start. In a TDM system, the length of the control cycle is fixed as M slots. The idea of using SA is to reduce this control cycle length and allow a node to transmit its data packet as soon as the reservation is made. One possible solution is to fix the number of control slots in a cycle. This number can be broadcast at the start of a reservation cycle. The data cycle will be initiated at the end of the control phase after all nodes have processed the successful control packets in the cycle. A distributed clock is used to maintain slot synchronization and the clock packet can be used to broadcast control cycle length. This definition of the protocol uses the Data-Cycle-Stall (DCS) scheme but is also applicable to the In-flight-Reservation (IRS) scheme.

The following steps describe the operation of the Slotted Aloha based protocol:

1. At the start of the time, the clock node fixes the number of control slots in a control cycle. This will be usually a small fraction of the number of nodes. In the TDM approach, this is be M, the number of nodes. The clock node broadcasts the control cycle length in the payload of the clock packet.

2. Each control slot is accessed using Slotted Aloha. Successful control packets are used to update the data cycle.

3. The data cycle is initiated after the control cycle ends. The computation of the data cycle is identical to the TDM control cycle. The data cycle is collisionless as before.

At the end of each control cycle, the clock node computes the number of collisions during the control cycle. The length of the control cycle may be periodically updated based on the collision information. A thresholding algorithm may be used to update the length. For example, the length may be doubled if the probability exceeds a threshold and may be reduced by half if the probability falls below a threshold. The modified control cycle length is then broadcast to all other nodes through the clock packet.

The advantage of the dynamically varying control length approach is that it allows the system to be switched between Slotted Aloha and TDM on the control cycle. Such an approach has been studied for a non reservation protocol in [134]. The clock node switches over to TDM mode on the control cycle if the collision probabilities exceed a threshold. A similar threshold may be used to switch to SA mode if the traffic is very light and collision probability drops below a certain level.

The disadvantages of the SA based approach:

1. The node has to wait the round-trip delay before it can ascertain success of the control packet. This makes the protocol vulnerable to propagation delay.

2. A node may not be able to retransmit in the same control cycle. The node has to wait for at least until the end of the current data cycle before it can transmit. This results in increased latency.

In the following section, we compare the performance of the SA approach to the TDM approach. The magnitude of reduction in latency under light loads is mainly studied here.

Performance Analysis

In this section, we study the performance of the two reservation approaches with varying number of nodes, propagation delay and γ.

The delay characteristics of the two schemes under light traffic loads is considered here. Control packet size is taken to be 53 bytes, the size of an ATM cell. The length of the data packet is 8K bytes which corresponds to 172 cells. Time is normalized to cell transmission time at 200 Mbps. Each unit of time corresponds to approximately 2.12 μs. Maximum propagation delay from the nodes to the partitioner is varied. For example, a one-way delay of 1Km corresponds to roughly 2.4 cells at 200 Mbps. At 2.56 Gbps, the same distance corresponds to roughly 31 cells.

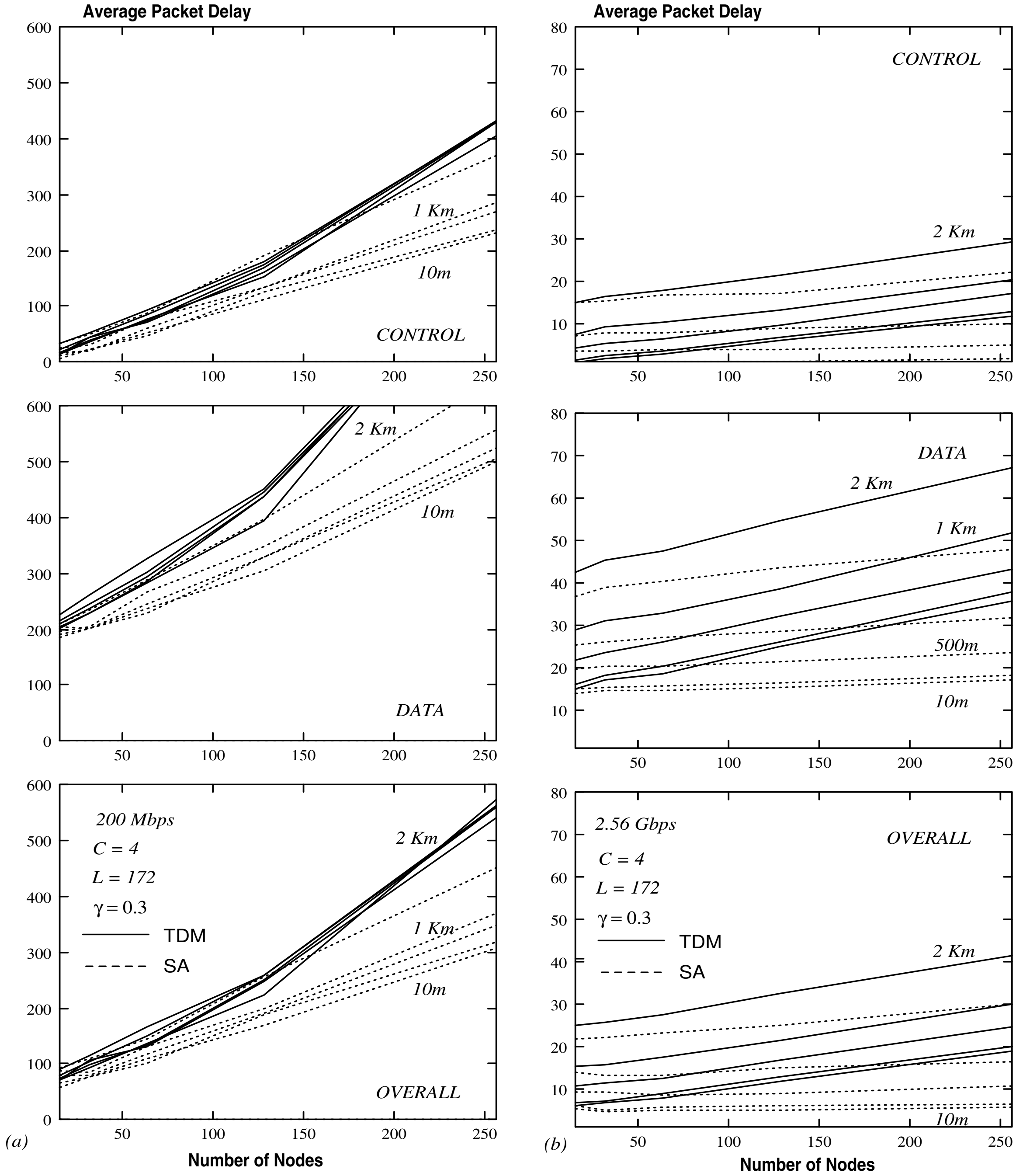

Figure 5.13: Comparison of performance with TDM and SA on control channel for $C = 4$, $L = 172$, $\gamma = 0.3$ at light loads for $M \in \{16, 32, 64, 128\}$. Propagation delay is varied as 10m, 100m, 500m, 1Km, 2Km: (a) Speed of 200 Mbps, and (b) Speed of 2.56 Gbps.

Fig. 5.13 studies the performance of SA and TDM strategies on the reservation channel for $C = 4$, $L = 172$, $\gamma = 0.3$ at light loads of 50 packets per second per node for $M \in \{16, 32, 64, 128\}$. Propagation delay is varied as 10m, 100m, 500m, 1Km and 2Km. Transmission speeds of 200 Mbps and 2.56 Gbps have been used in the simulation. For each case, the average packet delay of control and data packets and the overall average packet delay is shown in the graphs. A distributed clock mechanism is used for slot synchronization. The Data Cycle Stall (DCS) algorithm is used for frame synchronization.

Fig. 5.13(a) studies the performance at transmission speed of 200 Mbps. The reservation cycle length of SA at startup is taken to be $0.2M$. Nodes follow a geometric backoff strategy after collision with a probability of 0.05 of transmitting in the next control slot. For all system sizes and propagation delay, the delay characteristics of SA is seen to be better than that of TDM for control, data and overall traffic. The reduced control cycle length results in overall improved delay characteristics of the system. This is also due to the fact that the data cycle lengths are not long because of the low value of γ and the light traffic load. The decrease in latency is seen to improve with increasing M. The decrease in latency is seen to be 20% for $M = 128$, $C = 4$ and propagation delay of 1Km. An increase in M also increases the chances of collisions and hence increased latency. However, for the light loads studied, collisions were rare. Under heavy traffic, the data cycles tend to be longer and the system collapses due to collisions and long retransmission wait times.

Fig. 5.13(b) studies the performance at higher transmission speed of 2.56 Gbps. At this speed, the propagation delay impact is higher due to the high ratio of propagation delay to packet transmission time. Under light load conditions, SA provides lower latency for the propagation delay values studied. The decrease in latency improves with increasing number of nodes due to synchronization delay with TDM. The decrease in delay for $M = 128$, $C = 4$ and $L = 172$ at 2.56 Gbps is 80% for $\gamma = 0.3$ and node-to-partitioner propagation delay of 1Km. The improvement is seen to be higher at higher transmission speeds. This is due to the smaller data cycle transmission time at these speeds.

The study was extended the study the impact for higher loads of 750 packets per second. For propagation delay of up to a few hundreds of meters, the delay of SA is less than that of TDM. The delay of SA was comparable to that of TDM at around 500m one way propagation delay. For higher propagation delay, the time before a node knows about its control packet success is increased. In the worst case, a node may have to wait until the next control cycle before attempting retransmission.

