Implementation and Modeling of a Scheduled Optical Flow Switching (OFS) Network

by

Bishwaroop Ganguly

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philsophy Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2008

Author .....
Department of Electrical Engineering and Computer Science
May 23, 2008

Certified by .....
Professor Vincent Chan
Joan and Irwin Jacobs Professor, Electrical Engineering and Computer Science Department
Thesis Supervisor

Accepted by .....
Professor Terry P. Orlando
Chair, Department Committee on Graduate Students

Implementation and Modeling of a Scheduled Optical Flow Switching (OFS) Network

by

Bishwaroop Ganguly

Submitted to the Department of Electrical Engineering and Computer Science
on May 23, 2008, in partial fulfillment of the
requirements for the degree of
Doctor of Philsophy Computer Science and Engineering

Abstract

In this thesis we present analysis of Optical Flow Switching (OFS), an architectural approach for enabling all-optical user to user connections for transmission of Internet traffic. We first describe a demonstration of OFS on the ONRAMP test environment which is a MAN optical network implemented in hardware in the Boston geographic area. This demonstration shows the viability of OFS in an actual implementation, with good performance results and an assessment over OFS overheads. Then, we use stochastic models to quantify the behavior of an OFS network. Strong quantitative evidence leads us to draw the conclusion that scheduling is a necessary component of any architectural approach to implementing OFS in a Metro Area network (MAN).

Thesis Supervisor: Professor Vincent Chan
Title: Joan and Irwin Jacobs Professor, Electrical Engineering and Computer Science Department

Contents

Chapter 1

Introduction

1.1 Optical Flow Switching

Wide Area Networks (WAN) carrying Internet traffic today use optical Wavelength Division Multiplexing (WDM) technology almost without exception. This technology allows terrestrial fiber optic networks to carry literally hundreds of data channels in each fiber, each at data rates as high as 40Gb/s or higher. From a physical layer perspective, a number of recent technological advances have occurred in WDM networks such as hundreds of channels per fiber, dispersion managed fiber that allows signals to travel hundreds of kilometers without regeneration, and fast, high port count optical switches with hundreds of ports, reconfiguring in tens of microseconds. This technology has the potential to enable a network that provides *all-optical* connections between users. An all-optical network has the potential to change the way data is stored, shared and used on the Internet. Unfortunately, advances in network architecture and network design have not matched the rapid advances in WDM physical layer technology.

Today, high-bandwidth, agile all-optical network capabilities have largely not been made visible to the Internet user. This is in part due to a lack of architectural understanding of how these new technologies can benefit the end user. There is a need to study of the fundamental properties of all-optical networks, and their behavior. By studying this behavior, we can hope to design network architectures that utilize

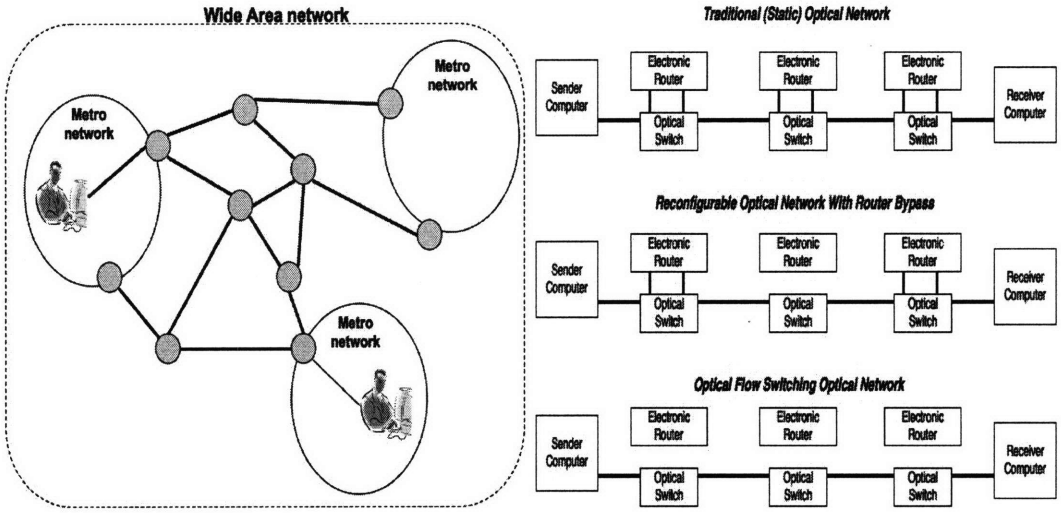
advanced optical network technology, while providing benefit to both the network owner/administrator and end user. In this thesis, we propose and study a model of an all-optical network approach termed Optical Flow Switching (OFS).

We first describe an implementation of OFS on the ONRAMP optical testbed. This demonstration shows the viability of OFS in a MAN network. We then analyze a stochastic model for OFS, and we study it's average-case behavior. The model is simplified, but detailed in that all non-trivial states of the system are numerically analyzed. We use both numerical and analytical results from this model to discern fundamental properties that apply to virtually all all-optical network approaches that have been proposed, including OFS.

A network employing OFS uses WDM technology to create all-optical user-to-user connections for high rate data transfer. These one-way transactions are also called flows. Flows can often be short duration (one second or less), so the problem of network management, control, and reconfiguration is highly dynamic. Current-day WAN optical networks are generally statically or quasi-statically configured. Dynamic OFS as defined here is a departure from this, as it is reactive to asynchronous individual user data (e.g. file) transfer requests. From a hardware standpoint, such a service is being enabled by advances in WDM network technology. These include faster all-optical switches, lower loss fiber, and tunable lasers and filters as demonstrated in [3].

By way of an example of OFS's potential usefulness, consider a remote user of a multi-processor supercomputer. She may be running a sophisticated simulation that requires visualization in real time to adjust program parameters and control the direction of the simulation. The data needed to visualize such simulations is generally on the order of the size of the physical memory of the supercomputer, which today can be as large as a terabyte. In the current day Internet, a transfer of this type of data can take an enormous amount of time, usually using an application such as File Transfer Protocol (ftp). Consequently, the supercomputer user cannot visualize and adjust her computation very often. However, if an OFS network was available to schedule the data transfer at optical rates, the transfer would take a matter of

seconds, and visualization could happen with more frequency. In this case, OFS has changed the computing paradigm of a supercomputer user, by allowing more rapid visualization and program parameter adjustment.



(a) WAN/MAN view of optical network (b) OFS vs. static network comparison

Figure 1-1: OFS motivation

Figure 1-1(a) shows a possible architecture for a network employing OFS. It shows a Wide Area Network comprised of several connected Metro-Area Networks (MAN) that provide (flow enabled) users with an ingress to the WAN. OFS connections are set up between users, and are short duration. It remains an open question whether WAN optical resources will be statically configured, or be dynamically controllable. In any case, it is likely that at some level of the network hierarchy (WAN or MAN), dynamic OFS will be advantageous. From the networks perspective, OFS's goal is efficient sharing of resources with minimal network cost. Our work will study OFS itself, treating the interactions between the dynamic and static parts of Figure 1-1(a) as future work.

The benefits of OFS in are numerous. For the user of OFS, low delay and high bandwidth is provided for one-way data transfers, as the data travels at fiber-optic rates. This can be up to 40Gb/s with current technology, so very large database transfers, for example, take a matter of seconds or even a fraction of a second. If this

can be provided to users at a low cost, then they will have incentive to send large transactions all-optically, and also design applications to use this technology.

Another benefit of OFS is transparency of the connection between users. Since the connection between the users is all optical with no optical-to-electronic conversion intervening, the transmission rate, or even modulation scheme can be negotiated between two users. The network need not know the manner in which the users use the connection given some reasonable signal-to-noise management. The network needs only to be concerned with the time duration of the OFS communication needed by the user.

From the network's point of view, OFS has the benefit of *electronic router bypass.* All-optical data transfers bypass all electronic routers, including both the ingress and egress routers, as shown in the bottom-most illustration of Figure 1-1(b). This is in contrast to other "all optical" approaches that we will detail in the next chapter which use traffic aggregation at the ingress to the network. In OFS, flow-switched data avoids being converted to electronics, routed and re-converted to optics at each hop, as happens in traditional static optical networks. This has the effect of lessening the burden on electronic router buffer memory, routing hardware and software, and optical port costs. In addition, we perceive that the use of all-optical bypass will help reduce power dissipation by the high-speed electronic components of the router. Generally speaking, high-speed electronic ports are the key cost for router manufacturers. Power dissipation is also fast becoming an issue for router manufacturers as higher speed, power hungry electronics are needed to modulate/demodulate at optical line rates.

Our view is that any all-optical network architecture, including OFS, must address the issue of cost at least qualitatively. We therefore preclude the use in our models of wavelength changers, optical buffers and electronic buffers which tend to add cost to the network. Electronic buffers and wavelength converters generally necessitate high-speed electronics which are the key cost of network equipment. Optical buffers are lower cost, but have a large footprint and introduce other architectural problems such as variable loss and timing issues. As we will discuss in the next chapter, related

work often presents results that assume wavelength conversion or optical or electronic buffering in the network. While this does help to enhance network performance results, it ignores one of the key original goals of all-optical switching which is lowering network cost. Lack of wavelength conversion and buffering in the network does complicate our models by introducing the issue of *wavelength continuity* which will be discussed later.

The lowest picture in the figure 1-1(b) shows that an OFS transaction bypasses all the electronic routers optically. This reduces the burden on the electronic router ports and buffer memory. This is becoming especially important as today optical data rates are making opto-electronic conversion high-rate and expensive in terms of dollars and power consumption, the so-called *opto-electronic bottleneck*. However, these benefits can only be reaped if the following can be shown: 1. A significant amount of Internet traffic can be handled by OFS, and 2. OFS is efficient, and can be implemented efficiently and cheaply.

1.2 Application of OFS to Internet Traffic

WDM switches using MEMS and other technologies and network firmware are becoming cheaper, faster and more robust. However, we recognize that the switching agility of even advanced optical devices does not match that of electronic switching in silicon. In other words, the *granularity* of the data units being switched optically must in general be larger than that of electronic infrastructure in order to amortize the switching time. Approaches such as Optical Packet Switching are an attempt to match these granularities but have not been successful outside of expensive experimental technology. OFS is an approach that focuses on current-day optical switching technology enabling transactions as we show below.

There is significant evidence that Internet traffic displays a heavy-tailed characteristic [1],[2]. Succinctly, this means that a large volume of total Internet traffic (bytes) is contained in a small number of large transactions (i.e. large flows). If OFS can capture a significant portion of this "heavy-tail" of traffic (i.e. large transactions),

then it can relieve network routers of a significant amount of traffic, in terms of bytes, making it beneficial to implement.

The best known model for a heavy-tailed traffic distribution is the Pareto distribution, with Probability Density Function (PDF):

$$P(x) = \alpha\, k^{\alpha}\, x^{-\alpha-1} \qquad k > 0, \alpha > 1, x \geq k$$

The relevant parameters for this distribution are α and k. α determines the tail weight, with the weight increasing as $\alpha \to 1$. Note that $\alpha \leq 1$ results in an infinite tail weight (see below). The parameter k describes the domain of the PDF, which begins at k and goes to infinity. The expectation of a random variable obeying a Pareto distribution is $\frac{\alpha k}{\alpha - 1}$ and its variance is infinite if $\alpha \leq 2$, which is our case of interest.

For any PDF, the weight $W(\tau)$ of the tail beginning at $x = \tau$ is defined as follows:

$$W(\tau) = \int_{\tau}^{\infty} P(x) x dx$$

The expression is similar to the expectation of the PDF conditioned on the event $x > \tau$, except for the absence a scaling factor. Assuming that Internet traffic obeys a Pareto distribution, Figure 1-2(a) shows the ratio $\frac{W(\tau)}{W(k)}$ vs. Flow Size (in bytes) for the Pareto distribution of flows with $\alpha = 1.06$, for two values of k. Here, flow size refers to the size of the transaction (in bytes) and is the argument to the Pareto PDF. In this analysis, think of τ as the threshold above which we will send the transaction all-optically. Thus, OFS would 'capture' all bytes in the distribution above the value τ. The parameter values (k, α) were chosen based on [1] and the current and predicted sizes of transactions in the Internet.

We can define the OFS threshold as a particular flow size, in bytes, where flows larger than the threshold are sent via OFS and smaller are handled by traditional electronic routing. Note that for OFS, transaction size can be described in terms of time or bytes since the transaction is transported at optical line rate. In the Figure 1-2(a), the two demarcation (star and solid dot) emphasize the value of this ratio

at one particular flow size threshold (1 Gigabit). For both values of k, a significant fraction of bytes transferred (>50%) are contained in the tails of the distribution where OFS will be active. This suggests that OFS is applicable to Internet traffic given that the heavy-tailed assumption holds.

It is important to note that we see OFS as working in conjunction with electronic switching infrastructure, and is not intended to replace it. Though optical switch technology is advancing at a rapid pace, we do not anticipate it ever being as agile as electronic switching. The goal of OFS is to offload large transactions onto optical infrastructure, while leaving smaller transactions to electronics. This symbiotic network design would can be adjusted as the various technologies (optical or electronic) evolve with time. However we do not anticipate using a large number of network resources (i.e. computation, wavelength, control network) allocated for OFS, so it must use them efficiently.

1.3 Thesis Description

This thesis is comprised of two principal sets of results. The first is of an OFS demonstration performed on the ONRAMP optical testbed. ONRAMP is an optical MAN network that was built in the greater Boston area (with a loopback long-haul link to demonstrate delay) in order to demonstrate optical technologies. We have implemented OFS on this test bed, and produced a number of results including OFS flow performance and an assessment of OFS overheads that impede performance. Overall, this demonstration showed that OFS is indeed a technology within grasp in the short term without using specialized hardware and software support.

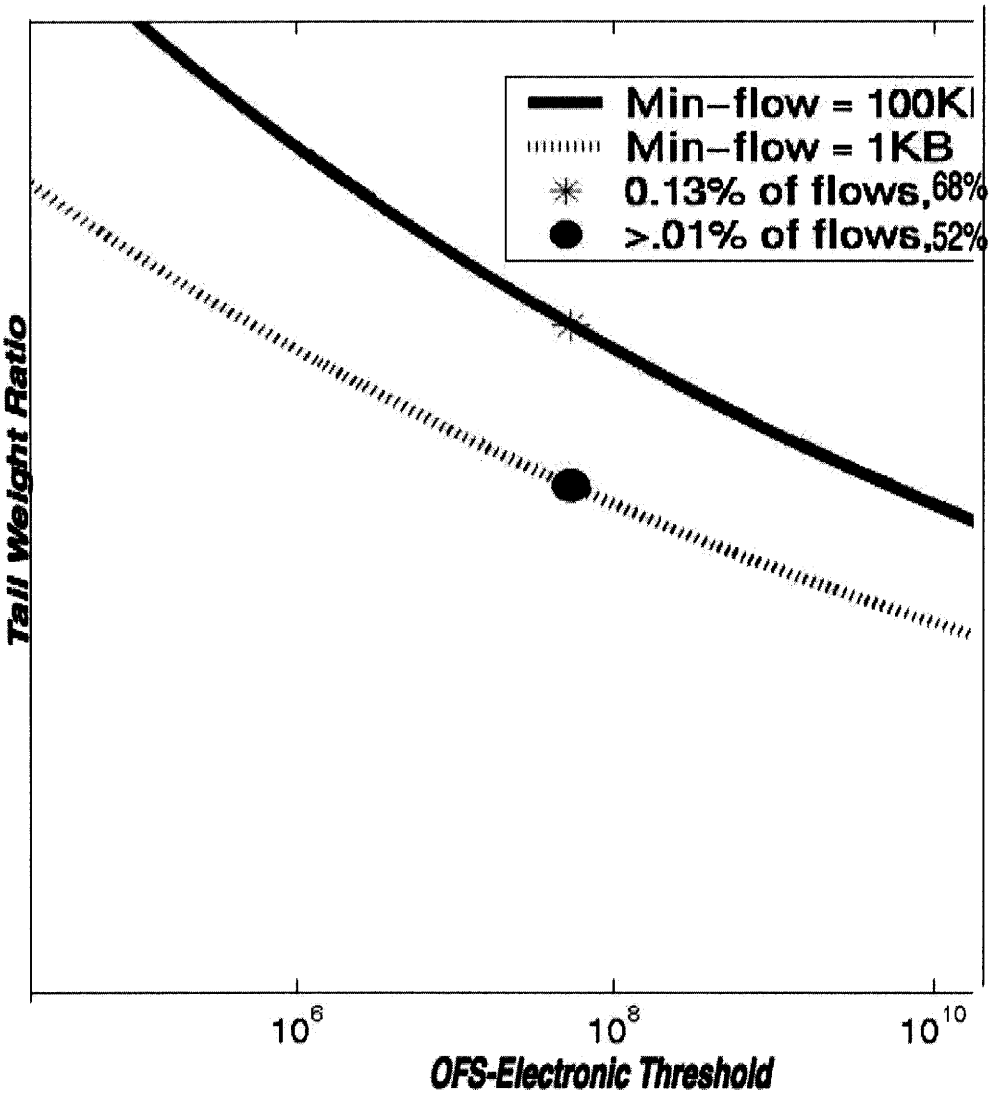
In the second part of the thesis, we study OFS using *stochastic modeling*. We have developed stochastic models that can be used to study the behavior of a network employing OFS. The reason for this choice of model is to study average case behavior. In other words, we pose the question: How do these models perform on average, given a particular traffic arrival and service model? In general, simulations attempt to find this behavior, but are run for a fixed number of iterations. Our analysis will solve

models for average-case or stationary distributions that have measurably converged to that distribution. We also present analytical results that reveal theoretical bounds on certain aspects of the model.