In general, the packet latency decreases with increased transmission speed as seen from the graphs. However, the impact of increased propagation delay is higher at 2.56 Gbps. The increase in delay when propagation delay is increased from 10m to 10 Km is 10$\%$ for M = 64, C = 4 at 200 Mbps under light loads. The increase in delay at 2.56 Gbps for similar parameters is more than 100$\%$. This is mainly due to the nature of the Data-Cycle-Stall frame synchronization algorithm. The impact of propagation delay is higher with SA since the node has to ascertain packet success before sending the data packet.

Fig. 5.14 studies the performance of the two schemes for speed of 2.56 Gbps for varying M, $C = 4$, $L = 172$, $\gamma \in \{0.5, 0.9\}$ and varying propagation delay at a load of 50 packets per second per node. Under these light load conditions, the latency advantage of SA over TDM is maintained as with lower value of γ. For lower transmission speeds, the disadvantage of SA was the long retransmission wait for high value of γ. This latency is reduced at high speeds because the transmission time of the data cycle is considerably reduced. Transmitting an 8K data packet using 172 cells requires 365 μs at 200 Mbps and 29 μs at 2.56 Gbps. This reduces the impact of the data cycle length. However, the propagation delay impact will be greater at higher transmission speeds.

The above discussion outlined the performance comparison of using TDM versus SA

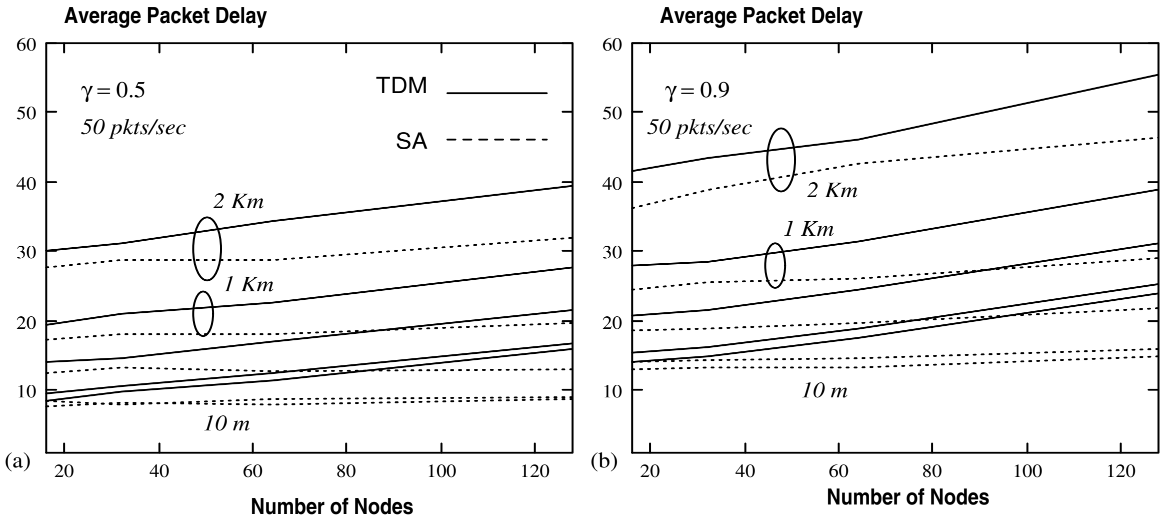

Figure 5.14: Comparison of SA and TDM on Control channel for varying M, $C = 4$, speed of 2.56 Gbps, and $L = 172$. Propagation delay is varied as 10m, 100m, 500m, 1Km and 2Km at a load 50 packets per second per node.

on the control channel. The proposed SA scheme with dynamic control cycle length has advantage when the fraction of memory block traffic is low and when the system loads are low. Under heavy loads, a dynamic capability to switch to TDM will eliminate instability. This does not address the long retransmission wait times due to the longer data cycles for high values of γ even under light loads. In this case, the nodes could switch to TDM mode by monitoring the average retransmission delays.

5.6 Synchronization

This section studies the synchronization requirements for protocol implementation with respect to bit, slot, and frame synchronization.

5.6.1 Bit Synchronization

The network is configured to operate in a multi access environment and a concern is bit synchronization delay at the receivers. In a point-to-point or virtual point-to-point network, a receiver and transmitter are locked by requiring the transmitter to send special synchronization frames when there are no data packets. In a multi-access approach, a receiver has to receive a packet from a different node following a packet from another node. The receiver is required to establish bit synchronization and clock acquisition with the new transmitter as rapidly as possible. Transmitting a 1000 bit packet at 200 Mbit/sec requires 5 μs. Receiver modules with sub-microsecond bit synchronization times are desirable for this environment.

The impact of bit synchronization on protocol design and performance is studied here. Let T_c denote the control packet length, T_d the memory block packet length, and T_s the bit synchronization time. A bit synchronization time of T_s is incurred each time the node receives a packet from a different transmitter. Let $L = T_d/T_c$ denote the ratio of memory block to control packet length, and $\theta = T_s/T_c$ denote the ratio of bit synchronization time to control packet length. Let γ represent the fraction of packets that are memory block packets. For the following discussion, time is normalized to control packet length. For control packet of size 1000 bits and transmission speed of 200 Mbps, each time unit represents $5\mu s$.

FatMAC

The control cycle length of FatMAC is given by $M(1 + \theta)$. There will be at most $M\gamma$ packets in the associated data cycle. The average data cycle length is given by $C_L(L + \theta)$ where C_L is given by Eqn. 5.7. Bit synchronization overhead is incurred twice for each memory block packet. Under light loads and for low values of γ, the data cycle length is typically $L + \theta$ where at most one packet is transmitted per cycle. The average delay per memory block packet under light loads is given by $M(1 + \theta) + (L + \theta)$. The maximum throughput obtained by FatMAC is given by the following equations where $\alpha = M\gamma/C$:

$$\Gamma_{fm} = \frac{M[L\gamma + (1 - \gamma)]}{M(1 + \theta) + C_L(L + \theta)} \tag{5.19}$$

The average data cycle length C_L can be approximated to $ML\gamma/C$ for $M \gg C$, high values of γ and under extreme loads.

The relative synchronization overhead in the control cycle is given by $\dfrac{\theta}{1 + \theta}$. For $T_s = 5\mu s$, this represents a 50% overhead per control packet transmission time. For fixed L, the impact of an increase in θ is significant. Let $\Delta\theta$ represent the increase in bit synchronization time. The relative increase in cycle length is given by $\dfrac{\Delta\theta}{1 + \theta}$.

As the ratio θ increases, the impact on system performance increases. The relative impact of θ on throughput is dependent on L and γ. For large values of L and γ, the relative impact of θ on throughput is reduced because of the long cycle lengths. An increase in M worsens the impact of θ on delay due to the longer cycle lengths.

The above discussion shows the severity of bit synchronization overhead on the performance of the protocol. The short control packets force very fast clock acquisition times. Fast clock recovery circuits have been discussed in [135–137]. The approach proposed in [137] achieve clock recovery on the first data transition. A maximum of 100 bits may be transmitted before clock synchronization occurs at speeds of up to 650 Mbps.

I-TDMA/L

I-TDMA can be implemented so that the slot size equals the memory block size. Let I-TDMA/L denote this protocol. I-TDMA/L does not have strict bit synchronization requirements as shown below. The cycle length of I-TDMA/L is independent of traffic characteristics and is given by $M(L+\theta)$. The advantage of I-TDMA/L is that the impact of high bit synchronization time can be mitigated due to the long slot length. The slot length of I-TDMA/L in a system with an 8 Kbyte packet is approximately 328μs at 200 Mbps.

Average packet delay under light loads is given by $M(L+\theta)/2$. Note that bit synchronization overhead is incurred only once per packet with I-TDMA/L unlike FatMAC. The maximum throughput of I-TDMA/L is given by:

$$\Gamma_{td/l} = \frac{C[L\gamma + (1-\gamma)]}{L+\theta} \tag{5.20}$$

Note that the impact of θ on throughput is dependent on the ratio between L and θ. If the packet is sufficiently long, the impact of θ will be small. For fixed L, an increase in θ reduces throughput as seen from the equation. The cycle length increases with θ and the relative increase in cycle length is given by $\frac{\Delta\theta}{L+\theta}$. The relative bit synchronization overhead per cycle is given by $\frac{\theta}{L+\theta}$. For $\theta = 1$ and $L = 64$, this represents a 2% overhead per slot. I-TDMA is attractive in this aspect since it relaxes tight bit synchronization requirements.

I-TDMA/S

In order to minimize the wasted slot utilization with low γ, I-TDMA slot size may be reduced. One solution will be to make the slot size equal to the control packet size. Let I-TDMA/S denote this protocol. The cycle length of I-TDMA/S is M slots. This reduces average latency especially under low loads and low γ. The disadvantage of this approach is that memory blocks are fragmented and sent across different cycles. The packet delay once it reaches the head of the buffer is ML units with I-TDMA/S and L with I-TDMA/L. Bit synchronization requirements will be strict as with FatMAC since slot size is smaller.

The maximum throughput of I-TDMA/S is

$$\Gamma_{td/s} = \frac{C}{1+\theta} \tag{5.21}$$

Note that the maximum throughput is independent of γ. For low values of θ, the maximum throughput approaches C. The average delay under light loads for a control packet is given by $M(1+\theta)/2$. The average delay under light loads for a memory block packet is given by

$$M(1+\theta)/2 + (L-1)M(1+\theta) + 1$$

The first term in the above expression represents the cycle synchronization wait and the next two terms for the packet transmission. For large values of θ, I-TDMA/S may not be appropriate since each memory block packet requires L bit synchronization intervals. I-TDMA/L requires one bit synchronization interval whereas FatMAC requires two for each memory block packet.

I-TDMA/S may be most advantageous under low values of γ. The maximum throughput will be higher than both I-TDMA/L and FatMAC. The downside will be the longer delays suffered by memory block packets due to segmentation. Its possible to implement an approach such as the *universal time-slot* [112] where a node is given a number of slots per frame depending on need, but this would require a signaling manager.

The long delays due to segmenting the packet into L units may be avoided by increasing the slot size to an intermediate value L/x. A memory block packet will be transmitted over x cycles. This reduces utilization during control packet transmission to $1/(L/x+\theta)$. Depending on the value of x, the bit synchronization requirements will be relaxed. A value of $x = 1$ corresponds to I-TDMA/L and $x = L$ corresponds to I-TDMA/S.

Performance Analysis

This section analyzes the impact of bit synchronization on I-TDMA and FatMAC. The above paragraphs provided the equations for throughput and delay considering bit synchronization. The advantage of I-TDMA/L is low bit synchronization overhead and higher throughput. FatMAC has a high bit synchronization overhead but a lower latency. Comparing the low load delay of both protocols, the latency of FatMAC will be less than that of I-TDMA/L as long as:

$$M(1+\theta) + L + \theta < M(L+\theta)/2 + L + \theta$$

which reduces to:

$$\theta < (L-2)$$

This condition specifies the latency advantage but does not represent the strongest condition. The loss in throughput due to θ needs to be compared. In the following equations, let $D_l = \lceil M\gamma/C \rceil$ represent the approximate memory block cycle length of FatMAC. The throughput of FatMAC will be greater than that of I-TDMA/L if the following condition is satisfied:

$$\frac{M(L\gamma + (1-\gamma))}{M(1+\theta) + D_l(L+\theta)} > \frac{C(L\gamma + (1-\gamma))}{L+\theta}$$

which reduces to:

$$\theta_{max} = \frac{M(L-C) - CLD_l}{M(C-1) + D_l}$$

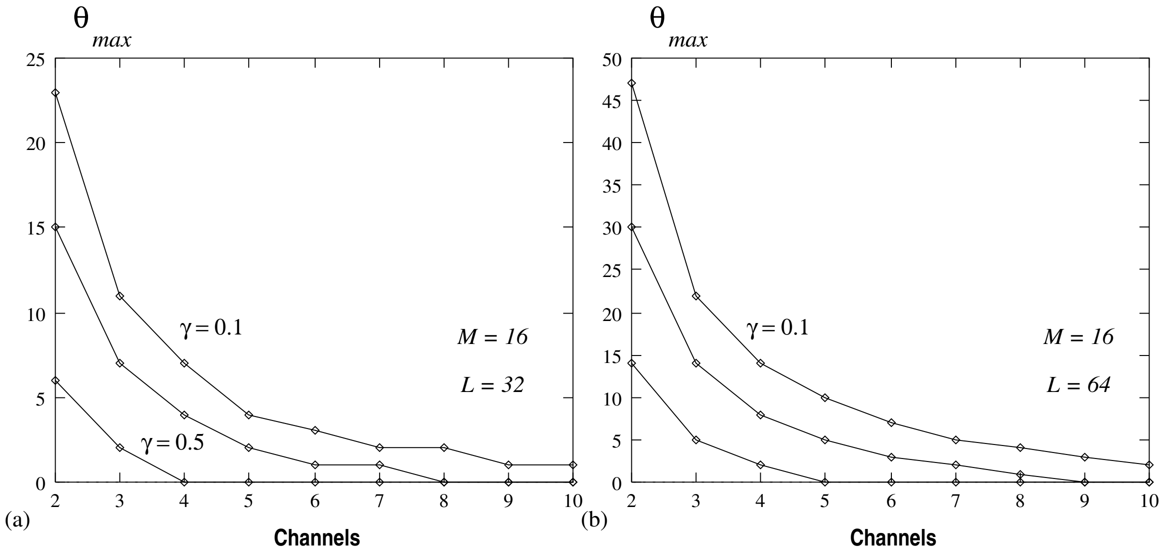

Figure 5.15: Study of θ_{max} for $M = 16$, $L \in \{32, 64\}$, $\gamma \in \{0.1, 0.25, 0.5\}$ and varying C: (a) $L = 32$, and (b) $L = 64$.

where θ_{max} represents the maximum value of θ for which throughput of FatMAC is greater than I-TDMA/L. For $M \gg C$, this equation may be simplified to:

$$\theta_{max} \approx \frac{L(1-\gamma) - C}{C + \gamma - 1}$$

In this case, θ_{max} is independent of M. From the equation, we see that θ_{max} increases with increasing L and decreases with increasing C.

For $M = 16$, $L = 64$, $C = 4$ and $\gamma = 0.1$, $\theta < 10$ for $\Gamma_{fm} > \Gamma_{td/l}$. As the value of γ increases, θ_{max} reduces. For a similar system with $\gamma = 0.25$, θ should be less than 7. For high values of γ, the maximum throughput of FatMAC is always less than that of I-TDMA/L for the same parameters.

Fig. 5.15 studies θ_{max} for $M = 16$, $L \in \{32, 64\}$, $\gamma \in \{0.1, 0.25, 0.5\}$ and varying C. For $L = 32$, θ_{max} is around 20 for $C = 2$ and $\gamma = 0.1$. There is a sharp drop in θ_{max} as C increases. This is due to the fact that I-TDMA/L can take better advantage of an increase in channels. The throughput advantage of FatMAC decreases with increasing γ due to increased slot utilization with I-TDMA/L. An increase in L shows an improvement in θ_{max} as seen from Fig. 5.15(b). This is due to poor slot utilization with I-TDMA/L for low γ.

5.6.2 Slot Synchronization

The last section discussed the bit synchronization requirements of the protocols. The next issue in protocol implementation is the slot synchronization mechanisms. This defines when each node accesses the network during the control phase. The objective is to develop a distributed synchronization algorithm with minimal centralized control. Several synchronization techniques have been studied for satellite systems using TDMA [88–90].

The slot synchronization scheme has to consider propagation delay in fiber, packet processing latency, and lambda partitioner latency. Packet processing is defined as the interval between packet arrival and packet decoding and cycle length information extraction. Lambda partitioner latency is the time required for the packet to propagate through various levels in the hierarchy before reaching the destination node. In a two-level system, a packet passes through one partitioner to reach nodes within the local cluster and through two partitioners to reach nodes outside the local cluster.

Synchronization Techniques:

There are two possible ways of implementing slot synchronization: *lock-step* and *distributed clock* which are described below.

Lock-step mechanism: The first scheme is a *lock-step* approach where each node transmits after transmission from its previous node is complete. This scheme is simpler to implement but is not suitable for large propagation delay. Node m_0 transmits a control cycle synchronization signal to the rest of the network. This is followed by a control packet. Each node knows the id of its predecessor node in the control cycle. When a node receives the control packet from its predecessor node, it transmits its control packet. This ensures that there is no collision in the control phase.

This approach is easier to implement since there is minimal centralized control except for the cycle synchronization signal. In the event that node m_0 fails, the next node in the cycle can take over this responsibility after a timeout period. Slot synchronization in the control phase is achieved by allowing each node to wait for reception of its predecessor's control packet. This leads to "gaps" in the control phase between successive control slots reducing system throughput. This is a serious concern for systems with large propagation delays.

Distributed clock: A distributed clock mechanism will be used in this approach to achieve network slot synchronization. This scheme is similar to the one proposed in [138]. The first node to join the system is designated as the clock node which transmits a clock packet at the start of each cycle. The clock packet contains the propagation delay of the clock node to the partitioner, the current time (to nanosecond accuracy). Each subsequent node that joins the system will also maintain a clock synchronized to the distributed clock. The local clock is set using the clock packet time and the propagation delays of the clock node and the receiving node. By using the time a clock packet is received the receiving node will know the beginning of the new cycle. Using this information, its position in the control cycle, and its own propagation delay, it can calculate the time it should begin broadcasting its control packet. If the propagation delay is very large, the current local clock may be skewed with respect to the global clock. Sufficient preamble and postamble should be provided in the transmitted packet to ensure collisionless transmission.

The performance of the two schemes is compared in detail in [85]. The lock-step scheme is suitable for systems with small propagation delays and processing times. The distributed clock scheme is applicable for systems with small and large propagation delays. At gigabit speeds, the relative propagation delay with respect to packet size will be large and the distributed clock scheme is better suited for this environment. The studies showed that the lock step scheme is suitable for low propagation delay of the order of up to 100 m. The distributed clock mechanism is suitable for values of propagation delay of the order of a few Km. This shows the applicability of the simpler lock step scheme in a multiprocessor environment with locally distributed processors.

Performance Analysis

This section studies the performance of the two schemes for varying propagation delays. It identifies under what conditions a distributed clock scheme is required.

Fig. 5.16 compares the performance of the two slot synchronization schemes under heavy traffic for $M = 16$, $C \in \{2, 4, 8\}$, $L = 64$, and $\gamma \in \{0.1, 0.5, 0.9\}$. Propagation delay has been normalized to lock-step control packet length transmission time. Let τ represent the maximum normalized propagation delay from the node to partitioner. With a 1000 bit control packet transmitted at 200Mbps, $\tau = 1$ would correspond to a distance of 1 Km assuming the propagation speed of 2×10^8 m/s in fiber. The length of the control packet in the distributed clock scheme was taken to have an overhead of 25% for the extra preamble and postamble. Both schemes have been studied with the Data Cycle Stall frame synchronization algorithm.