There has been a number of studies of performance of all-optical switching techniques, as will be discussed in the next chapter. These studies have typically used simulation. The issue with this is that the number of discrete states that a large all-optical network can take on is large. In many cases, it is unclear if the output of the simulation is representative of average case behavior. The stochastic modeling we present in this thesis alleviates these concerns, because of the concept of *convergence.* Convergence is the idea that all non-trivial states in the model have been assigned a non-zero probability of occupancy in the stationary distribution and that these describe the behavior of the model in the average case. This type of modeling is more computationally intensive, but we use specialized techniques and parallel processing to address the additional computation.

We apply several techniques to examine these models including, simulation with convergence checking, numerical analysis, and closed-form analysis. The goal is to quantify the performance of OFS networks under specified conditions and then to discern any fundamental properties of them.

The thesis is organized as follows: Chapter 2 presents background work and projects related to OFS. This will include work on optical network architecture, routing and wavelength assignment (RWA), network protocols, and optical hardware demonstrations. Chapter 3 details an OFS demonstration that we have done, and performance results. Chapter 4 presents analysis of control plane strategies for OFS. Chapter 5 details our basic OFS analytical network model, and justify design decisions and assumptions we have made. Chapter 6 presents numerical analysis that shows the benefits of a scheduled approach to OFS. Chapter 7 summarizes and concludes the thesis.



(a) Pareto Tail Ratio

Figure 1-2: Heavy-tailed distribution tail weight ration

Chapter 2

Background and Related Work

2.1 Optical Network Technologies

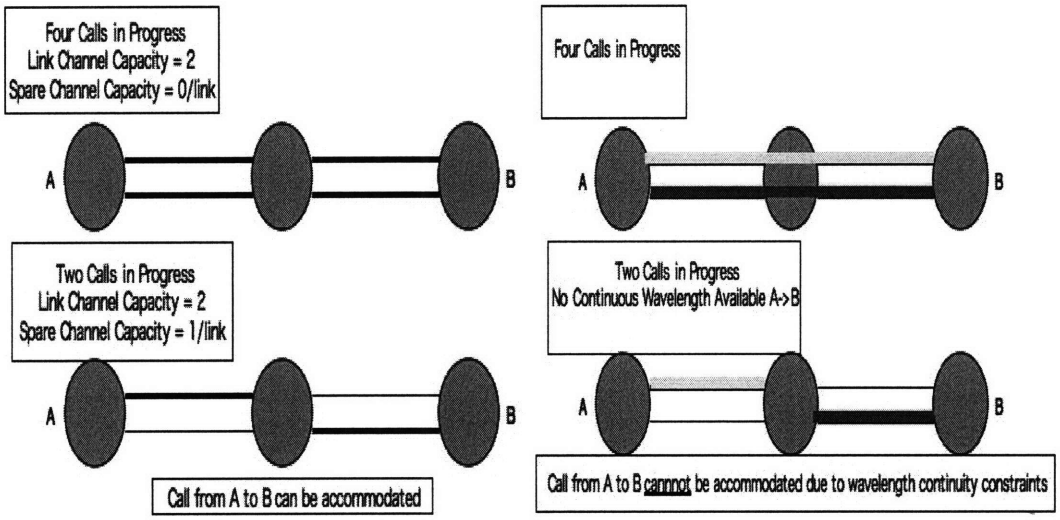
There has been a great deal of related work in the field of all-optical network architecture. Both [17] and [18] provide overviews of issues with all-optical networking. The former also makes reference to connection setup and connection scheduling issues at a qualitative level. All-optical networking can be further decomposed into two separate categories. These are connection-oriented and connection-less all-optical networks. OFS falls under the former category, and other approaches such as Optical Packet Switching (OPS) and Optical Burst Switching (OBS) comprise the latter. We discuss related work in these categories here.

2.2 Optical Network Architecture

2.2.1 Connection-oriented all optical networks

Connection-oriented optical networks are closely related to the theory of *circuit-switched networks* which has been studied for more than 50 years. Some of the seminal work in the field was done by Kelly [4] and also appears in [7]. This work uses mostly queuing theory to discern properties of networks that perform collective node allocations. This work models circuit-switched networks as networks of nodes

and links with each link having a certain capacity of calls. A node to node call that arrives to the network is admitted or not admitted based on the residual capacity of the links in the chosen route. If sufficient capacity is present the call is admitted, otherwise it is dropped. There are variations on this model, but this is the basic idea of a circuit-switched network.



(a) Circuit-Switched network (b) Wavelength continuous network

Figure 2-1: Circuit-switched versus wavelength continuous illustration

The key difference between circuit-switched networks and our work is the need for wavelength continuity in all-optical networks. In a wavelength continuous network, channels within a link are distinguished from one another. The reader can imagine that each channel in a link has a different color. If we assume the coloring scheme is the same for all links, then a call in a wavelength continuous network must use the same color channel in each of the links it traverses; spare capacity alone is not sufficient. See Figure 2-1 for an illustration of this difference. Since we have precluded the use of wavelength changers in our model, the model is wavelength continuous, which presents additional challenges as we shall see in subsequent chapters.

[19] presents a simplistic model for all-optical demand-assigned connections which model only receiver and transmitter conflicts, with no modeling of the internal network. [20] presents a review of blocking probability and some simple simulation results

for all-optical networks.

Barry [21] has analyzed blocking probabilities in all-optical networks. This work is the closest to our work with OFS, as it uses probabilistic models to obtain analytical form solutions for blocking probability. The model used in this work uses calls in a line network similar to some of the work in this thesis. The difference is that the hop-length of calls is governed by a per-node probability of leaving or entering the network according to a prescribed distribution. This model is different from ours which uses explicit arrival processes of traffic to model flows in the network, specifically hop length. Also, the tradeoff between blocking, network utilization and delay is not addressed. Blocking probability of reconfigurable networks has been studied in [10].

2.2.2 Routing and Wavelength Assignment

Issues in all-optical networking have been studied extensively recently, especially the problem of connection routing and wavelength assignment (RWA). [14] and [13] are examples of work that address the joint RWA problem. This work is related to our work with OFS, although it does not address the issue of connection setup, and the fundamental performance issues in optical networks which are actually independent of the RWA problem. RWA has largely been studied in the context of connection-oriented networks, but is a problem that connection-less networks must obviously also solve. In general work in RWA uses limited, simplistic probabilistic models or is based solely on simulation, due to the enormity of the state space of an all-optical network. Researchers have also proposed centralized network resource allocation for wavelength routed networks [22], but this doesn't appear to be scalable.

[23] presents a scalability analysis of wavelength-routed OBS network with centralized processing of OBS requests. The analysis considers request processing and propagation time requests being served in a FIFO manner by a centralized processor. In addition, two RWA algorithms, Shortest-Path First-Fit (SP-FF) and Adaptive Unconstrained Routing Exhaustive (AUR-E). The former is claimed to be computationally simpler than the latter approach which runs a full Dijkstra algorithm computation for each incoming request. The analysis shows that with the AUR-E RWA

approach a maximum of 20-30 nodes can be supported, while for SP-FF a maximum of 35-65 nodes were supported with much poorer blocking probability. These results show a centralized approach may be scalable to a small-scale MAN network, but probably not to larger MAN and WAN networks.

Control signaling and connection setup for connection-oriented networks has also previously been addressed in the literature. This work includes [11],[9] and [12]. [9] presents Generalized Multi-Protocol Label Switching (GMPLS), which is being implemented in some newly installed next generation optical networks. GMPLS is aimed at being a router-to-router reconfiguration approach as opposed to a vehicle for user-to-user transmission. GMPLS is generally viewed as a technology that would accommodate network reconfiguration on the order of minutes. Lower bounds of minutes for link state updates have been reported in [9]. This is mainly in order to limit the overhead traffic caused by periodic flooding, in a situation with a limited control network. [27] reported analysis using simulations of update intervals as small as 0.1 seconds. The conclusion was that blocking was greatly reduced in this case, however this work ignored propagation delay and processing delay of updates, as well as control network data rate limitations. [28] reports updates that are based on *triggering* mechanisms as opposed to periodically. The basic idea here is that when a local node notices that enough changes have occurred locally, it broadcasts its new state via flooding. While this concept is more bandwidth efficient than rapid periodic updates, the paper showed that it resulted in stale network information for lightpath establishment.

2.2.3 Connection-less all-optical networks

Connection-less all-optical networks have been studied extensively for some 20 years and are based loosely on IP inter-networking. The key difference between connection-oriented and connection-less networks is that in the former, an entire end-to-end connection is constructed before transmission begins. Connection-less networks generally use hop-by-hop routing for transmission, similar to an IP packet routing paradigm [38]. Two major approaches for connectionless all-optical networks are Optical Packet

Switching (OPS) and Optical Burst Switching (OBS). OPS uses optical datagrams or packets with optical headers, and attempts to mimic IP routing in an all-optical environment. Much of this work has been dealt with only on a component or physical level so it is not relevant to our discussion of network architecture.

OBS with capture aggregates data transmissions into optical 'bursts'. These bursts are preceded by a (generally electronic) burst header that is read by the OBS routers in the network in order to allocate resources for the burst. This header is most often carried out-of-band by a high Quality of Service (QoS) control network. Most OBS approaches require traffic aggregation to sufficiently amortize the cost of burst setup and delivery. This aggregation usually occurs at an ingress router of the network so that this router's electronic subsystem does incur the cost of the burst traffic. Note that no optical resource allocation is done before transmission, rendering this approach connection-less. Both [24] and [25] outline the idea of OBS and presents some performance issues. The latter contains a taxonomy of OBS signaling protocols.

A summary of performance issues encountered by OBS investigations is as follows:

- *OBS with capture captures optical resources hop-by-hop, and therefore can suffer from resource contention blockage* - There are a number of proposed solutions to this problem. Fiber Delay Lines can be employed by OBS routers in order to delay existing bursts in the network until resources for the delayed flow can become available. Bursts can also be electronically buffered at intermediate hops and retransmitted (possibly on another wavelength), an approach to wavelength conversion. Both of these solution impose increased cost and footprint on the optical network subsystem. Rerouting is another alternative solution to this issue, but can result in network instability as shown in [25].

- *For efficiency, the offset time between the burst header and burst must be small* - This requirement can cause errors in operation of the protocol, if sufficient guard time is not provided. OBS therefore must employ thorough analysis of the route being traveled by the burst, along with associated overheads, in order

to calculate an efficient offset time. In the case of burst buffering or rerouting the offset time must be recalculated.

Much of the work in OBS and OPS including [24], [32] and [33] implicitly or explicitly assume full wavelength conversion in their analysis. In so doing, the earlier stated wavelength continuity problem disappears and the problem becomes one of link capacity on a multi-channel link, similar to circuit-switched networks discussed earlier. While this assumption simplifies analysis and enhances network performance, it ignores the fact that wavelength conversion is costly or in its experimental stages. Our work on all-optical switching will assume no wavelength conversion. [31] looks at wavelength assignment strategies for OBS networks, assuming wavelength continuity and contains some limited analysis of this situation.

A key issue in both OBS and our work is that of resource contention between all-optical transactions. Turner has proposed Terabit Burst Switching [29], which uses a small amount of look-ahead to multiplex optical bursts. Since it is connection-less, OBS is generally forced to use non-scheduled schemes to mitigate resource contention. These include deflection routing [30] and Fiber Delay Lines for optical buffering [26]. [30] shows that deflection routing can result in instability of an OBS network, and presents a realistic quantification of blocking. However, other work [32] seeks to show deflection routing as viable, although it is unclear what traffic loading regime assumptions this work shows in its results. There has been analytical work studying the performance of optical buffers for small models [34]. Fiber Delay Lines are largely in experimental stages and are not widely in use. This is often due to size footprint issues. In general, the problem of resource contention and collision between bursts has not been adequately resolved except though the use of electronic buffers or wavelength converters.

Chapter 3

ONRAMP OFS Demonstration

In this chapter, we describe the ONRAMP Optical Flow Switching Demonstration. The ONRAMP Demonstration is one of the outputs of an optical network architecture consortium between MIT, MIT Lincoln Laboratory, ATT Research and a number of other participants. This consortium was successful at demonstrating a number of network architecture ideas on state of the art (at the time) optical hardware. It focused on optical networking issues at all seven of the OSI network layers to build an integrated network capable of high QoS. The demonstration described here uses the network that was built by ONRAMP to demonstrate the viability and low cost of OFS.

Figure 3-1(a) shows the ONRAMP network testbed consists of three nodes in a two fiber ring. A passive, remotely pumped Distribution Network using Erbium-Doped Fiber Amplifiers (EDFAs) connects OFS-enabled stations, labeled Xmitter and Receiver, to the access ring. These user stations were flow-enabled in that they had two connections to the ONRAMP network, one for IP packet transactions and one for all-optical OFS transactions. In the demonstration, the Xmitter station transmits over the Bossnet [39] long-haul network which connects Washington, DC and Boston. The two fibers in the network each carry 8 1540-1560nm channels in opposite directions. Each node in the ring has an Optical Cross Connect (OXC) and a node controller/IP router. The controller stations are Alpha processor based multi-processor computers, running the Linux operating system. A dedicated 1510nm control channel connects

all controller stations in the ring. ONRAMP employs an integrated optical/IP network control strategy. The control station/router switches both control and data packets for non flow-switched communication. It also contains software to configure and manage its local OXC.

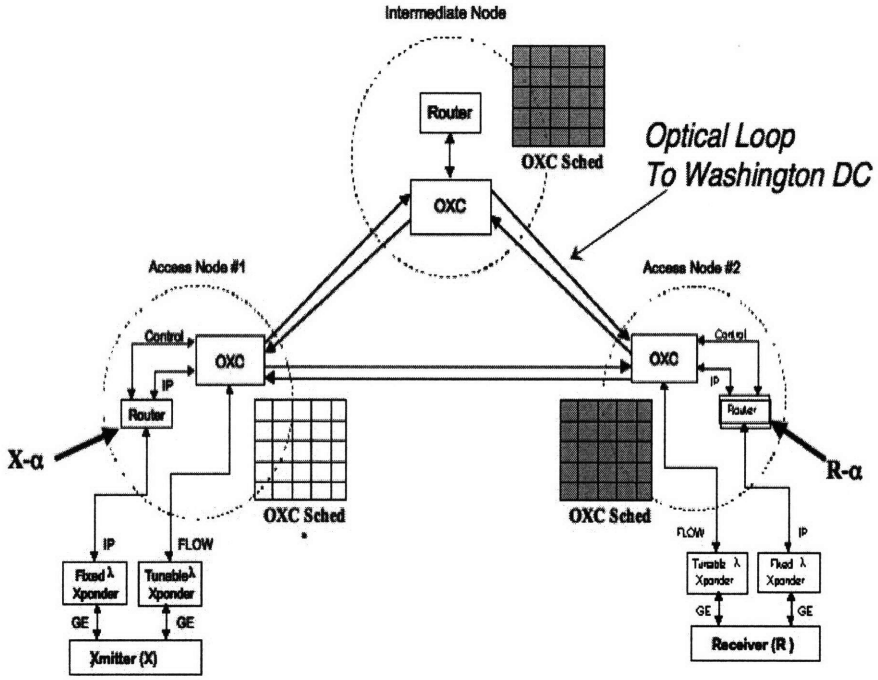
3.1 ONRAMP description

Figure 3-1(a) also shows Flow Switching Stations (FSS) in ONRAMP. Each FSS (Xmitter and Receiver) have dual connections to the network, using two Gigabit Ethernet cards. The connection labeled *IP* is fixed-tuned to a specific wavelength, and is the default communication channel for lower rate data and control packets. The network implements a simple QoS strategy using Diffserve [37] in order to prioritize control traffic in the IP subsystem. It is terminated at a port in the local controller/router. The other connection, labeled *Flow* is the wavelength-tunable flow switching connection. Flow feeds directly to the local OXC, and thereby onto the access ring. A network process runs on each FSS station to send, receive and process network control packets.

The controller stations execute Network Timing Protocol [36] to maintain a timing synchronization of clocks. This timing synchronization is accurate on the order of microseconds. NTP is a standard timing protocol and is freely available for download from the Internet. Given network timing information, many on-demand scheduled connection setup approaches are possible. As an example, we present connection setup in our current implementation in ONRAMP. (see Figures 3-1(a)(b)). Assume all control messages are sent over the IP part of the network, using appropriate QoS.

1. Flow Switching Station (Xmitter) makes a flow request to local ONRAMP Node controller (at Node 1). Assume transmission target is Receiver, and duration of transmission is known. This communication happens over the IP connection.

2. Node 1 computes the earliest timeslot with a free route, wavelength pair for Xmitters transmission (t_0). This scheduling is on-demand.



(a) ONRAMP Topology

	Slot 1	Slot 2	Slot 3	.....
λ 1	X		X	
λ 2		X		
λ 3				
λ 4	X			

(b) Example Node Schedule

Figure 3-1: ONRAMP Testbed

3. Node 1 controller communicates t_0 to Nodes 2 and 3 controllers, which update OXC schedules (see Figure 3-1(b)).

4. At time t_0, Nodes 1, 2 and 3 controllers issue commands to configure OXCs (as per their local schedule), to enable the OFS connection from Receiver and Xmitter. Receiver and Xmitters local node controllers (at Node 1 and 2 respectively) inform Receiver and Xmitter to tune to the chosen channel. Xmitter is then informed of the newly created connection.