For $\gamma = 0.1$ and low values of propagation delay up to $\tau = 0.1$, the performance of lock-step scheme is better due to the extra overhead and hence longer control cycle of the distributed clock scheme. For high values of propagation delay, the delay of the lock step

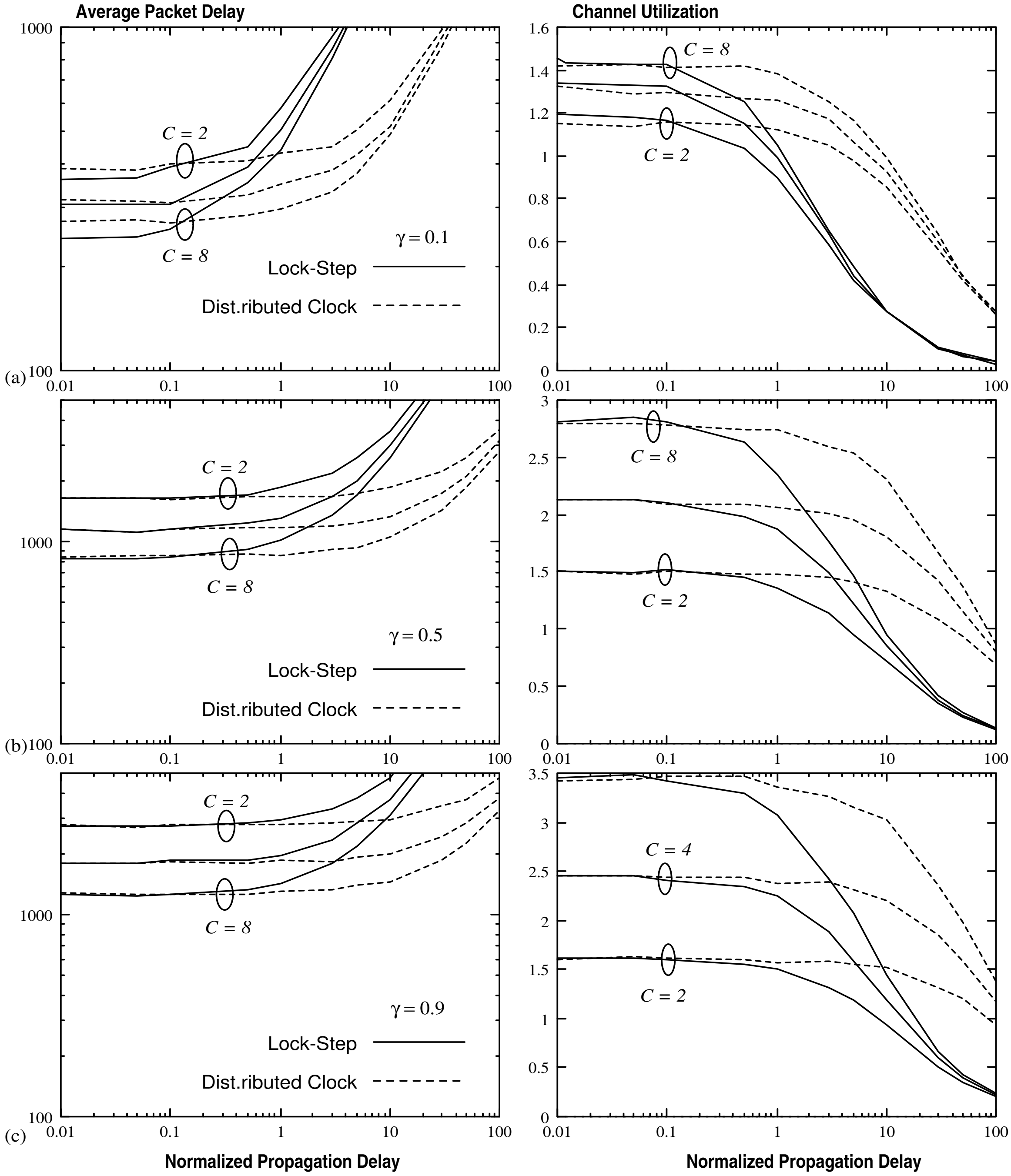

Figure 5.16: Simulation study of impact of propagation delay on lock-step and distributed clock algorithms for $M = 16$, $C \in \{2, 4, 8\}$, and $L = 64$: (a) $\gamma = 0.1$; (b) $\gamma = 0.5$; and (c) $\gamma = 0.9$.

mechanism increases drastically due to the longer control and data cycles. The distributed clock scheme cycle lengths are shorter and utilization is higher in the control and data cycles. The drop in maximum throughput is 30% with the lock-step scheme and negligible with the distributed clock for $M = 16$, $C = 8$ and $\gamma = 0.1$. The performance of the distributed clock scheme degrades after $\tau = 10$ for $\gamma = 0.1$. This is because the start of the data cycle is stalled until all nodes have processed the control packets.

For $\gamma = 0.5$, the performance of both the schemes are identical for small propagation delays up to $\tau = 0.1$ as with $\gamma = 0.1$. The performance of the lock step scheme degrades for higher values of propagation delay. The distributed clock scheme is not severely impacted by propagation delay up to $\gamma = 10$. After that, the performance steadily degrades with increasing propagation delay. Similar performance is observed from Fig. 5.16(c) for $\gamma = 0.9$. The rate of performance degradation with increasing propagation delay is seen to decrease with increasing γ. This is because the data cycle lengths are longer for high γ and the overall impact of propagation delay is reduced.

The above studies show that the lock step scheme is suitable for low propagation delay of the order of up to 100 m. The distributed clock mechanism is suitable for values of propagation delay of the order of a few Km. This shows the applicability of the simpler lock step scheme in a multiprocessor environment with locally distributed processors.

5.6.3 Frame Synchronization

A critical component of implementing the protocol is the frame synchronization mechanism. This determines when the nodes transmit the data cycle after the corresponding control cycle packets have been processes. Each cycle consists of a control phase and data phase as described in Section 5.3. In an ideal network with zero propagation, the data phase immediately follows the control phase. There will be a processing latency following the control phase to allow all nodes to compute their offsets and the cycle length. In a network with large propagation delays, the control packets reach nodes after a significant amount of time. Rather than allow the network to idle in the meantime, the cycles can be staggered so that the data phase for each cycle does not immediately follow its corresponding control cycle.

Synchronization Techniques:

Frame synchronization has been examined via two possible strategies: *Data-Cycle-Stall* and *In-Flight reservation* scheme. The first scheme allows the data phase to immediately follow the control phase. The second scheme staggers the data phase until all nodes have processed the control packets for the corresponding control phase.

Data-Cycle-Stall: If network propagation delay and processing latency are small, the data cycle can immediately follow the control phase. Such a scheme is simple to implement as will be shown later. The disadvantage of this approach is the processing and propagation latencies cause nodes to remain idle for a significant amount of time. This approach is appropriate for multiprocessor interconnection networks where propagation delays are typically small and for small processing latencies.

In-Flight Reservation Scheme: The wait-and-go scheme will not be appropriate for systems with large propagation delay or high processing latency. In this case, a pipelining approach may be used where the control phase is used to reserve access for a data cycle that will take place later.

At startup, nodes transmit the control packets corresponding to the cycle R_0. Each cycle is assigned a unique identifier which can be encoded within the control packet. Rather than wait for the corresponding data phase, nodes transmit control packets for cycle R_1 followed by packets for cycle R_2 and R_3. In the figure shown, all nodes have received and processed control packets for cycle R_0 by the end of control cycle R_3. Nodes transmit their data packets for the first cycle in the D_0 phase. This is followed by control packets for cycle R_4 and so on. The propagation delay and processing latency overcome by overlapping them with control cycle thereby improving network throughput. If propagation delay and processing times are small, the data cycle can immediately follow the control cycle. This approach is therefore identical to wait and go under these conditions.

The minimum time between the initiation of control cycle and start of data cycle will be given by $M + 2\tau + \Delta$. However, the system may not be able to initiate the data phase immediately after all nodes have processed the control packets. The system has to wait for termination of the current data phase or control phase.

Consider a system with 16 nodes, 5 μs control packet, $\tau = 100$ corresponding to 40 Km, and $\Delta = 10$ corresponding to 20μs. The minimum time between start of control cycle and start of data cycle will be 226 time units (452 μs). The first node is ready to initiate the next control cycle 16 time units after the start of the current control cycle. This requires fourteen control cycles to be transmitted before nodes start the first data cycle. The system can ensure continuous packet flow by requiring that a control phase should always follow a data phase. A control phase may be followed by a data phase for an earlier control cycle if the nodes are ready or by a control phase for a future data cycle (its possible that a cycle will have a null data transmission phase).

Performance Analysis

Fig. 5.17 compares the performance of the two frame synchronization schemes under heavy traffic for $M = 16$, $C \in \{2, 4, 8\}$, $L = 64$, and $\gamma \in \{0.1, 0.5, 0.9\}$ using distributed clock for slot synchronization. Propagation delay has been normalized to lock-step control packet

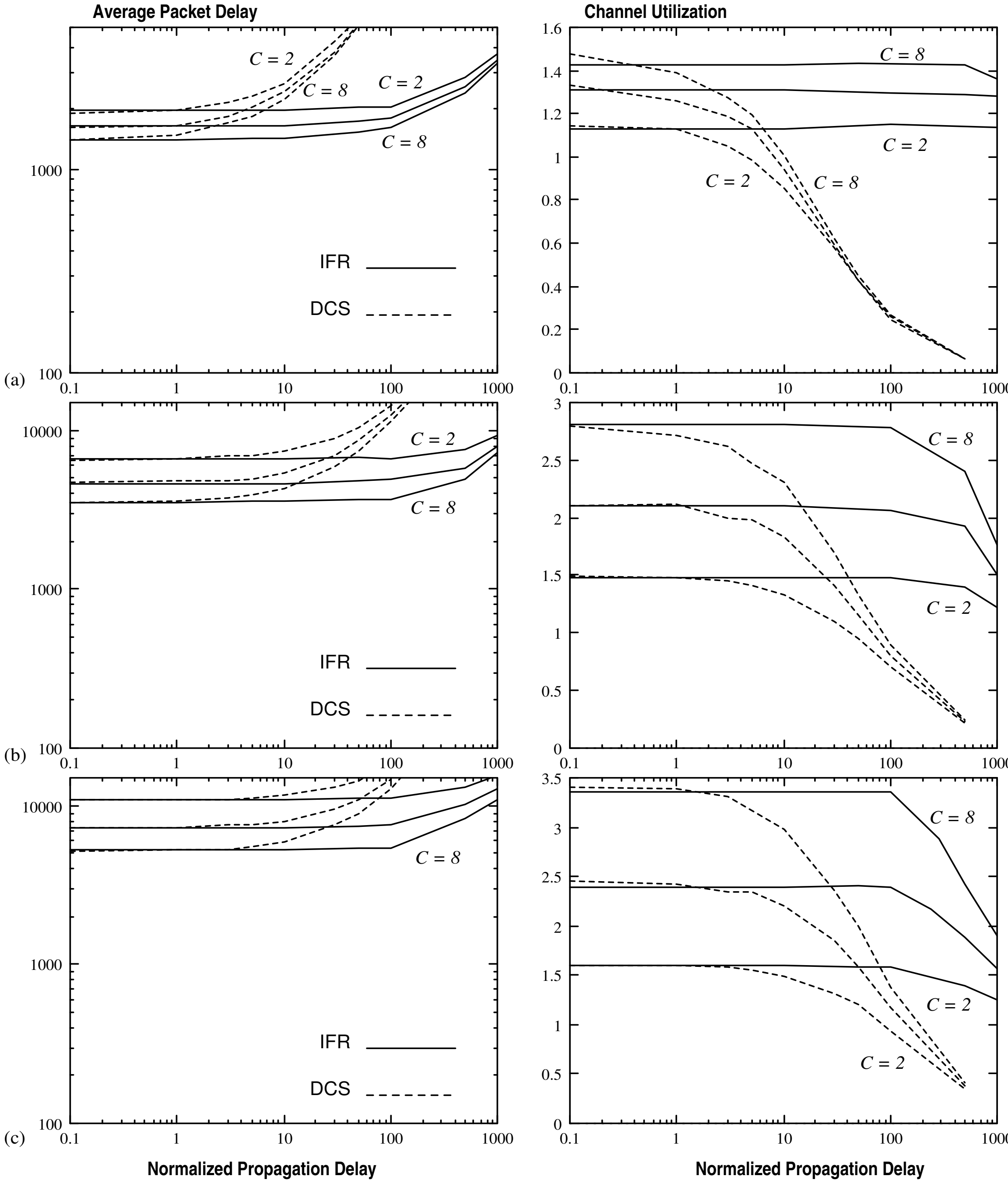

Figure 5.17: Simulation study of impact of propagation delay on frame synchronization schemes for $M = 16$, $C \in \{2, 4, 8\}$, and $L = 64$: (a) $\gamma = 0.1$; (b) $\gamma = 0.5$; and (c) $\gamma = 0.9$.

length transmission time as in the previous paragraphs.

Fig. 5.17(a) shows the relative insensitivity of IFR over the DCS algorithm for long propagation delays and low values of γ. Network utilization with DCS remains steady until $\tau = 1$ and then steadily decreases. The decrease is because of the very small data cycle lengths with small values of γ and hence the relative overhead due to propagation delay is higher. The nodes idle for long times waiting for all packets to be processed. Note that for low propagation delays, the IFR scheme behaves like the DCS scheme. The utilization of the IFR does not degrade even for very large propagation delays since the system consists mostly of small control cycles. The large gaps between the control and data cycle with the DCS scheme are now replaced by control cycles.

For $\gamma = 0.5$, the performance improvement of the IFR scheme is shown in Fig. 5.17(b). The performance of the IFR scheme degrades after $\tau = 100$ whereas the performance of DCS scheme degrades for much lower propagation delays. Similar behavior is exhibited by the two schemes for $\gamma = 0.9$. To summarize, the IFR scheme overlaps the propagation delay by staggering the start of the data cycle and transmitting control cycles for future data cycles. The DCS scheme is appropriate for propagation delays up to the order of a few Km. The IFR scheme is suitable for larger propagation delays up to a few hundreds of Km.

The performance improvement of the IFR scheme for very large propagation delays is dependent on the buffer capacity at the transmitter. Equivalently, it depends on the rate at which traffic is generated by the operating system. For a system with M nodes, the number of control cycles that can be transmitted before the start of the first data cycle is given by $(M + 2\tau)/M$. If the node has enough packets to transmit in these control cycles, utilization will be high. Another advantage of the IFR scheme is the reduction of head-of-line effects. After sending the reservation packet on the current data cycle, the node can send control packets on subsequent control cycles. This reduces the delay of the short packets.

5.7 Summary

This chapter examined the design of protocols that are optimized for the traffic of a DSM environment. The protocol design considered factors including propagation delay, processing latency, synchronization and ease of implementation. A hybrid protocol combining the advantages of reservation and pre-allocation was proposed. The performance of the protocol was analyzed using semi-Markov analytic models and simulation models. The performance was compared to a TDMA based approach. The different synchronization mechanisms for implementation of FatMAC were studied. Alternate strategies such as SA on reservation cycle were studied.

Chapter 6

Conclusions

This chapter summarizes the results of the research and provides scope for future direction.

The first part of the research analyzed random and static access based protocols for a passive WDM star coupled network. The network architecture consisted of a single tunable transmitter and a slow tunable/fixed receiver per node. Detailed semi-Markov analytic models were developed for the protocols. I-SA was shown to be insensitive to variations in system size as long as the ratio of nodes to channels is maintained. I-TDMA* is sensitive to the system size since the length of the cycle is proportional to the number of nodes. However, both protocols take advantage of an increase in the number of channels and I-TDMA* is shown to eliminate the head-of-line performance problems with I-TDMA. The performance gain in I-TDMA* with increasing channels is much higher than that of I-SA. In fact, I-TDMA* is able to achieve the maximum theoretical throughput obtained using C multi-access channels. To provide perspective, the performance of the two protocols was compared to P6, a recently proposed reservation based protocol. The performance of I-TDMA* was shown to exceed that of P6 in terms of higher system capacity and maximum throughput.

The support of acknowledgments and retransmissions is required in random access protocols such as I-SA. The impact of propagation delay and processing latency play a key role in the design of the acknowledgment strategy for these protocols. Stability issues for random access protocols have been considered earlier. This research focused mainly on two different acknowledgment schemes: one based on extended acknowledgment phase following data transmission, and the other based on explicit acknowledgments. The first scheme had low protocol complexity and excellent performance under low propagation delay and processing latency but exhibited poor performance under high propagation delay and processing latency. The crossover point, in terms of the relation of propagation delay and the processing latencies to the slot length, of where the explicit acknowledgment approach is required was identified. The explicit acknowledgment scheme had better performance characteristics under high propagation delay and processing latency.

The impact of optical device switching latency on the performance of pre-allocation protocols for WDM star-coupled networks was studied. A detailed performance analysis was developed for both protocols based on semi-markov analytic models. Two protocols were developed to reduce the impact of propagation delay and processing latency: I-SA* and I-TDMA*. Detailed discrete-event based simulation results were used to study the protocol behavior. I-SA* was shown to have lower packet delay at lighter loads compared to I-TDMA*. However, the performance of I-TDMA* under heavy traffic is superior to that of I-SA*since its saturation point is higher. I-TDMA* is more sensitive to increase in system size. I-SA* is sensitive to propagation delay because it requires acknowledgments unlike I-TDMA*.

The design of photonic local area networks is crucially dependent on the network interconnection topology. The bus topology is no longer considered impractical for optical networks due to improvements in optical amplifiers. This part of the dissertation examined the bus and star topologies with optical amplification with respect to fanout, access arbitration, scalability, cost and fault tolerance. Access arbitration schemes for the multiple multi-access channels were evaluated for both the topologies in terms of performance, scalability, complexity and fault tolerance. The star was shown to have better fault tolerance, fanout, protocol performance and implementational simplicity at the expense of increased fiber cost and complex network synchronization.

This part of the dissertation studied the design of a media access protocol for an optically interconnected multiprocessor system. In particular, it is optimized for the traffic characteristics of a distributed shared memory multiprocessor. One primary objective is minimum latency, and another is low system cost when $C \ll M$ requiring fixed receivers and tunable transmitters per node. A hybrid access scheme combining a reservation approach with fixed receiver architecture was proposed. This protocol made a trade-off of maximum capacity to achieve a significant reduction in latency. The impact of system traffic characteristics and system parameters on protocol performance was studied. The proposed protocol offers lower latencies under light loads and is stable under heavy loads unlike random access schemes. Alternate strategies such as Slotted Aloha on the control channel were also discussed.

The dissertation leads to interesting and challenging problems for future investigation:

- ▷ Extensive testing of the access protocols is required and their impact on overall performance for different traffic types studied. The protocols can be further optimized to provide optimal performance with minimal hardware complexity.

- ▷ The implementation of alternate access schemes for the architecture needs to be studied.

- ▷ Development of higher layer software that utilize the media access support for general interprocessor communication can be examined.

- ▷ Extension of the protocols for implementation on local, metropolitan and wide area networks needs to be thoroughly investigated.
- ▷ The design and implementation of distributed algorithms that incorporate finite communication latency based on the DSM model may be examined.

Bibliography

[1] C. A. Brackett, "Dense wavelength division multiplexing networks: Principles and applications," *IEEE Journal on Selected Areas of Communications*, vol. 8, pp. 948–964, Aug. 1990.

[2] P. Green, *"Fiber Optic Networks"*. Prentice Hall, 1993.

[3] "Special issue on dense wavelength division multiplexing," Aug. 1990.

[4] M. Schwartz, *Telecommunication Networks: Protocols, Modeling and Analysis*. Addison-Wesley, 1987.