5. Upon expiration of flow time, resources are released by the network for the next transmission. It is assumed that Receiver and Xmitter manage timing of their transmission.

This implementation maintains network state information at a central location (Node 1). However, in a more general scheme where state information is distributed, resources can be scheduled on demand using *accurate* information about the future network state. For a non-scheduled approach, these resources may have to be queried for availability at transmission time, leading to inefficiency.

The FSSs do not participate in timing synchronization (i.e. NTP), promoting scalability. The time that the network must wait between timeslots for the network to reconfigure is the *network settling time.* This is represented by the thick vertical lines in the OXC schedule, Figure 3-1(b). Settling time is overhead in a scheduled connection setup approach and should be minimized for high efficiency.

3.2 ONRAMP OFS Transport Layer

OFS data transactions are unidirectional high-rate data connections. While much of the focus of the ONRAMP design is on the physical, data link and network layers, attention must be paid to the transport protocol being used to utilize all-optical connections. Optical flows are short-duration, so the transport protocol being used must be able to rapidly utilize the bandwidth available to the flow for maximum efficiency.

As an example of this issue, we consider the ubiquitous TCP transport protocol and briefly analyze its behavior in the context of OFS. Figure 3-2(a) shows a simple network with a route between two nodes that has an OFS connection between them, beginning at time t_0. We assume the round trip time (RTT) of the network is 50ms, and that the line rate is 10Gb/s. This RTT consists only of time of flight of data, since the OFS connection has no intervening routers. We also assume that the network imposes a Maximum Transfer Unit (MTU) of 9000 bytes, meaning that the maximum packet size that can be sent is 9000 bytes. This is typical, if not a bit optimistic, for todays network.

Given an allocation of a one-second OFS transaction from the sending host (node A) to the receiving host (node B), Figure 3-2(b) shows a packet delivery timeline for TCP once the OFS connection has been opened. Readers that are unfamiliar with the basic working of TCP can refer to [35]. Recall that TCP uses a three-way handshake to establish a new connection and then uses slow-start to utilize the available bandwidth on the connection between sender and receiver. For this analysis we assume that there exists an adequate reverse path for acknowledgments, but note that data flows in only one direction.

We now analyze the time it takes for TCP to achieve full rate after opening the OFS connection at time t_0, assuming no spurious packet losses. First, the three-way handshake incurs a sender waiting time of one RTT = 50ms, since the sender must receive a "SYN/ACK" packet from the receiver before sending can begin. At this point, the connection is in slow start with a send-side window size of one packet. As can be seen in Figure 3-2(b), TCP slow start doubles the send-side window once for each RTT. It is known that maximum bandwidth utilization is achieved by TCP when the send-side window has a size equal to the bandwidth-delay product of the connection [35]. In this case, the bandwidth-delay product is $.05 \times 10 \times 10^9 = 5 \times 10^8$, in bits, or 62500000 bytes. Given the assumed MTU, this is a window size of $\frac{62500000}{9000} =$ 6945 packets. The approximate number of RTT that will be needed to achieve this window size is then equal to $log_2(6945) = 13$ RTT. In seconds, this is $.05 \times 13 = .65$. Therefore it will be almost .7 seconds before the TCP connection has achieved full

line rate. The duration of the transaction is only one second, so for a majority of the time, the line rate which OFS provides the user is under-utilized by the TCP transport layer. Note that for implementations of TCP that use linear increase for slow-start (as opposed to exponential increase as used here), this analysis provides a lower bound on the time to achieve full rate.

Because of these transport layer issues, we have chosen to use the connection-less UDP datagram delivery protocol. We have implemented a simple but effective *send-side rate limited* transport layer implementation. This implementation allows these short-lived transactions to fully utilize the Gigabit Ethernet link layer, achieving the theoretical maximum of 989Mb/s.

There are a number of advantages of this network layer over a two-way connection-based approach (e.g. TCP) First, this transport implementation matches the maximum send rate of the link layer immediately, as opposed to an approach such as TCP slow start, as discussed above. This is especially important for short duration connections. Second, there is no need for a highly available, low delay reverse path such as that needed for TCP acknowledgments. This avoids the need to allocate a dedicated path for protocol feedback, and averts the risk of loss of acknowledgments in a feedback path such as the Internet. Finally, this approach takes advantage of the low bit error (and therefore low packet error) rates provided by optical connections. It assumes that the majority of transactions will be error free, so that there is no need for an active feedback loop, when using OFS. Note that flows still require an acknowledgment of a completed transaction.

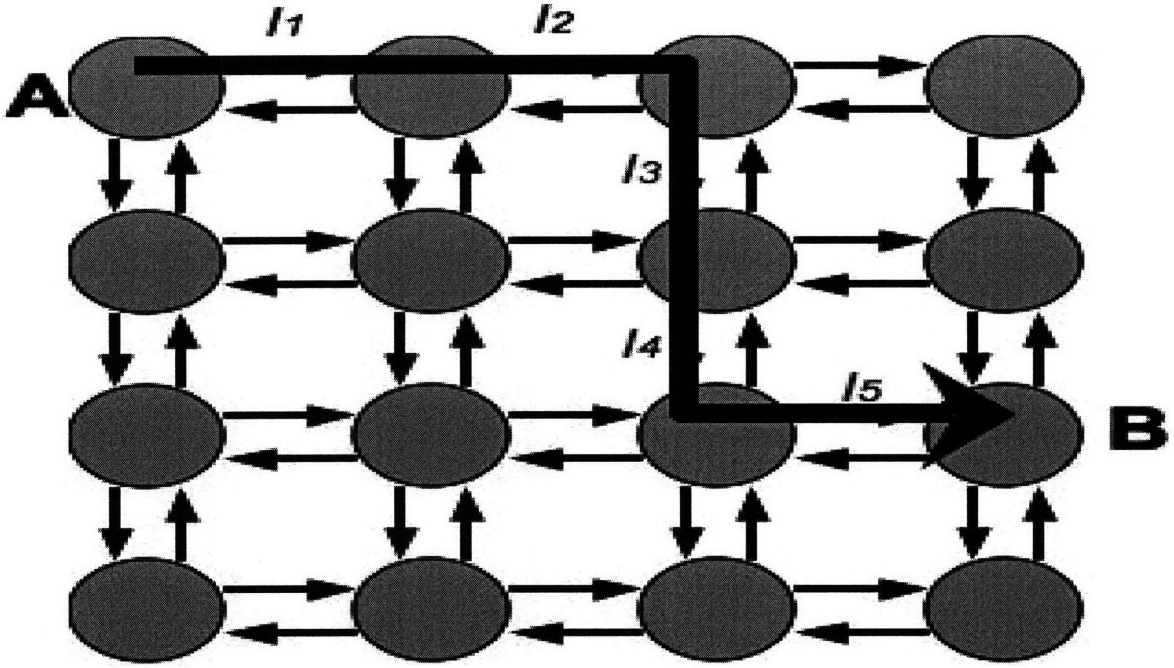
3.3 ONRAMP Demonstration Results

The performance results of our demonstration are shown in Figures 3-3(a)(b)(c). These results are for more than 200 single timeslot transactions, sent over the entire channel space in both directions of the ring from Xmitter to Receiver (see Figure 3-1(a)). Figure 3-3(a) shows that the scheduled connection setup achieved very high bit rates for the multiplexed flows, close to the theoretical maximum. Figure 3-3(b),

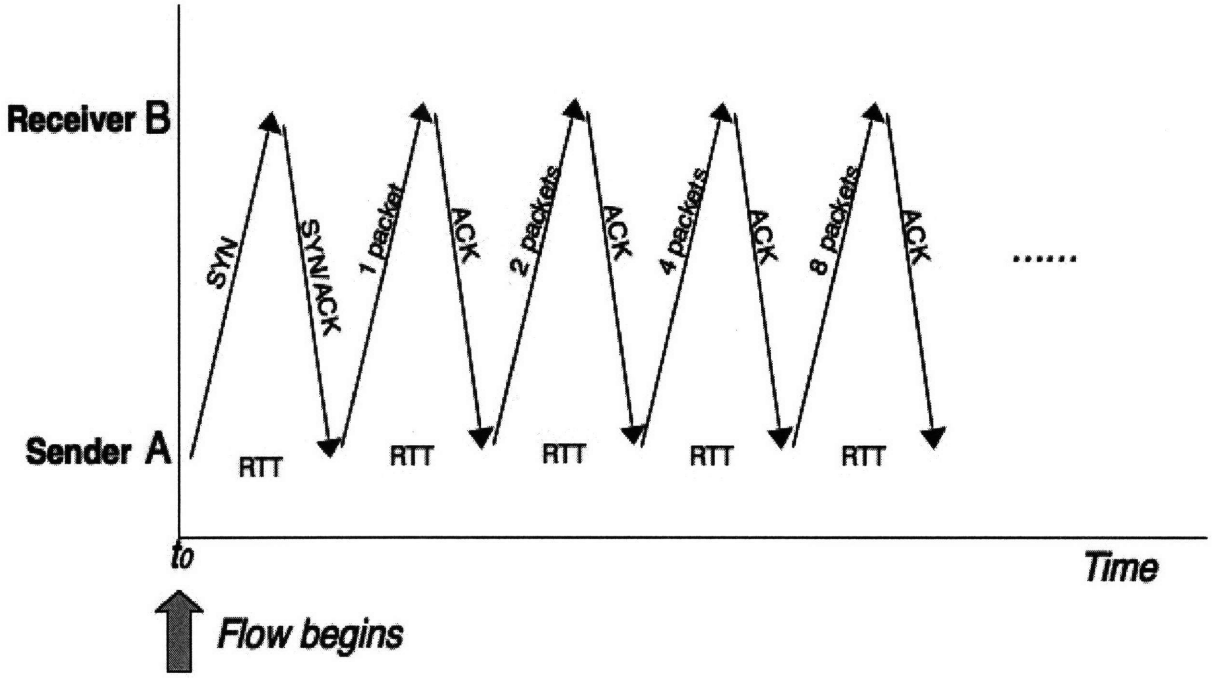
shows that for one-second flows, we achieve 90% efficiency, though we are changing routing and wavelength assignment on a second-by-second basis. Figure 3-3(c) shows that the limiting factor for performance for 0.5 second timeslots was the corruption of flows. We believe this is due to unresponsiveness of the link layer to reconfiguration and wavelength switching.

Figure 3-4(a) shows measured sources of overhead that were incurred by the hardware and software. Reduction of these overheads would increase the efficiency of OFS in ONRAMP, and also make the flow granularity flexible. In total, there was about .1 seconds of overhead measured experimentally, and this corresponds to the 90% efficiency seen by one-second flows.

Finally, an important output of the ONRAMP experiment is that it was implemented using commercially available hardware and software. The Alpha machines and Altitun lasers were stock, and not specialized. In addition the control plane for ONRAMP was implemented in standard Linux, with little modification to the operating system kernel. We consider it a promising result that OFS was made efficient using non-specialized hardware and non-real time software.

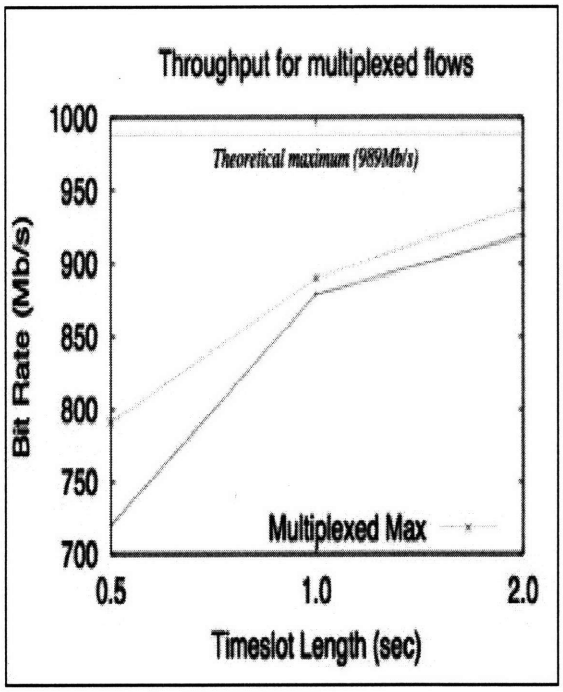


(a) OFS connection from node A to B at time t_0

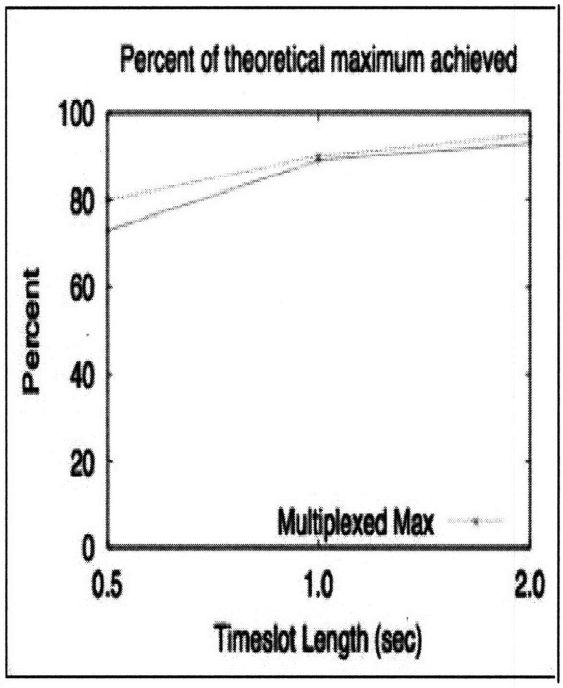


(b) TCP data packet delivery timeline

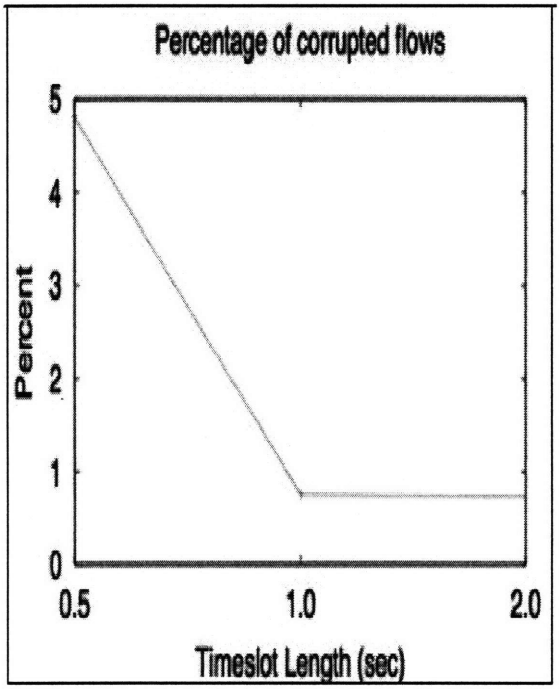
Figure 3-2: Illustration of TCP transport protocol in context of OFS



(a) Throughput

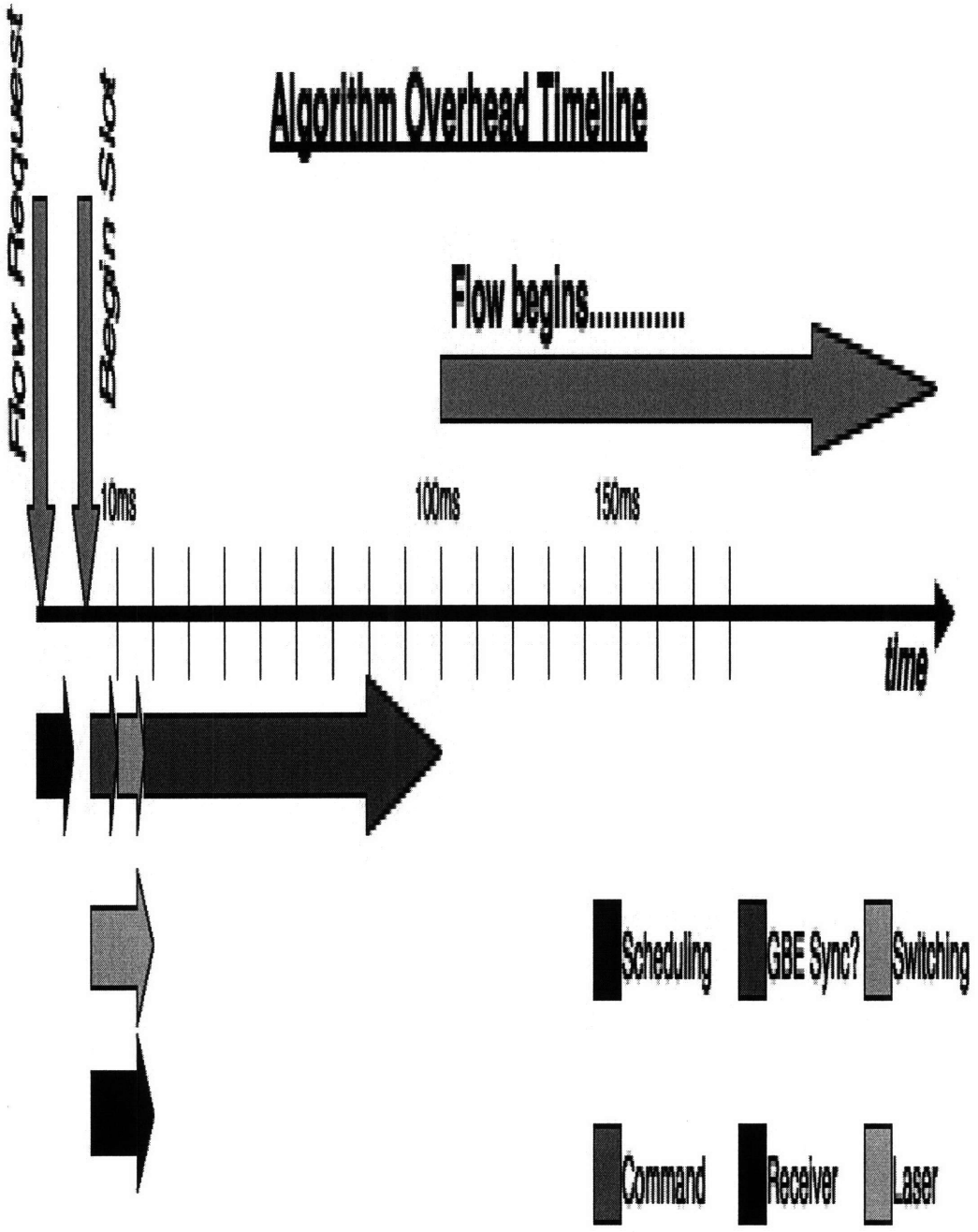


(b) Pct. of max. throughput



(c) Percent of flows corrupted

Figure 3-3: Demonstration Results



(a) Sources of overhead

Figure 3-4: ONRAMP OFS Demonstration Overhead Analysis

Chapter 4

Optical Network Control Plane Analysis

4.1 Introduction

In this chapter, we present transient analysis of control plane strategies for all-optical connection setup. The issue we address in these studies is that of *timescale*. The unique feature of wavelength continuous all-optical networks we study is that they are highly dynamic and that it contains no buffering internal to the network to handle resource contention. In addition, reasonable arrival rates of traffic can cause the network state of even small networks to change rapidly. The question we explore is whether certain control strategies can handle the small timescales (i.e. dynamic nature) of the all-optical network, such as an OFS network.

The control plane of the network is responsible for receiving user requests and making decisions about resource allocation. It is therefore burdened with having knowledge of the network state in order to make judicious choices in route and wavelength selection for user connections. Intuitively speaking, if the network state information is incorrect, the control plane has little basis to make choices about what resources to allocate to connection requests.

The analysis that follows investigates two currently suggested connection setup/network state information dissemination strategies. The analysis shows that the timescale is-

sue does indeed apply to these approaches and that there needs to be further study and invention of strategies to make network state information more useful.

4.2 Transient Analysis of Update-Based Optical Connection Setup

GMPLS [9],[28],[27] is a widely discussed approach to network architecture for dynamic all-optical connection setup in a MAN or WAN. It is an example of an *update-based* approach to network state information dissemination. This approach, among others, proposes setup based on network state information obtained from periodic broadcast updates. We analyze an all-optical network model employing an update-based control strategy, examining implications of this for connections of short duration ($\leq$ 1 second) and high arrival rate (i.e. high utilization). We have designed a network model that allows a transient behavior analysis. Our results show a timescale mismatch for the update based approach with a highly dynamic optical network, an example of which is an OFS network.

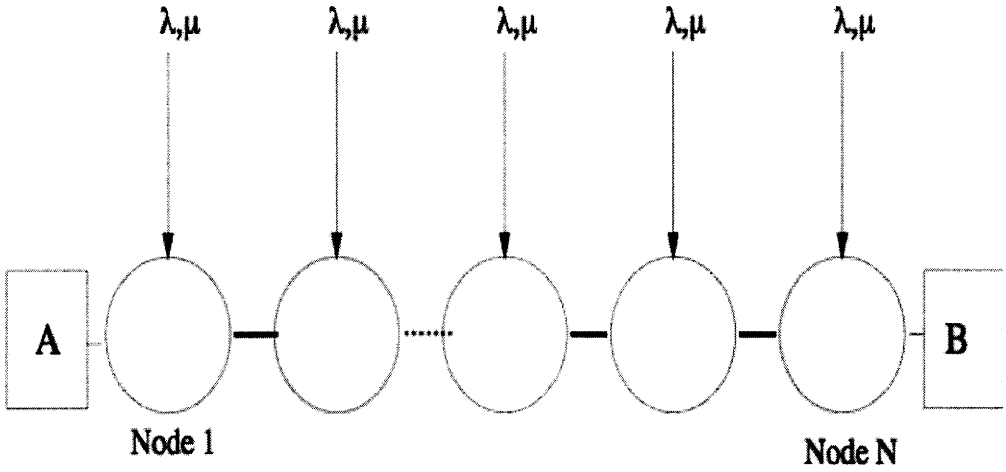
4.2.1 Motivation

An all-optical network must make choices for routing and wavelength assignment for all-optical connection setup. In GMPLS, these choices are based on a view of the state of network resources provided by periodic broadcast updates. These updates are sent by each optical node to every other in the network, and list current availability of local optical links. When a request arrives at a node for a connection, the node uses the information provided by the most recent valid updates to make a routing and wavelength assignment decision.

The issue we address is that the information provided by the updates may be *stale* by the time the information is needed or used. This is primarily due to the periodic nature of the updates, coupled with the dynamic nature of all-optical networks. Note that latency of control messages may also play a role in this but we idealize the

latency to be zero here. The question we address is: given a periodic update at a particular time, how soon does the information become stale, and therefore incapable of informing resource decisions?

4.2.2 Methodology



(a) Line Network with cross-traffic

Figure 4-1: Network for Analysis

Figure 4-1 shows the network model. The network has N intermediate nodes arranged in a line topology, with end-nodes A and B. We assume a single optical channel in the network. Cross traffic arrives to the intermediate nodes, and occupies them. We analyze the network under the following assumptions.

- A receives and instantaneous update from all nodes at time $t_0 = 0$, this update shows the all-optical path from A to B as clear.
- At some time in the future, A desires to send to B. We assume that during the course of this analysis, no other traffic is incident at A besides this flow. This is an idealization, where the traffic originating from A to B is not intense enough to incur more than one arrival in an update period. The problem of having a

stochastic arrival process at A for B involves steady-state analysis, and is not addressed by this particular model.

- Cross-traffic arrives at the intermediate nodes with an exponentially distributed interarrival time with parameter λ and a either an exponentially or Pareto distributed holding time of average μ. This is an idealization of a route inside a mesh network, since the arrival rate to the nodes in an actual route will not be Poisson.

- All cross-traffic has length 1, that is, any active cross-traffic occupies exactly one node, the node it arrived to.

- Presence of active cross traffic at any node results in blocking of the A $->$ B connection.

- We consider up to two arrivals of cross traffic per node (and therefore up to two possible subsequent departures). These two arrivals are modeled, although 0, 1 or both may arrive by the prescribed time t. We consider further arrivals during an update interval to be a second order effect, if t is small.

The problem formulation is as follows:

$$N : \text{Number of intermediate nodes}$$

$$\mu : \text{Service rate of all customers}$$

$$t_0 = 0 : \text{Time origin}$$

$$t : \text{Time of A to B send (free parameter) } (t > t_0)$$

$$\lambda : \text{Arrival rate of customers at all intermediate nodes}$$

Denote the state of the system at time t as:

$$\Phi(t) = (\phi_1(t) \quad \phi_2(t) \quad ... \quad \phi_N(t)) : \phi_i(t) = 1 \text{ if } \exists \text{ active call at node i at time } t, \ 0 \text{ otherwise}$$

Then blocking probability for this model is defined as:

$$P_b = P(\exists i \text{ s.t. } \phi_i(t) = 1 | \Phi(t_0) = 0)$$

This is the probability that there is an active customer in the system at time t, given an empty system at time t_0. Note that we assume no customers arrive at A in the interval $[0, t]$.

For example, for a 5 node network, if $\Phi(t)$ for some time t is:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

... this means that there are active calls at nodes 2 and 5, and therefore blocking for the A to B call at time t.

Each component of $\Phi(t)$ evolves as an M/M/1/1 queue, given the single channel assumption. Further, because each customer occupies exactly one node, the nodes are an independent set of queues. Stationary distributions and long term utilization results for these queues are well known. However, our interest here is in *transient analysis*, over a finite amount of time from t_0 to t.

Given independence of the queues, blocking is a union of independent events, as in:

$$P_b = P(\bigcup_{1 \leq i \leq N} \phi_i(t) = 1 | \Phi(t_0) = 0)$$

Again, we assume that no traffic arrives from A destined for B in the interval $[0, t]$ which is a reasonable assumption if the traffic intensity at A is low.

Lower Bound on P_b

The node arrival processes (i.e. the Poisson process) possess the property that there is a non-zero probability of any number of arrivals even for a finite amount of time [40]. In order to perform transient analysis, we limit the maximum number of arrivals considered per node to be 2. This restriction results in a lower bound on the

probability of node occupancy or equivalently $P(\phi_i(t) = 1)$. Since P_b depends on the occupancy of each node, this, in turn, admits a lower bound on blocking probability. To see this, and analyze the closeness of this lower bound, assume $t_0 = 0$. Then, for any time t, and node k, we can write:

$$P(\phi_k(t) = 1) = P(\phi_k(t) = 1|1 \text{ arrival in } [0,t)]P(1 \text{ arrival in } [0,t])+$$

$$...P(\phi_k(t) = 1|2 \text{ arrival in } [0,t])P(|2 \text{ arrival in } [0,t]) + ...$$

$$...P(\phi_k(t) = 1|\text{greater than 2 arrival in } [0,t])P(\text{greater than 2 arrival in } [0,t])$$

Removing the third term in the above sum, we get a lower bound on $P(\phi_k(t) = 1)$, involving either 1 or 2 arrivals. This is the analysis we perform in this study.

An upper bound on the third term is found by assuming that:

$$P(\phi_k(t) = 1|\text{greater than 2 arrival in } [0,t]) = 1$$

That is, if more than 2 arrivals occur by time t, occupancy at time t is certain. The unconditional probability $P(\text{greater than 2 arrival in } [0,t])$ is readily calculated, and results in the maximum possible disparity between $P(\phi_k(t) = 1)$ and our lower bound. A graph of this probability is shown below, for one node with various arrival rates and various t.

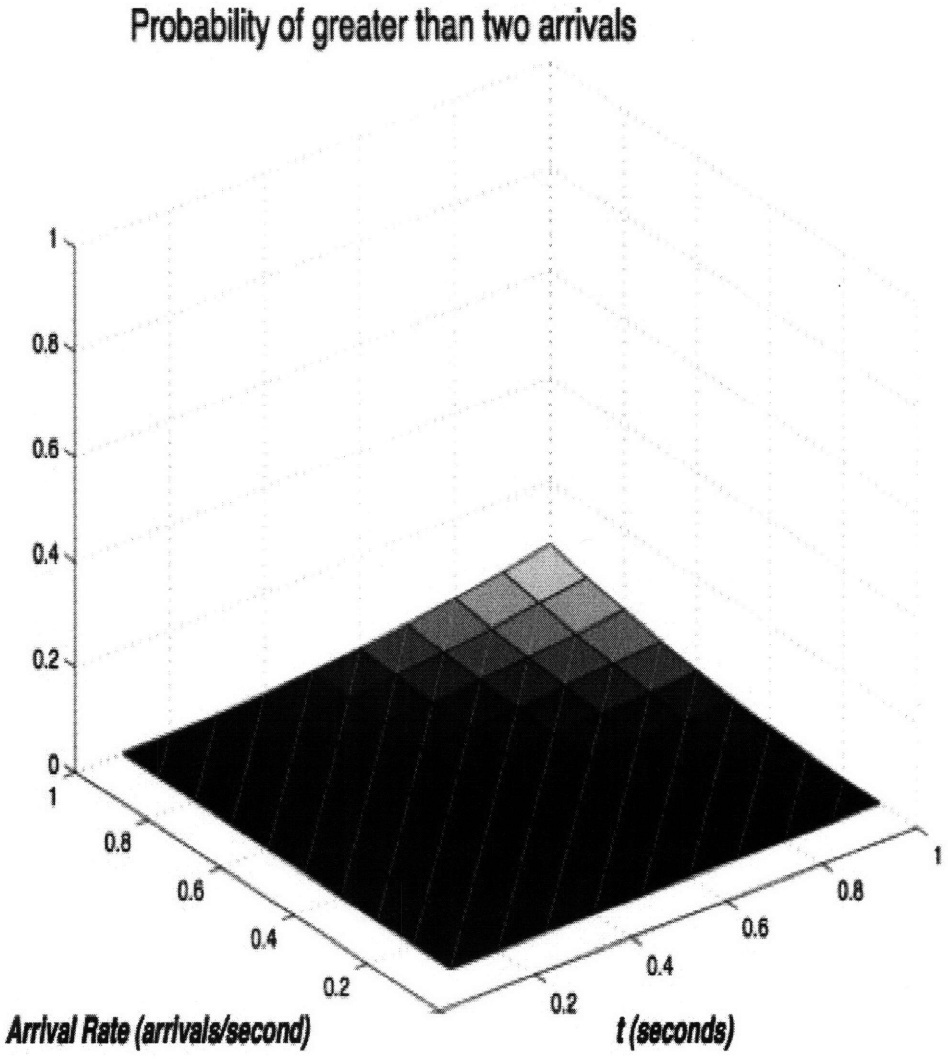
The Figure 4-2 shows that over all cases, we underestimate $P(\phi_k(t) = 1)$ for a given node by at most 0.1.

We can now define:

$$P^{LB}(\phi_k(t) = 1) = P(\Phi_k(t) = 1|1 \text{ arrival in } [0,t])P(1 \text{ arrival in } [0,t])+$$

$$...P(\phi_k(t) = 1|2 \text{ arrival in } [0,t])P(|2 \text{ arr in } [0,t]) \leq P(\phi_k(t) = 1)$$

Blocking occurs when any node is occupied at time t, i.e. $\phi_k(t) = 1$ for some $1 \leq k \leq N$. The overall blocking probability P_b is the probability of this event. The



(a) Probability of more than 2 arrivals, given node

Figure 4-2: Probability of more than 2 arrivals for various intervals, arrival rates

formula for $P^{LB}(\phi_k(t) = 1$ given directly above is a lower bound on the probability that a node is occupied. We can therefore define a lower bound on the blocking probability as a union of the events corresponding to P^{LB}, and therefore a lower bound on P_b as follows:

$$P_b^{LB} = P^{LB}(\bigcup_{1 \leq i \leq N} \phi_i(t) = 1 | \Phi(t_0) = 0) \leq P_b$$

We begin with a an approximate analysis of P_b^{LB}, to establish a useful upper bound on blocking probability. To proceed, assume that $\mu >> t$. Then we can assume that *any* arrival to any intermediate node will block the $A- > B$ transmission. This is clear, since any arrival to a node will, with high probability, have duration much longer than t and will therefore occupy a node at the send time of the $A- > B$ transmission. We know that the probability of zero arrivals Pr(0 arrivals at node i) in $[0, t]$ for a Poisson process of rate λ, at node i is:

$$\text{Pr(0 arrivals at node i)} = e^{-\lambda t}$$

The nodes are defined to be independent, and therefore the probability of zero arrivals at all N nodes Pr(0 arrivals to the system) is:

$$\text{Pr(0 arrivals to the system)} = \text{Pr(0 arrivals at node i)}^N = (e^{-\lambda t})^N$$

Finally, the probability of one or more arrivals at any node during $[0, t]$ is:

$$\text{Pr(an arrival to the system)} = 1 - \text{Pr(0 arrivals to the system)} = 1 - (e^{-\lambda N t})$$

This gives an upper bound on the blocking probability since some sessions may depart the system before t, which we expect to resemble the more detailed transient analysis which follows. We will include curves for this analysis in result graphs for this subsection as a point of comparison, under the label *Upper Bound.*

Transient Analysis of P_b^{LB}

Given an assumption of a maximum of two arrivals at a node in the interval $[t_0, t]$, we can now proceed with analysis of P_b^{LB} the system, which we will henceforth call the blocking probability of the system.

From before, we have:

$$P_b^{LB} = P^{LB}(\bigcup_{1 \leq i \leq N} \phi_i(t) = 1 | \Phi(t_0) = 0)$$

In order to complete the analysis it remains to compute $P^{LB}(\phi_i(t) = 1)$ for a node i, given a maximum of two arrivals at node i.

We can analyze the situation as follows, and define independent random variables:

$$X_1 : \text{ inter-arrival time of first arrival to i, calculated from time } t_0$$

$$Y_1 : \text{ duration of first arrival to i}$$

$$X_2 : \text{ inter-arrival time of second arrival to i}$$

$$Y_2 : \text{ duration of first arrival to i}$$

The PDFs for these variables are (assume the durations are exponentially distributed):

$$p_{X_1}(x) = p_{X_2}(x) = \lambda e^{-\lambda x} \quad (x > 0) \qquad p_{Y_1}(y) = p_{Y_2}(y) = \mu e^{-\mu y} (y > 0)$$

Also it is important to note that since the four variables are mutually independent, the joint PDF is the multiplication of the four marginals. Blocking occurs in two cases, it can either be due to the first arrival or the second. Analysis of the situation finds that $P(\phi_i(t) = 1 | \phi_i(t_0) = 0)$ is a disjoint union of the following two events:

$$((X_1 < t) \wedge (X_1 + Y_1 > t)) \quad (1)$$

$$((X_1 + Y_1 < t) \wedge (Y_1 < X_2) \wedge (X_1 + X_2 < t) \wedge (X_1 + X_2 + Y_2 > t)) \quad (2)$$

(1) is the event that the transmission is blocked by the first arrival and (2) the second. First, note that the two expressions are disjoint due to the opposite conditions on $(X_1 + Y_1)$ with respect to t. Thus, we can calculate the probabilities of the two events separately and sum them for the final result.