[5] M. M. Nassehi, F. A. Tobagi, and M. E. Marhic, "Fiber optic configurations for local area networks," *IEEE Journal on Selected Areas of Communications*, vol. SAC-3, pp. 941–949, Nov. 1985.

[6] 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," *IEEE Journal on Lightwave Technology*, vol. LT-5, pp. 1782–1793, Dec. 1987.

[7] N. Mehravari, "Performance and protocol improvement for very high speed optical fiber local area networks using a passive star topology," *IEEE Journal on Lightwave Technology*, vol. 8, pp. 520–530, Apr. 1990.

[8] M. S. Chen, N. R. Dono, and R. Ramaswami, "A media access control protocol for packet switched wavelength division multiaccess metropolitan area network," *IEEE Journal on Selected Areas of Communications*, vol. 8, pp. 1048–1057, Aug. 1990.

[9] P. W. Dowd, "Random access protocols for high speed interprocessor communication based on a passive star topology," *IEEE Journal on Lightwave Technology*, vol. 9, pp. 799–808, June 1991.

[10] K. Bogineni and P. W. Dowd, "A collisionless multiple access protocol for a wavelength division multiplexed star-coupled configuration: Architecture and performance analysis," *IEEE Journal on Lightwave Technology*, vol. 10, pp. 1688–1699, Nov. 1992.

[11] G. Sudhakar, N. Georganas, and M. Kavehrad, "Slotted Aloha and Reservation Aloha protocols for very high-speed optical fiber local area networks using passive star topology," *IEEE Journal on Lightwave Technology*, vol. 9, pp. 1411–1422, Oct. 1991.

[12] P. A. Humblet, R. Ramaswami, and K. Sivarajan, "An efficient communication protocol for high-speed packet-switched multichannel networks," in *Proc. ACM SIGCOMM'92*, (Baltimore, Maryland), pp. 2–13, Aug. 1992.

[13] I. Chlamtac and A. Ganz, "Frequency-time controlled multichannel networks for high-speed communication," *IEEE Transactions on Computers*, vol. 36, pp. 430–440, Apr. 1988.

[14] A. S. Acampora, M. J. Karol, and M. G. Hluchyj, "Multihop lightwave networks: A new approach to achieve terabit capabilities," in *Proc. ICC'88*, vol. 1, pp. 1478–1484, 1988.

[15] B. J. Ainslie, "A Review of Fabrication and Properties of Erbium-Doped Fibers for Optical Amplifiers," *IEEE Journal on Lightwave Technology*, vol. 9, pp. 220–227, Feb. 1991.

[16] K. Liu and R. Ramaswami, "Analysis of optical bus networks using doped-fiber amplifiers," in *LEOS Summer Topical on Optical Multiple Access Networks*, pp. 41–42, July 1990.

[17] O. K. Tonguz and K. A. Falcone, "Fiber optic interconnection of local area networks: physical limitations of topologies," *IEEE Journal on Lightwave Technology*, vol. 11, pp. 1040–1052, May/June 1993.

[18] P. W. Dowd, K. Bogineni, K. A. Aly, and J. Perreault, "Hierarchical scalable photonic architectures for high-performance processor interconnection," *IEEE Transactions on Computers*, vol. 42, pp. 1105–1120, Sept. 1993.

[19] M. Tam, J. M. Smith, and D. J. Farber, "A taxonomy-based comparison of several distributed shared memory systems," *Operating Systems Review*, vol. 24, pp. 40–67, July 1990.

[20] K. Bogineni and P. W. Dowd, "Performance analysis of two address allocation schemes for an optically interconnected distributed shared memory system," in *Proc. 6^{th} International Parallel Processing Symposium*, pp. 562–566, Mar. 1992.

[21] P. Stenström, "A survey of cache coherence schemes for mulitprocessors," *Computer*, vol. 23, June 1990.

[22] D. Comer, *Internetworking with TCP/IP: Principles, Protocols and Architecture*. Prentice Hall, 1988.

[23] H. Kobrinski and K.-W. Cheung, “Wavelength-tunable optical filters: Applications and technology,” *IEEE Communications Magazine*, vol. 27, pp. 53 – 63, Oct. 1989.

[24] K. W. Cheung, D. A. Smith, J. E. Baran, and B. L. Heffner, “Multiple channel operation of integrated acousto-optic tunable filter,” *IEE Electronic Letters*, vol. 25, pp. 375–376, Mar. 1989.

[25] K. W. Cheung, “Acoustooptic tunable filters in narrowband WDM networks: System issues and network applications,” *IEEE Journal on Selected Areas of Communications*, vol. 8, pp. 1015–1025, Aug. 1990.

[26] C. K. Campbell, “Applications of surface acoustic and shallow bulk acoustic wave devices,” *Proceedings of IEEE*, vol. 77, pp. 1453–1484, Oct. 1989.

[27] W. Warzanskyj, F. Heismann, and R. C. Alferness, “Polarization-independent electro-optically tunable narrow-band wavelength filter,” *Applied Physics Letters*, vol. 53, pp. 13–15, July 1988.

[28] D. A. Smith, J. E. Baran, J. J. Johnson, and K.-W.Cheung, “Integrated-optic acoustically-tunable filters for WDM networks,” *IEEE Journal on Selected Areas of Communications*, vol. 8, pp. 1151–1159, Aug. 1990.

[29] P. A. Kirby, “Multichannel wavelength-switched transmitters and receivers - new component concepts for broad-band networks and distributed switching systems,” *IEEE Journal on Lightwave Technology*, vol. 8, pp. 202–211, Feb. 1990.

[30] J. P. Laude and J. Lerner, “Wavelength division multiplexing/demultiplexing (WDM) using diffraction gratings,” *SPIE-Application, Theory and Fabrication of Periodic Structures*, vol. 503, pp. 22–28, 1984.

[31] C. Dragone, C. Edwards, and R. Kistler, "Integrated optics $n \times n$ multiplexer on silicon," *IEEE Photonic Technology Letters*, vol. 3, pp. 896–899, Oct. 1991.

[32] C. Cremer, N. Emeis, M. Schier, G. Heise, G. Ebbinghaus, and L. Stoll, "Grating spectrograph integrated with photodiode array in InGaAsP/InGaAs/InP," *IEEE Photonic Technology Letters*, vol. 4, no. 1, pp. 108–110, 1992.

[33] K. W. Cheung, S. C. Liew, D. A. Smith, C. N. Lo, J. E. Baran, and J. J. Johnson, "Simultaneous five-wavelength filtering at 2.2 nm wavelength separation using an integrated-optic acousto-optic tunable filter with subcarrier detection," *IEE Electronic Letters*, vol. 25, pp. 636–637, 1989.

[34] K. Kobayashi and I. Mito, "Single frequency and tunable laser diodes," *IEEE Journal on Lightwave Technology*, vol. 6, pp. 1623–1633, Nov. 1988.

[35] T. P. Lee and C. E. Zah, "Wavelength-tunable and single frequency semiconductor lasers for photonic communications networks," *IEEE Communications Magazine*, vol. 27, pp. 42 – 52, Oct. 1989.

[36] V. Jayaraman, D. A. Cohen, and L. A. Coldren, "Extended tuning range in a distributed feedback InGaAsP laser with sampled gratings," in *Conf. Record, Optical Fiber Communications Conference*, p. 165, Feb. 1992.

[37] H. Kobrinski *et al.*, "Fast wavelength switching of laser transmitters and amplifiers," *IEEE Journal on Selected Areas of Communications*, vol. 8, pp. 1190–1202, Aug. 1990.

[38] R. C. Alferness, U. Koren, L. L. Buhl, B. I. Miller, M. G. Young, T. L. Koch, G. Raybon, and C. A. Burrus, "Widely tunable InGaAsP/InP laser based on a vertical coupler intra-cavity filter," in *Conf. Record, Optical Fiber Communications Conference*, pp. 321–324, Feb. 1992.

[39] A. Hunwicks, L. Bickers, M. H. Reeves, and S. Hornung, "An optical transmission system for single-mode local loop applications using spectral sliced technology," in *12th International Fiber Optic Communications and Local Area Networks Exposition (FOC/LAN88)*, Sept. 1988.

[40] S. Wagner and H. Lemberg, "Technology and system issues for a WDM-based fiber loop architecture," *IEEE Journal on Lightwave Technology*, vol. 7, pp. 1759–1768, Nov. 1989.

[41] R. Rund and L. Bersiner, "Experimental demonstration of bidirectional WDM transmission with LED spectral slicing," in *8th Annual European Fibre Optic Communications and Local Area Network Conference (E-FOC90)*, (Munich, Germany), June 1990.

[42] M. M. Girard, C. R. Husbands, and R. Antoszewska, "Dynamically reconfigurable optical interconnect architecture for parallel multiprocessor systems," in *SPIE Proceedings (Optical Applied Science and Engineering)*, (San Diego, CA), July 1991.

[43] K. Liu, "Noise limits of spectral slicing in wavelength-multiplexing applications," in *Conf. Record, Optical Fiber Communications Conference*, p. 174, Feb. 1992.

[44] W. H. Loh, P. R. Morkel, and D. N. Payne, "Wavelength selection and tuning by optical control in a two segment erbium-doped fiber laser," *IEEE Journal on Lightwave Technology*, vol. 6, pp. 43–46, Jan. 1994.

[45] Y. T. Chieng and R. A. Minasian, "Tunable erbium doped fiber laser with a reflection mach-zehnder interferometer," *IEEE Photonic Technology Letters*, vol. 6, pp. 153–156, Feb. 1994.

[46] M. Amman and S. Illek, "Tunable laser diodes utilizing transverse tuning scheme," *IEEE Journal on Lightwave Technology*, vol. 11, pp. 1168–1182, July 1993.

[47] C. Chang-Hasnain, J. R. Wullert, J. P. Harbison, L. T. Florez, N. G. Stoffel, and M. W. Maeda, "Rastered, uniformly separated wavelengths emitted from a two-dimensional vertical-cavity surface-emitting laser array," *Applied Physics Letters*, vol. 58, pp. 31–33, Jan. 1991.

[48] C. J. Chang-Hasnain *et al.*, "Monolithic multiple wavelength surface emitting laser arrays," *IEEE Journal on Lightwave Technology*, vol. 9, pp. 1665–1673, Dec. 1991.

[49] M. Maeda *et al.*, "Multigigabit/s operation of 16-wavelength vertical-cavity surface-emitting laser array," *IEEE Photonic Technology Letters*, vol. 3, pp. 863–865, Oct. 1991.

[50] S. Kitamura, K. Komatsu, and M. Kitamura, "Polarization insensitive semiconductor optical amplifier array grown by selective MOVPE," *IEEE Photonic Technology Letters*, vol. 6, pp. 173–175, Feb. 1994.

[51] M. A. Newhouse, R. F. Bartholomew, B. G. Aitken, L. J. Button, and N. F. Borelli, "Pr-doped mixed halide glasses for 1300 nm amplification," *IEEE Photonic Technology Letters*, vol. 6, pp. 189–191, Feb. 1994.

[52] Y. Ohishi, T. Kanamori, Y. Terunuma, M. Shimuzu, M. Yamada, and S. Sudo, "Investigation of efficient pump scheme for pr^{3+}-doped flouride fiber amplifiers," *IEEE Photonic Technology Letters*, vol. 6, pp. 195–198, Feb. 1994.

[53] M. Fine and F. A. Tobagi, “Demand assignment multiple access schemes in broadcast bus local area networks,” *IEEE Transactions on Computers*, vol. c-33, pp. 1130–1159, Dec. 1984.

[54] C. Tseng and B. Chen, “D-Net, A new scheme for high data rate optical local area networks,” *IEEE Journal on Selected Areas of Communications*, vol. 1, pp. 493–499, Apr. 1983.

[55] F. A. Tobagi, F. Borgonovo, and L. Fratta, “Expressnet: A high-performance integrated-services local area network,” *IEEE Journal on Selected Areas of Communications*, vol. SAC-1, pp. 898–912, Nov. 1983.

[56] J. O. Limb and C. Flores, “Description of Fasnet - A Unidirectional Local-Area Communications Network,” *Bell System Tech. Journal*, vol. 61, pp. 1413–1440, Sept. 1982.

[57] P. W. Dowd, “Optical bus and star-coupled parallel interconnection,” in *Proc.* 4^{th} *International Parallel Processing Symposium*, (Los Angeles, CA), pp. 824–838, Apr. 1990.

[58] P. W. Dowd, “Optical interconnections for computer communications,” Tech. Rep. TR01.A961, IBM Corporation, Apr. 1989.

[59] E. Arthurs, M. Goodman, H. Kobrinski, and M. Vecchi, “Hypass: An optoelectronic hybrid packet switching system,” *IEEE Journal on Selected Areas of Communications*, vol. 6, pp. 1500–1510, Dec. 1988.

[60] M. S. Goodman, H. Kobrinski, M. Vecchi, R. Bulley, and J. Gimlett, “The lambdanet multiwavelength network: Architecture, applications, an demonstrations,” *IEEE Journal on Selected Areas of Communications*, vol. 8, pp. 995–1004, Aug. 1990.

[61] A. S. Acampora, "A multichannel multihop local lightwave network," in *Proc. GLOBECOM'87*, (Tokyo, Japan), pp. 37.5.1–37.5.9, Nov. 1987.

[62] J. A. Bannister, *The Wavelength-Division Optical Network: Architectures, Topologies, and Protocols*. PhD thesis, Computer Science Department, University of California, Los Angeles, California, Mar. 1990.

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

[64] I. Chlamtac, A. Ganz, and G. Karmi, "Circuit switching in multi-hop lightwave networks," in *Proc. ACM SIGCOMM'88 Symposium*, (Stanford, California), pp. 188–199, Aug. 1988.

[65] M. Eisenberg and N. Mehravari, "Performance of the multichannel multihop lightwave network under nonuniform traffic," *IEEE Journal on Selected Areas of Communications*, vol. 6, pp. 1063–1078, Aug. 1988.

[66] M. G. Hluchyj and M. J. Karol, "ShuffleNet: An application of generalized perfect shuffles to multihop lightwave networks," *IEEE Journal on Lightwave Technology*, vol. 9, pp. 1386–1397, Oct. 1991.

[67] J.-F. P. Labourdette and A. S. Acampora, "Partially reconfigurable multihop lightwave networks," in *Proc. IEEE GLOBECOM'90*, pp. 300.6.1–300.6.7, 1990.

[68] J.-F. P. Labourdette and A. S. Acampora, "Wavelength agility in multihop lightwave networks," in *Proc. IEEE INFOCOM'90*, pp. 1022–1029, 1990.

[69] J.-F. P. Labourdette and A. S. Acampora, "Logically rearrangeable multihop lightwave networks," *IEEE Transactions on Communications*, vol. 39, pp. 1223–1230, Aug. 1991.

[70] A. Ganz, W. Gong, and X. Wang, "Wavelength assignment in multihop lightwave networks," in *Proc. IEEE INFOCOM'93*, (San Francisco, CA), Apr. 1993.

[71] B. Li and A. Ganz, "Virtual topologies for WDM star LANs: The regular structures approach," in *Proc. IEEE INFOCOM'92*, 1992.

[72] D. A. Smith and J. J. Johnson, "Low drive-power integrated acoustooptic filter on X-cut Y-propagating $LiNbO_3$," *IEEE Photonic Technology Letters*, vol. 3, pp. 923–925, Oct. 1991.

[73] P. E. Barnsley and P. J. Fiddyment, "Wavelength conversion from 1.3 to 1.55 μm using split contact optical amplifiers," *IEEE Photonic Technology Letters*, vol. 3, pp. 256–258, Mar. 1991.

[74] P. E. Barnsley and P. J. Chidgey, "All-optical wavelength switching from 1.3 μm to a 1.55 μm WDM wavelength routed network: System results," *IEEE Photonic Technology Letters*, vol. 4, pp. 91–94, Jan. 1992.

[75] P. W. Dowd, "Wavelength division multiple access channel hypercube processor interconnection," *IEEE Transactions on Computers*, vol. 41, pp. 1223–1241, Oct. 1992.

[76] B. Mukherjee, "Architectures and protocols for WDM-based local lightwave networks Part I: Single-hop systems," *IEEE Network*, pp. 12–27, May 1992.

[77] B. Mukherjee, "Architectures and protocols for WDM-based local lightwave networks Part II: Multi-hop systems," *IEEE Network*, July 1992.

[78] S. S. Lam, "Satellite packet communications – Multiple access protocols and performance," *IEEE Transactions on Communications*, vol. COM-27, pp. 1456–1466, Oct. 1979.

[79] F. Tobagi, "Multiaccess protocols in packet communication systems," *IEEE Transactions on Communications*, vol. COM-28, pp. 468–488, Apr. 1980.

[80] S. Tasaka, "Multiple-access protocols for satellite packet communication systems: A performance comparison," *Proceedings of IEEE*, vol. 72, pp. 1573–1582, Nov. 1984.

[81] R. Rom and M. Sidi, *Multiple Access Protocols: Performance and Analysis*. Springer-Verlag, 1990.

[82] A. Ganz, "End-to-end protocols for WDM star networks," in *IFIP/WG6.1-WG6.4 Workshop on Protocols for High-Speed Networks*, (Zurich, Switzerland), May 1989.

[83] A. Ganz and Z. Koren, "WDM passive star-protocols and performance analysis," in *Proc. IEEE INFOCOM'91*, pp. 9A.2.1. – 9A.2.10., Mar. 1991.

[84] K. M. Sivalingam, K. Bogineni, and P. W. Dowd, "Pre-allocation media access control protocols for multiple access WDM photonic networks," *Computer Communication Review*, vol. 22, pp. 235–246, Oct. 1992.

[85] K. M. Sivalingam, D. Crouse, and P. Dowd, "A collisionless hybrid WDMA protocol with low-latency support and dynamic bandwidth allocation," Tech. Rep. TR-08-93-02, State University of New York at Buffalo, Department of Electrical and Computer Engineering, Aug. 1993.

[86] K. M. Sivalingam, K. Bogineni, and P. Dowd, "Design and performance analysis of preallocation protocols for WDM photonic networks," *Proceedings SPIE (High-Speed Fiber Networks and Channels)*, vol. 1784, pp. 193–204, Sept. 1992.

[87] K. M. Sivalingam and P. W. Dowd, "Latency hiding strategies of pre-allocation based media access protocols for WDM photonic networks," in *Proc. 26th IEEE Simulation Symposium*, pp. 68–77, Mar. 1993.

[88] W. G. Schmidt, "The Application of TDMA to the INTELSAT IV Satellite Series," *COMSAT Tech. Rev.*, vol. 3, pp. 257–276, Fall 1973.

[89] C. R. Carter, "Survey of Synchronization Techniques for a TDMA Satellite Switched System," *IEEE Transactions on Communications*, vol. COM-28, pp. 1291–1301, Aug. 1980.

[90] K. Feher, *Digital Communications:Satellite/Earth Station Engineering*. Prentice-Hall Inc., 1983.

[91] K. Bogineni, K. M. Sivalingam, and P. W. Dowd, "Switching latency impact on star-coupled WDM photonic network pre-allocation protocol performance," *Journal on High Speed Networks*, vol. 1, no. 4, 1992.