In (1), the first term $(X_1 < t)$ ensures that the first arrival happens before t, and the second term $(X_1 + Y_1 > t)$ forces the departure of that arrival to be after t, hence blocking. In (2), the first term $(X_1 + Y_1 < t)$ ensures that the first arrival departed before t. The second term $(Y_1 < X_2)$ ensures that the second arrival happened after the first departure, because if otherwise, the second arrival would be lost (loss network). The third term $(X_1 + X_2 < t)$ ensures that the second arrival happened before t, and finally the last term $(X_1 + X_2 + Y_2 > t)$ ensures that the second departure happened after t, hence blocking.

We analyze a closed form solution for both (1) and (2) involving multiple integrations of the joint PDFs, which are known. Analysis shows that the integral for (1) is:

$$\int_0^t \int_{t-x_1}^{\infty} p(x_1, y_1) dy_1 dx_1$$

The limits of the above integral come from the terms in (1). We proceed with a detailed explanation of them from the inner integral of the expression out.

- *Limit:* $t - x_1$ *to* ∞ *for the integral over* y_1 - y_1, the duration of the first arrival makes the second term of expression (1) true when the sum $x_1 + y_1$ is greater than the time t, forcing the first arrival to be active at time t. The maximum duration of y_1 is unlimited so the upper limit of the integration is ∞

- *Limit:* 0 *to* t *for the integral over* x_1 - x_1, the interarrival time of the first arrival must be shorter than the time t, in order to make the first term in (1) true.

Therefore the limits are $[0, t]$

When evaluated when $\mu, \lambda = 1$ this yields:

$$t(e^{-t}) \quad t > 0$$

The integral for (2) is obviously more complicated:

$$\int_0^{\infty} \int_0^{t-y_1} \int_0^{\infty} \int_{max(y_1, t-y_2-x_1)}^{t-x_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1 \qquad (*)$$

A discussion of the limits of the integration follows, proceeding form the innermost integral to the outermost:

- *Limit:* $max(y_1, t-y_2-x_1)$ *to* $t-x_1$ *for the integral over* x_2 - The second term of (2) stipulates that x_2 must be greater than y_1 to ensure a true result, meaning that the interarrival of the second arrival must be greater than the duration of first arrival, because otherwise, the second arrival would be blocked. The fourth term of (2) also stipulates that the sum $x_2 + y_2 + x_1$ must be greater than t, as this ensures that the second arrival is active at time t. These two conditions must simultaneously be met, resulting in the expression $max(y_1, t-y_2-x_1)$ for the lower bound of this integral. Resolution of this max function is discussed immediately below. The third term of (2) imposes an upper limit on the integral stating that the sum $x_1 + x_2$ must be less than t, in other words, both arrivals have occurred by time t.

- *Limit:* 0 *to* ∞ *for the integral over* y_2 - (2) imposes no explicit limitation on y_2 in this integration order, so it can take on any non-negative value.

- *Limit:* 0 *to* $t - y_1$ *for the integral over* x_1 - The first term of (2) shows that x_1, the interarrival time of the first arrival must be less $t - y_1$.

- *Limit:* 0 *to* ∞ *for the integral over* y_1 - (2) imposes no explicit limitation on y_2 in this integration order, so it can take on any non-negative value.

The order of integration was chosen for simplicity of analysis in this case. The inner integral involves a max function, which must be resolved into the sum of two integrals over separate ranges. This results in the sum of the following two integrals:

$$\int_0^t \int_0^{t-y_1} \int_{t-y_1-x_1}^{\infty} \int_{y_1}^{t-x_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1 \qquad (3)$$

$$\int_0^t \int_0^{t-y_1} \int_0^{t-y_1-x_1} \int_{t-y_2-x_1}^{t-x_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1 \qquad (4)$$

In order to resolve the max function in the integral, consider two cases: First, assume that $y_1 > t - y_2 - x_1$, then the max function resolves to y_1 and the integral (3) is the result. Note that for the integral for y_2 we have changed the limits since we now have the condition $y_2 > t - y_1 - x_1$ from the resolution of the max function. The second case is where $y_1 < t - y_2 - x_1$, and this results in the max function being resolved as in expression (4). Again, we have changed the limits of the integral for y_2 since we now have the condition $y_2 < t - y_1 - x_1$ from the max function resolution assumption.

It remains to show that the lower limits of all the integrals are positive. Since all variables have exponential PDF, the only lower limits in the integrals that need analysis are $t - y_1 - x_1$ in the x_1 integral in (3), and the limit $t - y_2 - x_1$ in the x_2 integral in (4). First consider $t - y_1 - x_1 = t - (y_1 + x_1)$ in (3). In the model, this quantity is the send time minus the sum of the interarrival and service time of the first arrival. By construction of the event (2) this quantity must be greater than 0, since t must be greater than the sum, otherwise, blocking by the second arrival would not occur. Now consider the limit $t - y_2 - x_1$ in (4). From the integral for y_2 in (4), the maximum value that y_2 can take on is $t - y_1 - x_1$, which was just shown to be positive. Then the limit in (4) has a minimum value of $t - (t - y_1 - x_1) - x_1 = y_1$. We know that y_1 is positive, so this ensures positivity of the limit.

Exponential PDF for cross traffic durations

We use MATLAB symbolic integration to produce the expressions for these integrals. MATLAB uses the Maple application to perform the symbolic integration of multi-variable expressions with symbolic limits. The analysis results in the following two expressions, with λ, μ, t as as symbols. The expression that results from integrating (3) is:

$$\frac{\mu}{\lambda}(-3e^{-(\lambda+\mu)t} - \lambda te^{-(\lambda+\mu)t} + 3e^{-\mu t} + \frac{1}{2}\lambda^2 t^2 e^{-\mu t} - 2\lambda te^{-\mu t})$$

The expression that results from integrating (4) is:

$$\frac{\mu}{\lambda}(2e^{-(\lambda+\mu)t} + \lambda te^{-(\lambda+\mu)t} - 2e^{-\mu t} + \lambda te^{-\mu t})$$

It is instructive to examine the limits as t is varied. Examining these expressions we analyze the limits. For a given λ, μ both expressions (and hence, the sum) go to zero as t approaches 1 and ∞. This makes sense as blocking by the second arrival is very unlikely for an update interval that is very short (first arrival, or no arrival will block system), or very long update intervals (both arrivals will have arrived and been serviced).

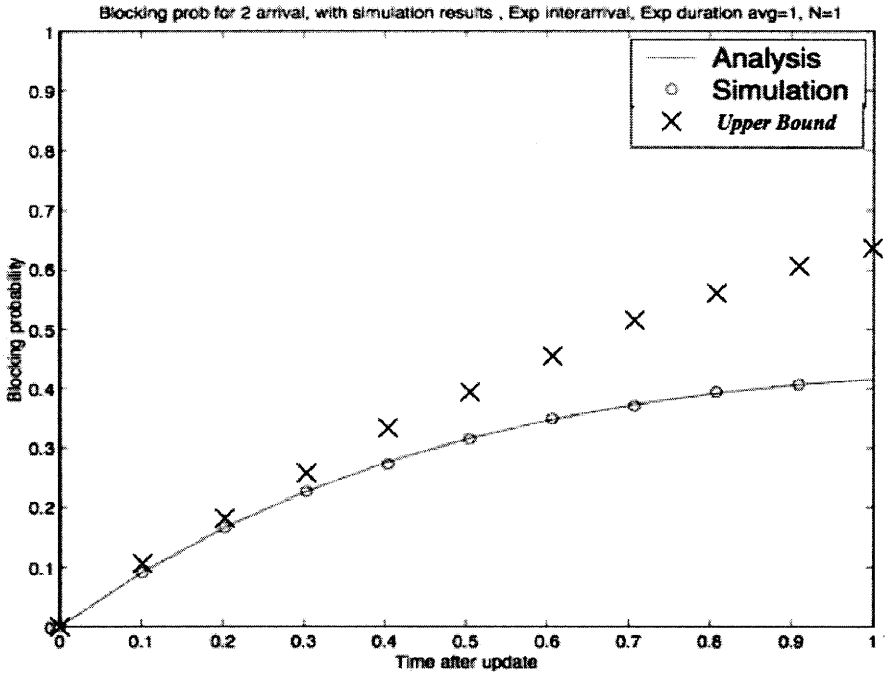
Evaluating the two expressions yield the following two expressions with $\mu, \lambda = 1$, respectively for (3) and (4).

$$-3e^{-2t} - te^{-2t} + 3e^{-t} + \frac{1}{2}t^2 e^{-t} - 2te^{-t}$$

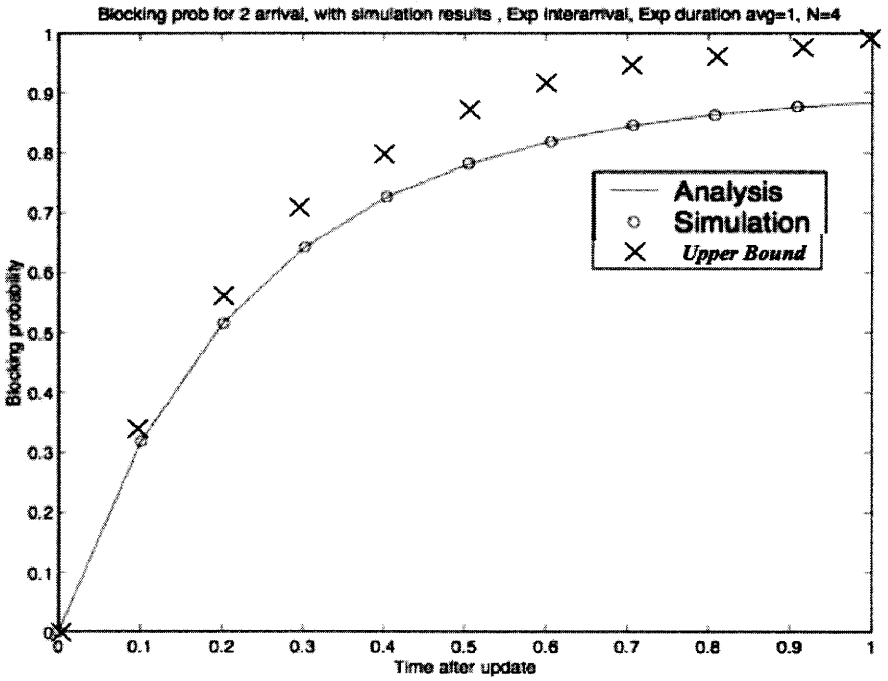
$$2e^{-2t} + te^{-2t} - 2e^{-t} + te^{-t}$$

Summing these two expressions for (3) and (4) along with the expression resulting from (1) yields the following combined expression for $P^{LB}(\phi_i(t) = 1)$, which we graph below:

$$\frac{1}{2}t^2 e^{-t} - e^{-2t} + e^{-t}$$



(a) 1 second Exp. duration, Exp. interarrival, N=1



(b) 1 second Exp. duration, Exp. interarrival, N=4

Figure 4-3: Blocking probability for exponential duration

For both exponential and Pareto distributions, we have written a simulation of the transient analysis to verify the results of the analysis. The simulation models nodes A, B, intermediate nodes, and models cross traffic and arbitrary sends as prescribed by the descriptions above. For interarrival times and durations for the cross traffic are drawn from MATLAB implementations of the appropriate distributions. The simulation was run for tens of thousands of iterations until the statistic in question, in this case, blocking probability, had stabilized, not varying more than 10^{-4} after a few thousand more iterations. We expect the simulation results to match that of the analysis in all cases.

Figures 4-3(a)(b) show results for $\mu, \lambda = 1$. The four node results in Figure 4-3(b) were calculated using the formula for the union of independent events as shown above. The upper bound curve in both of the figures show the rough upper bound we described in the analysis. This has the benefit of bracketing the blocking probability with an upper bound and lower bound. Notice that the upper bound becomes less tight in both cases as t increases. This is to be expected, since a longer t means that there are more dynamics/arrivals that can happen after the update. Nevertheless the upper bound tracks the behavior and shape of the lower bound closely, and is a useful result. Overall, the results show that the blocking probability lower bound is very significant, even for low arrival rates.

Pareto PDF for durations

If the durations Y_1, Y_2 have pareto (heavy-tailed) distributions, we must alter the above integrals to reflect this. Recall that the PDF for a pareto distribution has the form:

$$P(x) = \alpha\, k^{\alpha}\, x^{-(\alpha+1)} \qquad k > 0, \alpha \geq 1, x \geq k$$

In particular, the domain of the distribution is from $k\, to\, \infty$. Therefore, care must be taken so that the limits of the integrals reflect this restriction. In the case of single arrival blocking, the integral for (1) remains the same as long as $t > k$. Case (2) (the

integral (∗)) changes considerably as the two integrals (3) and (4) have new limits, although we have resolved the max function in the same way as for the Exponential PDF. This results in the two integrals (3a) and (3b), which we analyze below:

$$\int_k^t \int_0^{t-y_1} \int_{max(k,t-y_1-x_1)}^{\infty} \int_{y_1}^{t-x_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1 \qquad (3a)$$

We present the limit analysis for the integral (3a), proceeding from the innermost integral outward. This integral results from the max function in (∗) resolved as $y_1 > t - y_2 - x_1$:

- *Limit:* y_1 *to* $t - y_1$ *for the integral for* x_2 - x_2, interarrival time of the second arrival must be greater than y_1 by the second term of (2) and it must be greater than $t - x_1$ by the first term of (2).

- *Limit:* $max(k, t - y1 - x1)$ *to* ∞ *for the integral for* y_2 - The duration of the second arrival y_2 must be greater than k, the Pareto PDF domain lower limit, otherwise it cannot block at time t. It must also be greater than $t - y_1 - x_1$ by the assumption that we have made for integral (3a) with respect to the max function in (∗). The second arrival duration can be as long as the distribution will allow or a limit of ∞.

- *Limit:* 0 *to* $t - y_1$ *for the integral for* x_1 - The first and third terms of (2) show that x_1, the interarrival time of the first arrival must be less than $t - y_1$.

- *Limit:* k *to* t *for the integral for* y_1 - Given that the above conditions are met, y_1, the duration of the first arrival can be any positive value less than t as stipulated by the first term of (2). In this case the lower limit of the Pareto PDF is k, so this is the lower limit of this integral.

$$\int_k^t \int_0^{t-y_1} \int_k^{max(k,t-y_1-x_1)} \int_{t-y_2-x_1}^{t-x_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1 \qquad (4a)$$

We present the limit analysis for the integral $(4a)$, proceeding from the innermost integral outward. This integral results from the max function in $(*)$ resolved as $y_1 < t - y_2 - x_1$:

- *Limit:* $t - y_2 - x_1$ *to* $t - x_1$ *for the integral of* x_2 - The lower limit comes from the fourth term of (2), as well as the resolution of the max function described above. The upper limit comes from the first term of (2) which says that x_2 must be greater than $t - x_1$.

- *Limit:* k *to* $max(k, t - y_1 - x_1)$ *for the integral of* y_2 - The lower limit on y_2 is k, which is the lower limit of the Pareto PDF. The upper limit comes from the need for the duration to be at least k, if not as large as the limit imposed by the resolution of the max function from $(*)$, or $y_1 < t - y_2 - x_1$.

- *Limit:* 0 *to* $t - y_1$ *for the integral of* x_1 - The lower limit on the interarrival time of the first arrival is obviously 0, the upper limit comes from the first term of (2).

- *Limit:* k *to* t *for the integral of* y_1 - The lower limit k is the lower bound on the domain of the Pareto PDF, the upper limit is t, since the duration of the first arrival cannot be larger than t, since it cannot cause blocking in the case of (2).

Resolving of the max functions in the same manner as above results in sum of four integrals. We continue to assume $t > k$. We explain the origins of each integral in turn as they relate to either (3a) or (4a).

First, it is useful to write down the two integrals that result from resolution of the max function in (3a). We will label them (3a*) and (3a**)

$$\int_k^t \int_{max(0,t-k-y_1)}^{t-y_1} \int_k^{\infty} \int_{y_1}^{t-x_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1 \qquad (3a^*)$$

This integral comes from resolving the max function in (3a) as $k > t - y_1 - x_1$. This implies that $x_1 > t - k - y_1$, as reflected by the lower limit for the integral for x_1 in (3a*). The max function is introduced to ensure positivity of the lower limit. We therefore need to further resolve (3a*) into two more integrals as follows:

$$\int_{t-k}^{t}\int_{0}^{t-y_1}\int_{k}^{\infty}\int_{y_1}^{t-x_1} p(x_1,x_2,y_1,y_2)dx_2dy_2dx_1dy_1$$

Here we have resolved the max function in (3a*) to say that $0 > t-k-y_1$, implying that $y_1 > t-k$. This restriction is reflected in the lower limit of the integral for y_1. Note that we are currently assume that $t > k$ so $t-k$ is guaranteed to be positive.

$$\int_{k}^{t}\int_{t-k-y_1}^{t-y_1}\int_{k}^{\infty}\int_{y_1}^{t-x_1} p(x_1,x_2,y_1,y_2)dx_2dy_2dx_1dy_1$$

Here we have resolved the max function in (3a*) to say that $0 < t-k-y_1$, implying that $y_1 < t-k$. This restriction is reflected in the upper limit of the integral for y_1.

The second integral from (3a) is (3a**):

$$\int_{k}^{t}\int_{0}^{t-k-y_1}\int_{t-y_1-x_1}^{\infty}\int_{y_1}^{t-x_1} p(x_1,x_2,y_1,y_2)dx_2dy_2dx_1dy_1 \qquad (3a^{**})$$

This integral comes from resolving the max function as $k < t-y_1-x_1$. This implies that $x_1 < t-k-y_1$, as reflected in the upper limit in the integral for x_1 in (3a**). This restriction also implies that $y_1 < t-k-x_1$. Since we know x_1 is a positive value from its PDF, we can conclude that $y_1 < t-k$, and this is reflected in the upper limit in the integral for y_1.

This analysis of (3a) results in a sum of the integrals:

$$\int_{t-k}^{t}\int_{0}^{t-y_1}\int_{k}^{\infty}\int_{y_1}^{t-x_1} p(x_1,x_2,y_1,y_2)dx_2dy_2dx_1dy_1$$

$$\int_{k}^{t-k}\int_{t-k-y_1}^{t-y_1}\int_{k}^{\infty}\int_{y_1}^{t-x_1} p(x_1,x_2,y_1,y_2)dx_2dy_2dx_1dy_1$$

$$\int_{k}^{t-k}\int_{0}^{t-k-y_1}\int_{t-k-y_1}^{\infty}\int_{y_1}^{t-x_1} p(x_1,x_2,y_1,y_2)dx_2dy_2dx_1dy_1$$

Analysis of (4a) results in one more integral, with derivation below:

$$\int_k^t \int_0^{t-k-y_1} \int_k^{t-k-y_1} \int_{t-y_2-x_1}^{t-x_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1$$

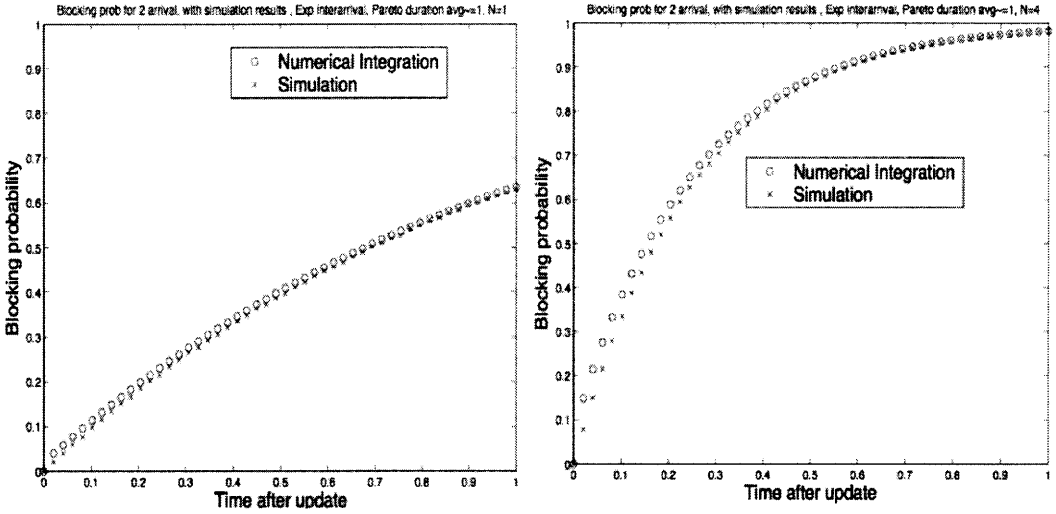
If we resolve the max function in (4a) for the y_2 integral to such that $k > t-y_1-x_1$, then the integration limits become from k to k and the result is zero. Therefore, the resolution of the max function must be the term $t - y_1 - x_1$ and this is the upper limit of the y_2 integral. y_1 can clearly only be in the range $[k, t]$, since if it is longer it would be active at time t. x_1 must be short enough to allow y_1 to complete by t, so $x_1 < t-k-y_1$, which is a tighter bound than the original $t-y_1$ in (4a). We therefore change the upper limit of the integral for x_1 to reflect this.

The sum of the above four integrals form the blocking probability lower bound when $t > k$.

In the case $t < k$, any blocking must be due to the first arrival. If $t < k$, blocking corresponds to X_1 being less than t, since if this condition exists, we have guaranteed blocking. Recall that the domain of the pareto distribution is $> k$. Then the integral for (1) becomes:

$$\int_0^t p(x_1) dx_1$$

These integrals do not admit a simple closed-form solution for the Pareto PDF as the Exponential PDF did. Therefore we resort to using numerical integration in MATLAB using a four-nested loop. We discretize a range vector for the Pareto PDF for (y_1, y_2) as well as the exponential PDF for (x_1, x_2), as four vectors of equal size. We loop over each of these vectors, checking for specific conditions specified by the limits of the integrals above. Note that the loops must be nested in the same order as the order of integration. To calculate the volume of the increments, we use the MATLAB *int*() function. The MATLAB *int*() function calculates a discrete approximation to an integral of a specfied dimension using numerical limits in each dimension and a functional description of the objective function. In this case we use it to calculate a 4-dimensional, discrete approximation to the integral of the joint PDF for each increment of the four variables, summing when the limit conditions are met. This



(a) 1 second Pareto duration, Exp. inter-arrival, N=1

(b) 1 second Pareto duration, Exp. inter-arrival, N=4

Figure 4-4: Blocking probability for Pareto duration

was done for each of the integrals above, and the result was summed to get the final result.

We have also implemented a simulation of the two arrival scenario that draws numbers from an exponential distribution for arrival times and a Pareto distribution for holding times. The simulation was run for tens of thousands of iterations until the statistic in question, in this case, blocking probability, had stabilized, not varying more than 10^{-4} after a few thousand more iterations. The simulation models the same number of nodes and distributions as the analysis. Figure 4-4(a)(b) show results for the transient analysis assuming flows follow a Pareto distribution with an average of one second. These results show a higher blocking probability for the analysis using the the heavy-tailed Pareto distribution, when compared to the exponential. This is likely because the durations in the Pareto case have a non-zero lower bound and longer duration flows are more probable. The simulation results match closely the numerical integration results which shows the accuracy of the analysis.

4.2.3 Summary

The curves show clearly the problem with timescale introduced by periodic updates. In summary, an update based approach must update very rapidly ($<< 1$ *second*) if it is to provide valid network state information. Recent GMPLS implementations have reported an update interval of approximately 3-4 updates every 30 seconds. This periodicity appears to be much higher than that needed for a reasonable blocking probability. Also note that these updates suffer time-of-flight latency as well. GMPLS allows tuning of these parameters to reduce the broadcast period, but this approach may result in excessive control traffic for the update-based approach.

4.3 Analysis of Reservation-Based Optical Connection Setup

Current literature proposes a reservation-based approach for optical connection setup in a MAN or WAN. This approach proposes setup based on network state information found by on-demand network exploration packets, making reservations on the a return trip to the sender. We analyze a simple reservation-based network examining implications of this for connections of short duration ($\leq$ 1 second) and high arrival rate (i.e. high utilization). We perform transient behavior analysis on a network model to analyze a reservation-based approach. Our results show that control network latency results in significant blocking probability for the reservation-based approach in a highly dynamic network (e.g an OFS network)

4.3.1 Motivation

An OFS network must make choices for routing and wavelength assignment for all-optical connection setup. A reservation-based approach uses network state information gathering control packets to determine the availability of network resources on-demand. In this approach, upon arrival of a connection request the sending node sends packets to examine the state of the network along one or more routes to the

destination. At the destination, the information gathered by these packets are evaluated and a route and wavelength are selected in order to make a *reservation* i.e. hold optical resources for the request. A reservation control packet is then sent along the selected route and actual reservations are made in the reverse direction. Upon receipt of the reservation packet, the sender is informed of the successful reservation and the flow is sent.

The issue we address in this study is that both the information packets and reservation packets suffer latency. At minimum this latency is time-of-flight delay, but it can also include queuing and processing delays. These delays result in the information gathered being stale i.e. out of date by the time they are received. In a similar way, the reservation packet suffers delays during which the state of the network can change from the protocols original view. We investigate the impact of these delays on a simple network model.

4.3.2 Methodology

The network model for analysis is structurally the same as the network shown in Figure 4-1. In studying the reservation-based protocol, we analyze the blocking probability of the network using the following assumptions/protocol.

- A connection request to B arrives at A at time t_0, there are N intermediate nodes. A single arrival blocks the reservation.
- Cross-traffic arrives at the intermediate nodes with an exponentially distributed interarrival time with parameter λ.
- Control packets suffer a time-of-flight delay at each hop of d.
- A sends an information-gathering packet at time t_0 to B along the route.
- The information packet finds each intermediate node free (otherwise, the reservation would be canceled).
- Upon receipt of the information packet, B sends a reservation packet to A to reserve the resources that are apparently free.

Our analysis proceeds in two independent phases. It is possible to analyze the protocol using a single phase, involving the calculation of the probability of blocking given a round trip time. We have chosen the phase-based approach as it makes the phases explicit, and allows for manipulation of the phases independently, if desired. First, we analyze the information gathering phase from A to B. In this phase we calculate the probability of an arrival during the phase after the information packet passes each node. We assume that cross traffic arrives at each node as a Poisson process with parameter λ. Thus for any node i, the probability that there will be an arrival at i by the end of the phase (P_{b1}) is:

$$P_{b1}(i) = 1 - \lambda\, d(N - i + 1)\, e^{-\lambda\, d(N-i+1)}$$

That is, at node i, we wait $(N - i + 1)$ latencies for the Poisson process to have an arrival. The probability of an arrival clearly decreases with increasing i. Then, the overall probability of an arrival at any node after the phase is:

$$P_{b1} = \bigcup_{i=1}^{N} P(arr\, at\, node\, i)$$

This is a union of N independent events, with the probability of each depending on i. This is then:

$$P_{b1} = 1 - \prod_{i=1}^{N} (1 - P_{b1}(i))$$

Analysis of the reservation phase proceeds similarly. Define P_{b2} to be the blocking probability during this phase. Given that there was no arrival during the information phase, the probability that there will be a blocking arrival to node i during the reservation phase is the same as before:

$$P_{b2}(i) = 1 - \lambda\, d(N - i + 1)\, e^{-\lambda\, d(N-i+1)}$$

Making the probability of blocking in phase two:

$$P_{b2} = 1 - \prod_{i=1}^{N}(1 - P_{b2}(i))$$

As an optimistic assumption, we condition overall blocking on the event that blocking does not occur in *both* phase one and phase two. This results in the following:

$$P_b = P(b1 \cup b2|\overline{b1 \cap b2}) = P(b1|\overline{b1 \cap b2}) + P(b2|\overline{b1 \cap b2}) - P(b1 \cap b2|\overline{b1 \cap b2}) =$$

$$P(b1|\overline{b1 \cap b2}) + P(b2|\overline{b1 \cap b2})$$

Where the last equality holds because of mutually exclusive events. Use of Bayes Rule and DeMorgans Law shows that:

$$P(b1|\overline{b1 \cap b2}) = \frac{P(b1)(1 - P(b2))}{2 - P(b1) - P(b2) - (1 - P(b1))(1 - P(b2))}$$

Similarly for blocking in phase two we have:

$$P(b2|\overline{b1 \cap b2}) = \frac{P(b2)(1 - P(b1))}{2 - P(b1) - P(b2) - (1 - P(b1))(1 - P(b2))}$$

The overall blocking probability is the sum of these two as described above.

Accounting for duration

A slightly more detailed analysis takes duration of cross-traffic arrivals (Y) into account. In this case, we assume that arrivals have independent randomly distributed durations. While we still consider only one arrival at each node, we now account for the case of the arrival departing before it can actually block the $A- > B$ reservation. In this case, the expression for blocking changes for both phases of the protocol. Define $RTT(i, d)$ to be the round trip time from node i with inter-node latency d, and $1way(i, d)$ to be the one way delay from node i given inter-node For a node i, the probability of blocking for the phases are is:

$$P_{b1} = P(arr\,at\,node\,i \wedge duration > RTT(i,d))$$

$$P_{b2} = P(arr\,at\,node\,i \wedge duration > 1way(i,d))$$

These expressions are intersections of independent events so we can multiply probabilities:

$$P_{b1} = P(arr\,at\,node\,i)P(duration > RTT(i,d))$$

$$P_{b2} = P(arr\,at\,node\,i)P(duration > 1way(i,d))$$

Here, RTT(i) is the round trip time from i through B back to i, and 1way(i) is the one way trip time from B to i. Given the PDF of Y, we can calculate these probabilities as functions of i and d:

$$P(duration > RTT(i,d)) = 1 - \int_0^{2(N-i+1)d} p_Y(y)dy$$

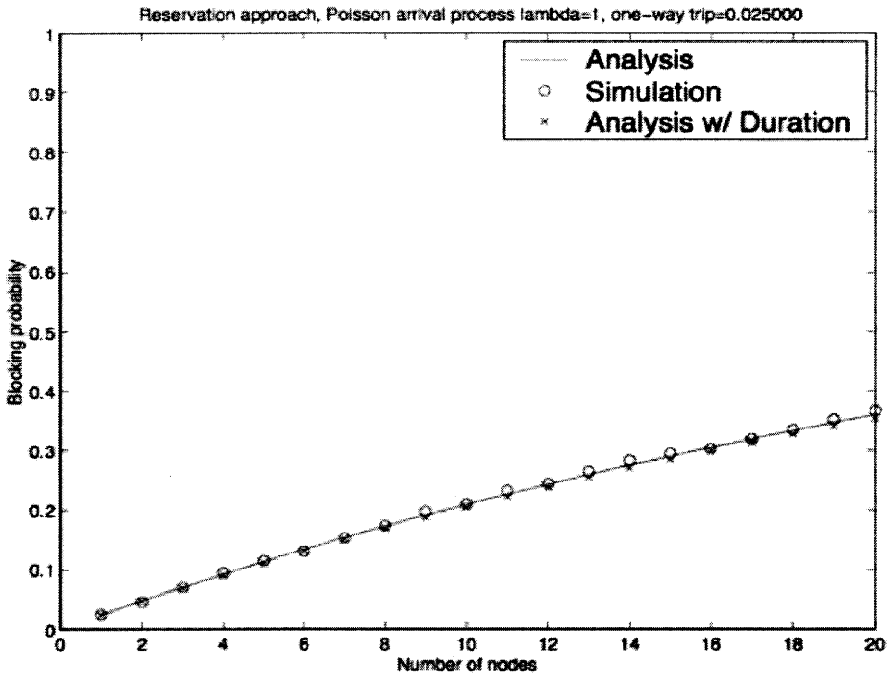
$$P(duration > 1way(i,d)) = 1 - \int_0^{(N-i+1)d} p_Y(y)dy$$

For Y with an exponential PDF this evaluates to:

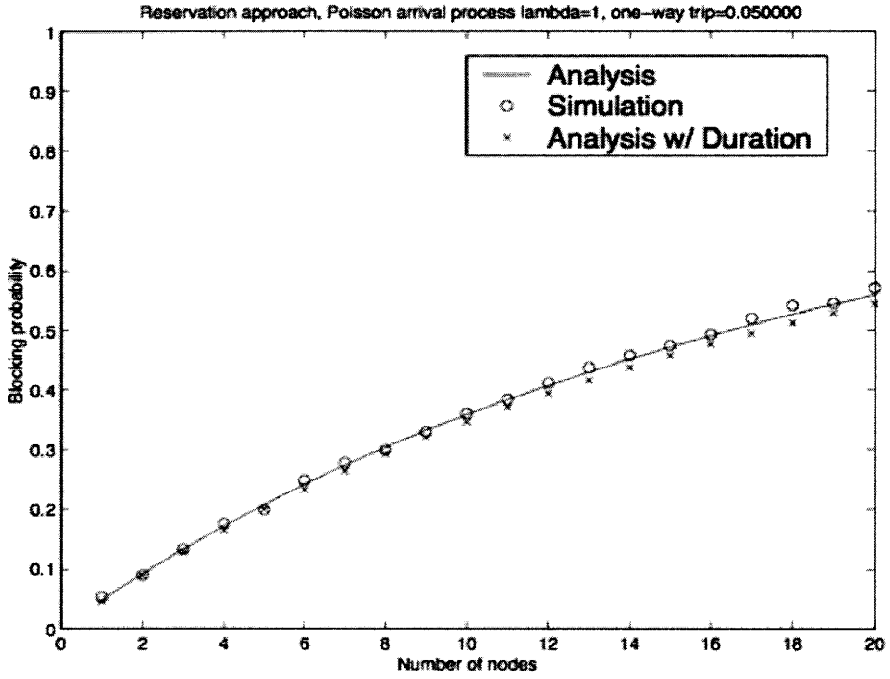
$$P(duration > RTT(i,d)) = 1 - (1 - e^{-2(N-i+1)d}) = e^{-2(N-i+1)d}$$

$$P(duration > 1way(i,d)) = 1 - (1 - e^{-(N-i+1)d}) = e^{-(N-i+1)d}$$

Figures 4-5(a) and (b) show the blocking probability results for network round trip times (RTT) of 50 and 100 milliseconds respectively, for up to N=100 nodes. 50 ms is close to time-of flight delay for a WAN, while 100 ms may include some software delay or other processing overheads. The average interarrival time and service time



(a) 50 ms round trip time



(b) 100 ms round trip time

Figure 4-5: Blocking probability vs. number of nodes, analysis and simulation results, average interarrival time and service time both equal 1 second.

is one second, and both are exponentially distributed. Recall that a single arrival in either phase blocks the reservation. The dashed lines in the plots show the results of a simulation of the reservation-based approach carried out in MATLAB. The blocking probabilities obtained are virtually the same as the analysis and verify the result.