[92] 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 of Communications*, vol. 11, pp. 590–604, May 1993.

[93] K. M. Sivalingam, K. Bogineni, and P. W. Dowd, "Acknowledgment techniques of random access based media access protocols for a WDM photonic environment," *Computer Communications*, vol. 16, pp. 458–471, Aug. 1993.

[94] M. Girard, C. Husbands, and P. Dowd, "Media access protocols for spectral sliced WDMA testbed," *SPIE Proceedings (High-Speed Fiber Networks and Channels)*, vol. 1784, pp. 169–180, Sept. 1992.

[95] T. N. Mudge and H. B. Al-Sadoun, “A semi-markov model for the performance of multiple-bus systems,” *IEEE Transactions on Computers*, vol. C-34, pp. 934–942, Oct. 1985.

[96] S. E. Ross, *Stochastic Processes*. John Wiley & Sons, 1983.

[97] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis*. McGraw Hill, 1991.

[98] J. R. Jump, *YACSIM Reference Manual*. Rice University, Department of Electrical and Computer Engineering, 1.2 ed., August 1992.

[99] K. M. Sivalingam and P. W. Dowd, “A performance study of photonic local area network topologies,” in *Proc. International Workshop on Modeling and Simulation of Computer and Telecommunications Systems (MASCOTS)*, pp. 79–83, Jan. 1994.

[100] M. Karol, “Exploiting the attenuation of fiber-optic passive taps to create large high-capacity LAN’s and MAN’s,” *IEEE Journal on Lightwave Technology*, vol. 9, pp. 400–408, Mar. 1991.

[101] A. E. Willner, A. Saleh, H. Presby, D. DiGiovanni, and C. Edwards, “Star Couplers with Gain Using Multiple Erbium-Doped Fibers Pumped with a Single Laser,” *IEEE Photonic Technology Letters*, vol. 3, pp. 250–252, Mar. 1991.

[102] I. Chlamtac and A. Ganz, “A multibus train communication architecture (AMTRAC) for high-speed fiber optic networks,” *IEEE Journal on Selected Areas of Communications*, vol. 6, pp. 903–912, July 1988.

[103] S. Banerjee and B. Mukherjee, “FairNet: A WDM-based multiple channel lightwave network with adaptive and fair scheduling policy,” *IEEE Journal on Lightwave Technology*, vol. 11, pp. 1104–1112, May/June 1993.

[104] K. W. Cheung and V. Mak, "EQEB - A multiple channel extension of the DQDB protocol with tunable channel access," in *Proc. GLOBECOM*, Dec. 1992.

[105] K. M. Sivalingam, K. Bogineni, and P. W. Dowd, "Pre-allocation media access control protocols for multiple access WDM photonic networks," in *Proc. ACM SIGCOMM'92*, pp. 235–246, Aug. 1992.

[106] R. M. Newman, Z. L. Budrikis, and J. L. Hullett, "The QPSX Man," *IEEE Communications Magazine*, vol. 26, pp. 20–28, Apr. 1988.

[107] E. L. Hahne, A. K. Choudhury, and N. F. Maxemchuk, "DQDB Networks with and without Bandwidth Balancing," *IEEE Transactions on Communications*, vol. 40, pp. 1192–1204, July 1992.

[108] K. Bogineni and P. W. Dowd, "An optically interconnected distributed shared memory system: Architecture and performance analysis," *International Journal on High Speed Computing*, vol. 4, pp. 179–212, Sept. 1992.

[109] P. W. Dowd and K. Bogineni, "Performance and simulation techniques for optically interconnected distributed shared memory computer systems," in *Proc. International Conference on Simulation in Engineering Education*, pp. 31–35, Jan. 1992.

[110] L. N. Bhuyan and D. P. Agrawal, "Design and performance of generalized interconnection networks," *IEEE Transactions on Computers*, vol. C-32, pp. 1081–1090, Dec. 1983.

[111] K. Bogineni, *Design and Performance Analysis of Optically Interconnected Parallel Computers*. PhD thesis, State University of New York at Buffalo, July 1993.

[112] R. Thompson, R. Anderson, J. Camlet, and P. Giordano, "Experimental modular switching system with a time-multiplexed photonic center stage," in *Proceesings of*

the OSA Topical Meeting on Photonic Switching, (Salt Lake City, Utah), pp. 212–218, Optical Society of America, Mar. 1989.

[113] I. Jacobs, R. Binder, and E. Hoverstein, "General purpose packet satellite networks," *Proc. IEEE*, vol. 66, pp. 1448–1468, Nov. 1978.

[114] P. Dowd and K. M. Sivalingam, "A multi-level WDM access protocol for an optically interconnected parallel computer," in *Proc. IEEE INFOCOM'94*, (Toronto, Ontario Canada), June 1994.

[115] V. F. Kolchin, B. A. Sevastyanov, and V. P. Chistyakov, *Random Allocations*. New York: John Wiley and Sons, 1978. Translated from the Russian original *Slučajnye Razmeščeniya*.

[116] G. H. Gonnet, "Expected length of the longest probe sequence in hash code searching," *J. ACM*, vol. 28, no. 2, pp. 289–304, 1981.

[117] J. S. Vitter and P. Flajolet, "Analysis of algorithms and data structures," in *Handbook of Theoretical Computer Science* (J. van Leeuwen, ed.), vol. A: Algorithms and Complexity, ch. 9, pp. 431–524, North Holland, 1990.

[118] D. Chaiken, C. Fields, K. Kurihara, and A. Agarwal, "Directory-based cache coherence in large-scale multiprocessors," *IEEE Computer*, pp. 49–58, June 1990.

[119] J. P. Singh, W. D. Weber, and A. Gupta, "Splash: Stanford parallel applications for shared mermory," Tech. Rep. CSL-TR-91-469, Stanford University, 1991.

[120] Y. Gao and A. Ganz, "Efficient algorithms for SS/TDMA scheduling," *IEEE Transactions on Communications*, vol. 40, pp. 1367–1374, Aug. 1992.

[121] A. Ganz and Y. Gao, "Traffic scheduling in multiple WDM star systems," in *Proc. ICC'92*, (Chicago), June 1992.

[122] W. Crowther, R. Rettburg, D. Walden, S. Ornstein, and F. Heart, "A system for broadcast communication: Reservation aloha," in *Proc. 6th Hawaii Internat. Conf. System Sciences*, pp. 371–374, 1973.

[123] R. Binder, "A dynamic packet-switching system for satellite broadcast channels," in *Proc. of the International Conference on Communications*, (San Francisco, CA), pp. 41.1–41.5, June 1975.

[124] L. Roberts, "Dynamic allocation of satellite capacity through packet reservation," in *AFIPS Proc. National Comput. Conf.*, vol. 42, (New York), pp. 711–716, 1973.

[125] S. Tasaka and Y. Ishibashi, "Stability and performance of a packet reservation system for satellite broadcast channels," in *Proc. IEEE GLOBECOM*, pp. 35–42, Nov. 1982.

[126] S. Tasaka and Y. Ishibashi, "A reservation protocol for satellite packet communication – a performance analysis and stability consideration," *IEEE Transactions on Communications*, vol. 32, pp. 920–927, Aug. 1984.

[127] M. Gerla, L. Nelson, and L. Kleinrock, "Packet satellite multiple access: Models and measurements," in *National Telecomm. Conf.*, pp. 12:2.1–12:2.8, 1977.

[128] F. Borgonovo and L. Fratta, "SRUC: a technique for packet transmission on multiple access channels," in *Proc. Int. Conf. Computer Communication*, (Kyoto, Japan), pp. 601–607, 1978.

[129] J. E. Wieselthier and A. Ephremides, "A new class of protocols for multiple access in satellite networks," *IEEE Transactions on Communications*, vol. 25, pp. 865–880, 1980.

[130] E. P. Greene and A. Ephremides, "Distributed reservation control protocols for random access broadcasting channels," *IEEE Transactions on Communications*, vol. 29, pp. 726–735, May 1981.

[131] E. Wong and T. S. Yum, "A controlled multiaccess protocol for packet satellite communications," *IEEE Transactions on Communications*, vol. 39, pp. 1133–1140, July 1991.

[132] X. Wang, J. Kaniyil, Y. Onozato, J. Lui, S. Shimamoto, and S. Nogichi, "Performance analysis of a combined random-reservation access scheme," *IEEE Transactions on Communications*, vol. 39, pp. 478–481, Apr. 1991.

[133] A. K. Elhakeem, S. Bohm, M. Hachicha, T. Le-Ngoc, and H. Mouftah, "Analysis of a new multiaccess/switching technique for multibeam satellites in a prioritized ISDN environment," *IEEE Journal on Selected Areas of Communications*, vol. 10, pp. 378–390, Feb. 1992.

[134] M. Carrato and P. W. Dowd, "Dynamic Cycle Interleaved Slotted Aloha - A Pre-allocation based WDM Access Protocol," *SPIE Proceedings (High-Speed Fiber Networks and Channels)*, vol. 1784, pp. 217–228, Sept. 1992.

[135] D. G. Messerschmitt, "Synchronization in digital system design," *IEEE Journal on Selected Areas of Communications*, vol. 8, pp. 1404–1419, Oct. 1990.

[136] C. S. Li and C. J. Georgiou, "Parallel Optical Interconnects for Scalable Coherent Interface," in *ASIC*, pp. 482–485, IEEE, Sept. 1993.

[137] M. Banu and A. Dunlop, "Clock recovery circuits with instantaneous locking," *Electronics Letters*, vol. 28, pp. 2127–2130, Nov. 1992.

[138] G. Semaan and P. Humblet, "Timing and dispersion in WDM optical star networks," in *Proceedings of IEEE INFOCOM*, pp. 573–577, Mar. 1993.