Both graphs illustrate the issue with reservation-based approaches for connection setup. For N=10 nodes, the 50ms and 100ms RTT graphs show a blocking probability of nearly 20% and 30% respectively. This shows a timescale mismatch between this approach and the problem of rapid dynamic connection setup. The effect of including durations is small with blocking slightly lower.

Consideration of Cross-Traffic Length

A concern for the blocking probability analysis above is that it is too pessimistic as it assumes that each cross traffic arrival occupies exactly one node. This is a pessimistic view because it ignores the case that a short duration arrival blocks a subsequent long duration arrival due to path length, and overall blocking does not occur due to this loss.

In this subsection, we calculate an upper bound on the probability of this event based on an assumption of up to two arrivals to the system. In order to ensure an upper bound, we assume that all arrivals occupy the route to B and the time we consider is an entire round-trip reservation interval R. The system is now a loss network with two incoming arrivals with independent variables defined as follows:

$$X_1 : interarrival\, time\, of\, first\, arrival$$

$$Y_1 : duration\, of\, first\, arrival$$

$$X_2 : interarrival\, time\, of\, second\, arrival$$

$$Y_2 : duration\, of\, second\, arrival$$

Then the event we are seeking has the form:

$$(X_1 + Y_1 < R) \wedge (X_2 < Y_1) \wedge (X_1 + X_2 + Y_2 > R) \qquad (5)$$

The first two terms in the event ensure that the second arrival is lost before the reservation interval. The third term ensures that the second arrival *would have* blocked the system if it had not been lost. As these are independent variables, the joint PDF is known so the probability of the event is an integration over each variable. The resulting integral is:

$$\int_0^R \int_0^{R-y_1} \int_{R-x_1-x_2}^{\infty} \int_{max(0,R-y_2-x_1)}^{y_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1$$

An explanation of the limits of this integral follows:

- *Limit:* $max(0, R - y_2 - x_1)$ *to* y_1 *for the integral of* x_1 - This lower limit comes from the third term of (5) with the max function used to ensure positivity of the limit. The upper limit comes from the second term of (5).

- *Limit:* $R - x_1$ *to* ∞ *for the integral of* y_2 - This lower limit comes from the third term of (5). Note that this limit is guaranteed to be positive because the x_1 must be less than R because of the first two terms of (5). The duration of the second arrival can is unbounded making the upper limit ∞.

- *Limit:* 0 *to* $R - y_1$ *for the integral of* x_1 - The lower limit says that the interarrival time of the first arrival can be as low as 0. The upper limit comes directly from the first term of (5).

- *Limit:* 0 *to* R *for the integral of* y_1 - The duration of of the first arrival can be as large as R but no larger, due to the limits imposed by the first term of (5).

The resolution of the max function in the integral above results in a sum of two integrals

$$\int_0^R \int_0^{R-y_1} \int_{R-x_1}^{\infty} \int_{R-y_2-x_1}^{y_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1 \qquad (6)$$

The above integral comes from the case where $R - y_1 - x_1 > 0$. This ensures positivity of all lower limits in the integral.

$$\int_0^R \int_0^{R-y_1} \int_{R-x_1}^{\infty} \int_0^{y_1} p(x_1, x_2, y_1, y_2) dx_2 dy_2 dx_1 dy_1 \qquad (7)$$

The above integral comes from the case where $R - y_1 - x_1 < 0$. This ensures positivity of all lower limits in the integral.

For our analysis we assume all variables in the expressions take on exponential PDFs so:

$$p_{X_1}(x) = p_{X_2}(x) = \lambda e^{-\lambda x} \quad (x > 0) \qquad p_{Y_1}(y) = p_{Y_2}(y) = \mu e^{-\mu y} (y > 0)$$

We can use the MATLAB symbolic integration package to get the expressions that result from integration of (6) and (7). This package uses the Maple mathematical routines to perform integration for an arbitrary number of integrands. These results will result in expressions in terms of λ, μ, R. The expression that results from integrating (6) is:

$$\frac{\mu}{\lambda + \mu}(2e^{-(\lambda+\mu)R} + \frac{1}{2}e^{-(\lambda+2\mu)R} + \lambda e^{-(\lambda+\mu)R}R + \lambda e^{-\mu R}R - \frac{5}{2}e^{-\mu R})$$

The expression that results from integration of (7) is:

$$\frac{\mu}{\lambda + \mu}(2e^{-(\lambda+\mu)R} - \frac{1}{2}e^{-(\lambda+2\mu)R} + \lambda e^{-\mu R}R - \frac{3}{2}e^{-\mu R})$$

It is informative to look at the limits as R varies. We can see from that for both expressions (and hence, their sum), the value goes to 0 as R goes to ∞ and 0. In the case that R is very small, it is very unlikely that the second arrival would occur during the reservation interval so the probability it would be lost is close to zero. As R goes to ∞, it becomes unlikely that the second arrival would be long enough to block the system (if it had been able to enter the system), so this probability should go to 0 as well.

R	P_{loss}
.025	$> 2 \times 10^{-8}$
.05	$> 3\times, 10^{-7}$
.1	$> 4 \times 10^{-6}$

Figure 4-6: Probability of loss due to maximum path length

The results below are for the case when $\lambda = \mu = 1$. The integral has been performed using the MATLAB symbolic integration package, which performs symbolic integration of formulas using symbolic integral limits. The package calls into the Maple mathematical library to perform the integration symbolically. The analysis resulted in the following expression:

$$e^{-2R} + 1/4e^{-3R} + e^{-2R}R + 1/2e^{-R}R - 5/4e^{-R} + -1/4e^{-3R} + e^{-2R} + 1/2e^{-R}R - 3/4e^{-R}$$

The first five terms in this expression come from the integral (6), and the rest from the integral (7). Using this expression, we can calculate the probability of (5) given a value for R. The results are tabulated in the table below for 3 R values.

As can be seen by the calculations in Figure 4-6, the effect of path length is minimal mainly due to the small timescales being considered for the reservation approach. Compared to the blocking probabilities that were found by the reservation-based analysis earlier in this section, the values in 4-6 are at least 4 orders of magnitude smaller. This shows that path length is at most a secondary effect in the analysis.

4.4 Discussion

We have seen the following from the analysis:

- An update-based network information dissemination approach provides information that rapidly becomes stale and less useful with time.
- A reservation-based approach suffers from latency that causes apparently promising reservations to fail with high frequency. This occurs when the arrival rate λ

and average duration μ are on the order of a second, and the number of nodes is on the order of 10. In this particular case, we have seen that an end-to-end latency of .05 seconds can result in a large blocking probability for reservations. There are other timescales for these parameters that can result in high blocking as well. In general, that a tradeoff exists between the rate of arrival λ, the service time μ, and the round-trip time, defined by and d and N for our model. A for a given N, a higher arrival rate and smaller duration for flows will result in smaller timescale changes in network resource allocation. If the order of these changes are on the order of the RTT time, then a reservation-based approach will suffer blocking, as seen in the transient analysis. Obviously a higher number of intermediate nodes, N, will also result in higher blocking, as there is more incident cross traffic in this case.

These two issues show that a control plane for an all-optical network must address the issue of timescale if it is to be effective. As a point of comparison, timescale is less important in packet-based IP networks [41] because of the presence of buffering capabilities at routers. This discussion does not explore possible solutions to the timescale issues although they do exist, and are being studied.

As a concluding note, the studies contained in this chapter can be considered to be largely independent of those of subsequent chapters. This is because subsequent studies use an idealized control network that makes allocations in the network immediately and wholesale. The reader may want to consider the performance of the networks in subsequent chapters if modeling of a control plane such as that we have described here was included.

Chapter 5

Basic OFS Model

5.1 OFS Model Assumptions

We wish to study the behavior of OFS using closed form and numerical analysis. The first step is to choose a model for an OFS network architecture that accurately models the essential behavior of the network. In this section, we define a basic model of OFS, a single-channel general mesh network topology. In order to focus the study, we make certain assumptions about the design of this model. We list these here, along with our choices, along with some justification.

- **Scale and Topology** - As stated earlier, we are studying OFS in a MAN connecting to WAN scenario. While the scale of these networks seem large in terms of geographical scale, they are small in terms of the number of optical components and terminals. We expect the diameter of both optical MAN and WAN networks to be less than 10 nodes.

- **Physical Infrastructure** - Network components are idealized versions of optical fibers, switches and end terminals. We do not model physical infrastructure in detail because it is complex and not germane to the statistical outputs we are trying to obtain. In addition, we do not model control messages in the current analysis.

- **Application Layer** - Our model assumes connection requests are generated by abstract arrival processes, as opposed to concrete applications. OFS flows will have probabilistic holding or transaction times specified by probability distributions. This makes the models solvable using Markov theory and this is sufficient for our purposes. For the same reasons, we do not model application layer or user interface effects.

- **Channel Layout** - The networks to be studied have single or multiple channels per fiber. In the multiple channel case, we assume no wavelength conversion. This assumption imposes the wavelength continuity constraint on our models.

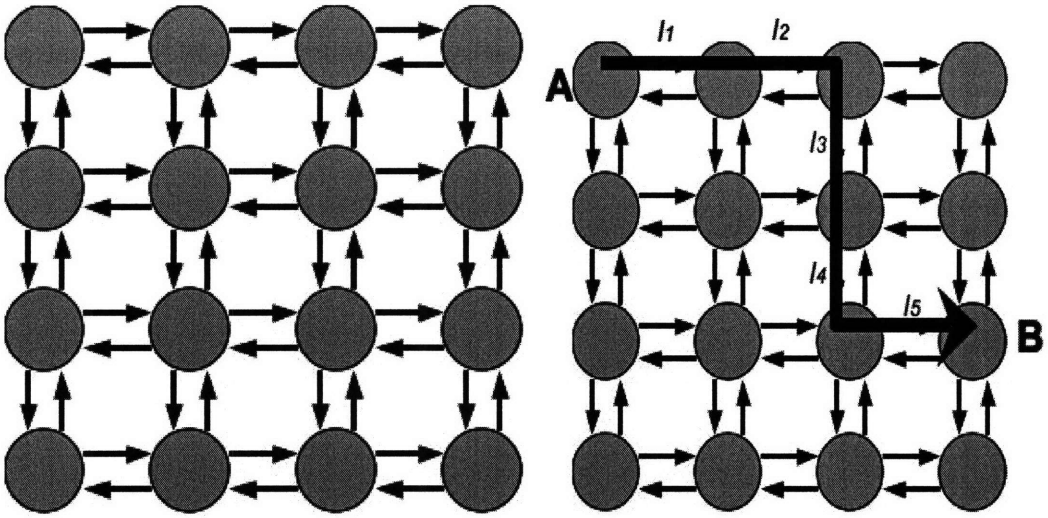
- **Metrics** - Two user-based outputs of the model are in terms of blocking probability, and delay of OFS flows. In addition, we examine utilization of optical network resources as a metric of interest to network designers. Mathematical definitions for each of these metrics will be presented later in this thesis.

These assumptions lead to the model of choice, which we describe here. Note that for subsequent specific studies presented in later chapters, additional assumptions about the scenario need to be made. However, the basic assumptions of the model (specification, physical layer) remain the same throughout all studies presented here. This formulation is similar to the circuit switch model described in [4]. We have modified it to suit our purposes in modeling OFS.

5.2 Description of Model Network Components

Figure 5-1(a) shows a general mesh network. It contains a collection of nodes, in this case a 4x4 mesh, and directional links between the nodes for optical connections. Each link can have a number of channels, and we assume this number to be equal among all links in the network. For this initial expository discussion, we assume that there is a single channel per link but we will expand this assumption in later models.

The nodes are presumed to be idealized Optical Cross-Connects (OXCs). They are channel selective and fully non-blocking, and can reconfigure instantaneously. These



(a) General mesh network (b) Network route from node A to B

Figure 5-1: Network for Analysis

assumptions are not realizable in real-world networks, but they serve both to simplify the model as well as lend the hypothesis of this thesis more credence. Links will be denoted as l_l and will have generic indices for the network of interest. Channels within links are also distinct, but we do not give notation for them here, since our current model is of a single-channel per link. A channel in the directional link is said to be *utilized* if both it and its source nodes output port are in use. This is an important concept as the metrics of these models will depend on utilization of channels (i.e. links).

We now define a route inside the network. A route is defined to be a series of consecutive directional links (and associated nodes) that does not contain a the same node twice. Figure 5-1(b) shows an example route in the 4x4 mesh network. We will designate a route as r_n^{XY} where the subscript n is a generic index, and the values X and Y denote the beginning and ending node of the route. The route shown in Figure 5-1(b) would be written r_n^{AB} since it is from node A to node B and can have an arbitrary index. Generally a route r_n^{XY} can be expressed as a collection of consecutive links as follows:

$$r_n^{XY} = \{l_i l_j ... l_k\}$$

In this example, the source node of link l_i is node X and the end node of link l_k is node Y. As long as the intermediate nodes are consecutive in the network and no node is encountered twice, this is a valid route. The route in Figure 5-1(b) would be written, given the links labeled in the figure:

$$r_n^{AB} = \{l_1 l_2 l_3 l_4 l_5\}$$

A *routeset* R in a network may be defined as a set of valid routes. The cardinality of a routeset R is denoted $|R|$. A routeset is defined as follows:

$$R = \{r_1^{AB} r_2^{AC} r_3^{AD} ... r_n^{YZ}\}$$

A valid routeset defines a routing on the network in that it defines all the active node pairs and what routes are used to communicate between them. The subscript of each route denotes its index; the superscript denotes the starting and ending nodes of the route. An example of this might be a shortest-path routing between specific pairs of nodes. The route shown in Figure 5-1(b) is a shortest path route between nodes A and B. There are others, and the entire collection of these routes might form an example routeset. The maximum cardinality of a routeset with l links has order $O(2^{(l^2)})$

5.3 Description of OFS Traffic Model

5.3.1 Arrival and Departure Process Definition

We model traffic as a set of stochastic processes at each of the routes in the network. These processes are defined by free parameters in the model and are considered to be inputs. The processes are stochastic and will be described by probability distributions. In particular for the traffic process to be completely specified, it needs a flow *arrival*

distribution and a flow *holding time distribution.* Analysis then proceeds using these distributions to calculate metrics of interest about the network model.

For the present discussion, we present the a traffic model using basic distributions. Later models use more complex distributions. For an arrival process model, we choose the Poisson arrival process. This process defines an inter-arrival distribution for flows that is exponential with a single parameter λ, which is the *average arrival rate.* More details about the Poisson arrival process can be found in [7]. For a routeset R as defined above, we can define a collection of arrival processes that define the distribution of call arrivals to each of the individual routes. We write such a set Λ:

$$\Lambda = \{\lambda_1\lambda_2\lambda_3....\lambda_n\}$$

Note that this set must have the same cardinality as the routeset, $|R|$. It defines the distribution of the arrival process of flows at each route in the network. We can similarly define the holding time distribution of the flow at each route. For this discussion, we use random variable with an exponential distribution for the holding times, with parameter μ, and Probability Density Function (PDF):

$$P_x(x) = \mu e^{-x\mu}, \quad x \geq 0$$

Therefore, given the routeset R defined above, we can describe the holding time distributions for the routeset, $\overline{\mu}$ as follows:

$$\overline{\mu} = \{\mu_1\mu_2\mu_3....\mu_n\}$$

Again, the cardinality must match that of the routeset. These distributions define the traffic that arrives at the OFS network. The traffic model here is related to that of basic queuing models as discussed in [7], but these models are more complex due to the interaction of the routes in the topology of the network.

5.4 OFS State Space Model Description

5.4.1 System State Definition

We now define a state of the OFS model. Qualitatively, for a given network, the state describes the occupancy of each route inside the network by OFS flows. Since routes are comprised of links and nodes, a valid state description describes the occupancy of each of the links and nodes of the system and in fact the entire network. The model state evolves with time. As the arrival and departure processes evolve, the system will change state, that is, undergo state transitions. This is due to flows being admitted to the network or flows departing the network. A flow arrival to a route will cause the network to transition to another state (possibly to the same state). The departure of a flow will cause the network to transition to another distinct state. By studying the time average state that the system is in, we can quantify the performance of OFS for a number of models of interest.

For a given network, assume that we have a routeset R, of cardinality $|R|$, that describes all the routes that are defined for the network. Recall that we are currently studying networks with a single channel per link. The state $S(t)$ of the network as a function of time, can therefore be expressed as a length $|R|$ binary vector that evolves with time, where we denote $\overline{S}(t)$ as a vector:

$$\overline{S}(t) =< b_1 b_2 b_3 ... b_{|R|} >$$

Here, each of the b_i correspond to a binary digit, for example, at a particular time T, we might have:

$$\overline{S}(T) =< 011...0 >$$

We stipulate that $b_i = 1$ iff route r_i is occupied by an OFS flow. So, in the above example, routes b_2 and b_3 have active flows. If a flow is active at a route, all links involved in route r_i are occupied. We now define the conditions for *admissibility* of a particular state. Given the above formulation, it is clear that there are $2^{|R|}$ possible

states. However, due to constraints on channels (equivalent to link constraints for the present discussion), not all of these state assignments are admissible. In particular, each link can only be occupied by one flow. Therefore, in any particular state, if two or more routes that involve the same link are active, then the state is not admissible. The converse and inverse of this statement are also true. Using more formal notation, we define a boolean value function of a state ADM($S(t)$) as follows:

$$ADM(S(t)) = 1 \Leftrightarrow \bigcap_{\text{links in } r_i \in R \text{ s.t. } b_i=1} = \emptyset$$

This formula states that the ADM takes on a value of 1 iff none of the routes that are active in the argument state occupy the same link. This is a basic condition for admissibility in this single channel network.

In general, our analysis will deal only with admissible states of the system. That is, states $S(t)$ where $ADM(S(t)) = 1$.

5.4.2 Transition Definition

In order to fully specify the structure of the state space, we must define inter-state *transitions*. Transitions define the structure of the state space model, specifically state-to-state transitions given an arrival or departure of traffic. We have already seen arrival and holding time processes defined for routes inside the network. The processes we defined will form the rates of transition in the continuous time Markov model. An arrival of a flow to a route r_i will cause a transition to another state iff the state generated by making $b_i \rightarrow 1$ is admissible as defined above. Otherwise, we consider the arrival blocked and it departs the system. This will be further discussed when we discuss performance metrics below. A departure of a flow will always result in an admissible state.

Given definitions for states of the system we can define a Markov Chain that describes transitions between these states. The rates for these transitions are defined by the Λ and $\overline{\mu}$ parameters in the traffic model. More formally, define two state transitions exiting in the state S, τ_D which is a departure transition and τ_A, an

arrival transition as follows:

$$S \rightarrow_{\tau_D} S'$$

$$S \rightarrow_{\tau_A} S''$$

Assume that both the transitions are associated with the route r_i in R. Then for the departure transition τ_D we have:

$$S =< b_1 b_2 b_3 .... b_i = 1 b_{i+1} ... > \rightarrow_{\tau_D} S' =< b_1 b_2 b_3 .... b_i = 0 b_{i+1} ... >$$

Here the application of τ_D to state S has resulted in a state S' where $b_i = 0$ or equivalently where there is no active call at route r_i. By earlier definitions, the rate of transition from S to S' is $\tau_D = \mu_i$.

In the case of an arrival transition, there are several cases. If $b_i = 1$ in state S, then the arriving call cannot be admitted. Therefore in this case, $S'' = S$, and the call is blocked. Otherwise, if $b_i = 0$ in state S, then we apply the ADM function to the state $S_{b_i=1}$ which is the same as S except that $b_i = 1$. If $ADM(S_{b_i=1}) = 0$ then the call cannot be admitted and $S'' = S$ and the arriving call is blocked. Otherwise, the call can be admitted and becomes active at the route r_i. This means that $S'' = S_{b_i=1}$ and the rate of the transition from S to S'' is $\tau_A = \lambda_i$.

These definitions of states and state-to-state transitions define the entirety of this OFS Markov model. We will apply numerical and closed-form analysis to this model in the subsequent chapters and find quantitative and theoretical results for the OFS model presented. Modifications to this basic model will be described as they are encountered in the analysis sections, but the basic state-space structure will remain consistent. In the chapters that follow, the basic assumptions of the model described here will be expanded to include:

- Specific network topology types
- Multiple channel networks
- Scheduled OFS networks

5.5 Metric Definitions

The state space model we have described allows average case analysis. Techniques exist to calculate the stationary distribution of discrete-state Markov Chains such as the one we have described here. This distribution assigns a long-term probability of the system being in a particular state for every state in the model. Both numerical and closed-form solutions exist for this problem. We now define the three generic but rigorous metrics of interests in our studies: blocking probability (P_b), utilization (U) and delay (D). In order to discuss these metrics, we introduce the following variable definitions:

- Define a Markov Chain representing an OFS system with state space set $\overline{S}$.
- Define S to be a variable representing an individual state in $\overline{S}$.
- Define P_S to be the stationary probability of the state S.
- Define L_S to be the number of links in use in state S, recalling that we are dealing with a single channel case at the moment.
- Define $|R|$ to be the number of possible routes in the network; assuming routes are indexed from 1 to $|R|$ sequentially.
- Define P_b to be the overall blocking probability in the network modeled by $\overline{S}$.
- Define U to be the utilization in the network modeled by $\overline{S}$.
- Define D to be the average delay in the network modeled by $\overline{S}$.
- The definitions for Λ, μ from the Section 5.3 apply.

Given these definitions, we proceed to the metric definitions.

5.5.1 Blocking Probability

The blocking probability metric answers the following question: At some arbitrary time when the system is in steady-state, given a flow arrival, and given an equal

likelihood of the arriving flow originating and terminating at any node, what is the probability that the flow is blocked, i.e. not admitted to the system. Note that this question explicitly assumes that at any type of call (route, length) is equally likely. This is a particular assumption that we have made for our model; others are possible.

We now define steady-state blocking probability. In steady state, each state S has some steady-state probability, P_S. Define $S_{b_i=b_i+1}$ to be the state S with 1 added to the component b_i. Then the expression $ADM(S_{b_i=b_i+1})$ returns a 1 iff route r_i can admit an arrival in state S. If this this value is 1, there is no blocking of route r_i in S. We assume that every route is equally likely to have an arrival in steady state. Thus the contribution of route r_i in state S to the overall blocking probability is the stationary probability of S, or P_S, times the fraction of P_S contributed by route r_i, or $\frac{1}{|R|}$, recalling that in a given state we an arrival to any route equally likely. Finally, we sum these contributions over all routes and all states to get the final expressions. Note that the assumption that each route is equally likely to have an arrival can be adjusted to reflect other models of arrivals.

The following is the definition for P_b, define $S_{b_i=b_i+1}$ to be the state S with 1 added to the component b_i:

$$P_b = \sum_{S \in \overline{S}} \sum_{1 \leq i \leq |R|} ADM(S_{b_i=b_i+1}) \times \frac{1}{|R|} \times P_S$$

In words, the above formula sums over each state S and over each type of arrival to route r_i where $1 \leq i \leq |R|$ of the following: The admissibility function value (either 0 or 1) associated with adding a flow to the route r_i times the reciprocal of the number of call types that can possibly arrive which is the reciprocal of the number of routes $\frac{1}{|R|}$, times the probability of being in the state S $(= P_S)$.

This sum adds up to P_b, the overall blocking probability of the system, in steady state.

5.5.2 Utilization

The utilization metric answers the following question: In the long-term average, what percent of links (channels) are in use in the system? This question looks at resource occupation in the network, and is important from a network designers perspective. Although modern WDM networks have a large number of channels, it is beneficial to limit OFS to as small a proportion of the network as possible.

The following is the definition for U:

$$U = \sum_{S \in \overline{S}} P_S \times L_S$$

This summation averages the number of links in use by individual states over the stationary distribution.

5.5.3 Delay

Flow delay is only applicable in a scheduled OFS model. Since the model has not been presented yet, we defer its definition and calculation description to later in the thesis.

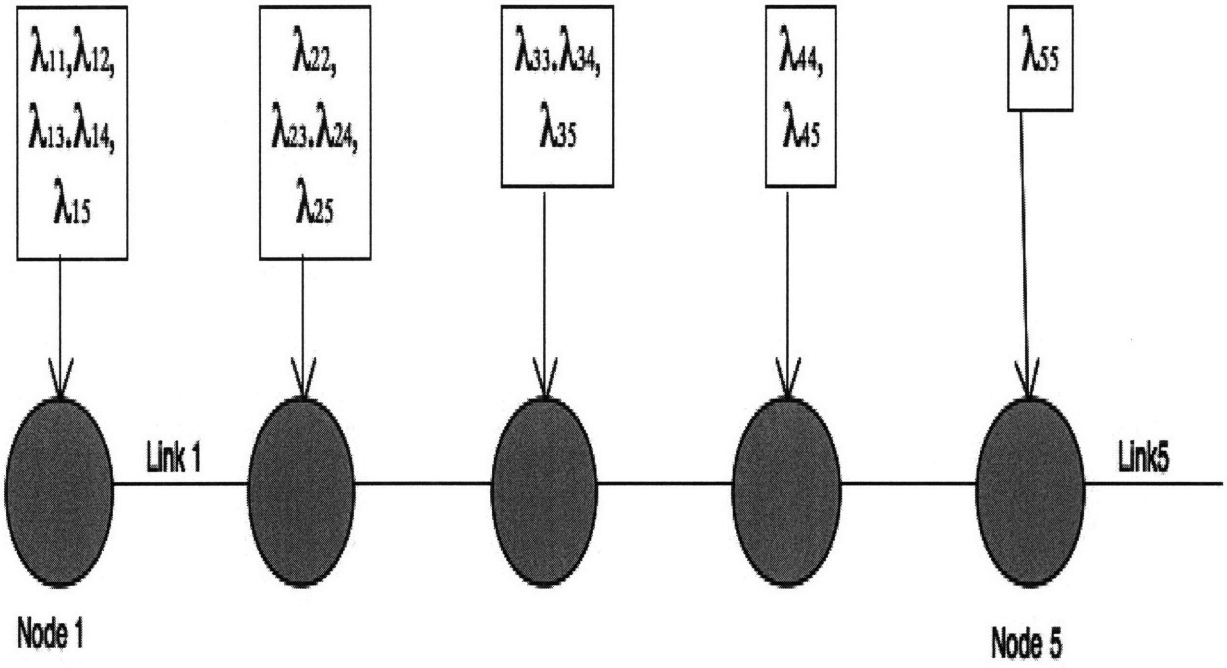
Chapter 6

OFS Analysis

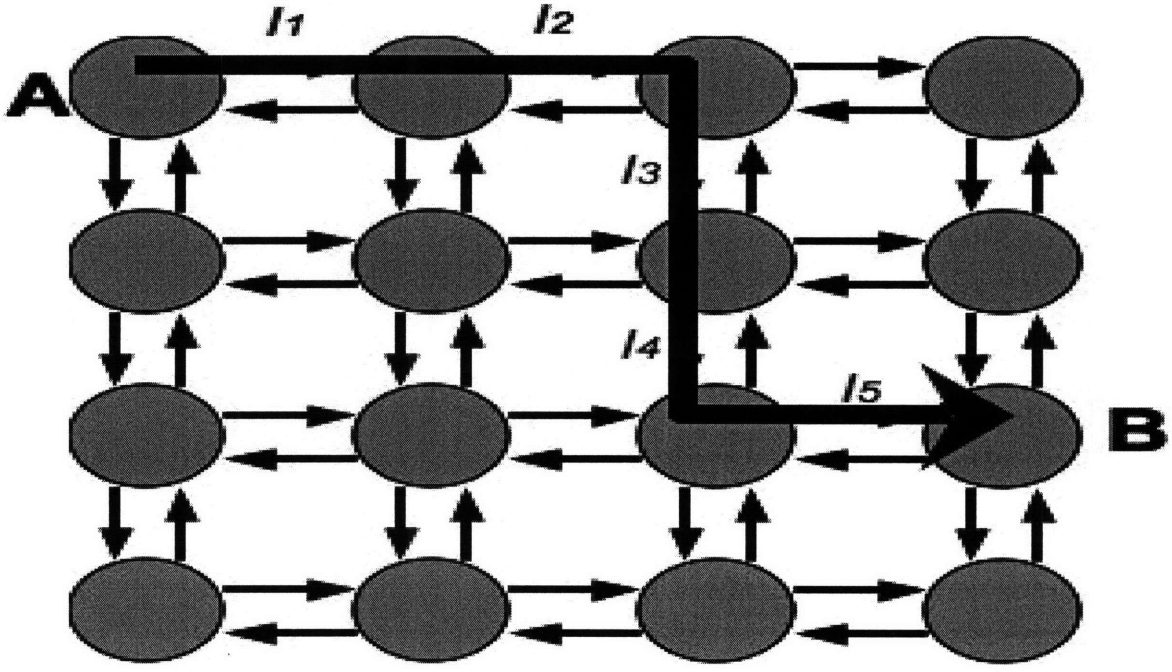
This chapter presents numerical analysis of the performance of an OFS system. The models presented here can be considered to be specializations or extensions of the basic OFS model presented in the previous chapter. Each section of the current chapter will present an explanation and motivation of a particular instantiation of an OFS model and numerical results for the model including the metrics involved. Different metrics are involved in various models and the model description will include a list of metrics where applicable. The model outputs studied here together form strong evidence of a fundamental issue that we have identified for OFS and in fact many all-optical network architecture approaches. The final section presents a simple model for *scheduled* OFS which seems to give improved performance at the cost of delaying calls an acceptable amount. We begin with a single-channel line network model for OFS.

6.1 OFS in a Single-Channel Line Network

A line network is a multi-hop network with all nodes having a nodal degree of two except for two nodes which are degree one. Figure 6-1(a) shows an example of a line network. This network has N=5 nodes and a single optical channel connecting the 5 nodes. Note that the links in the network are directional, so traffic only flows in one direction (left to right in the Figure 6-1(a)). The routes for flows to travel that are



(a) Five node single line network



(b) Route inside a general mesh network

Figure 6-1: Line network model and justification

possible in this network are any number of contiguous links. For example, a route from node 1 to node 5 exists involving all links, a route from node 2 to node 3 exists involving the links emanating from nodes 2 and 3 and so on. Single node/link routes also exist in the network model. In general, a single directional line network with N nodes has $\sum_{i=1}^{N} i$ distinct routes in it.

Figure 6-1(b) shows a general, single channel mesh OFS network, with directional links. In the figure, a particular directional route is highlighted between two given nodes. If we assume that OFS calls emanate from any node and terminate at any node, then calls will arrive to this route and use various contiguous segments of it according to some arrival process, definable per-node. We can approximate this situation as a line network, with arrival processes active at each of the nodes in the line network. Of course, this is a rough approximation because inter-route interactions in the mesh network make exact modeling of the arrival processes of flows to the nodes in the route difficult. In this section, we will study a line network outside of the framework of a mesh, and assign approximate arrival processes to each of the routes in the network. We hope that this this will provide some analytical insight into the issues faced by the full mesh problem.

As stated in the basic OFS model description in Chapter 5, we need to define a state space and state transitions for any OFS model of interest. For the line network, we now introduce notation that will make it simpler to specify each network model, given a network of length N nodes (and therefore N links).

For a single channel line network of length N, we can define the following parameters in order to manipulate the model for study:

N : Number of intermediate nodes

μ : Service rate of all calls/flows

$\Lambda(i,j)$: NxN matrix, of arrival rates of customers from node i to node j

λ_i : Aggregate arrival rate of customers (to any destination) at node i

These quantities are analogous to the route sets, arrival and holding parameters defined in Chapter 5, but specified in a different format for convenience. This definition allows for a parameterized arrival process for each node, length pair via the Λ matrix. Each entry $\Lambda(i,j)$ in the matrix represents the rate of calls arriving to the route beginning at node i destined for node j, obviously of length (i-j+1). The calls have a time duration that is exponentially distributed with a parameter μ We now define states and transitions for this single-channel line network model, as well as admissibility conditions. These are a specialization of the basic OFS model presented in Chapter 5.

6.1.1 State Space Model

We define a state space for the model and *admissibility* of a state as follows.

$\Phi(t,i,j)$: NxN matrix, element i,j = n if route from i to j contains n flows at time t; 0 otherwise

Since we are using the single channel assumption for these definitions, the only admissible value of n is 1. By occupied, we mean that a single flow occupies the route in the network from node i to node j (i.e. $\Phi(t,i,j) = 1$).

We define a demand transformation, given $\Phi(t,i,j)$:

$D[\Phi] = D^1 D^2 ... D^N$: N-vector representing the number of calls active at each node in Φ

$$D^i = \sum_{1 \le j \le i} \sum_{i \le k \le N} \Phi(t,j,k)$$

The free variables in the above sum j,k range from $[1,i]$, and $[j,N]$ respectively. This makes the sum over all routes that begin before the node i and terminate past node i. Summing the elements of Φ over these routes, yields the number of active calls at node i. The demand transformation contains one component for each node

in the network and describes the number of active calls in the the state Φ. We can now define the state of the system S at time t to be:

$$S(t) = \{\Phi(t, i, j)\}$$

The state S is evidently a single Φ. We will need to describe S as a set in later parts of this chapter, but for now it can be a singleton. We can now define admissibility of a state of the scheduled system:

$$S : \text{admissible} \iff \Phi \text{ admissible}$$

$$\Phi : \text{admissible} \iff D^i[\Phi] = D^i \leq 1 \quad \forall \quad 1 \leq i \leq N$$

The second definition says that a state Φ is admissible iff all of the demands that it places on the links, i.e. the elements of the $D[\Phi]$ transformation are less than one. This is due to the single channel assumption. We will only consider admissible states of the network in the analysis of this section.

6.1.2 Transition Model

Given definitions for states of the system we can define a Markov Chain that describes transitions between these states. The rates for these transitions are defined by the Λ and μ parameters in the traffic model. Define two state transitions exiting the state S, $\tau_D(i, j)$ which is a departure transition and $\tau_A(i, j)$, an arrival transition as follows:

$$S \rightarrow_{\tau_D(i,j)} S' \text{where } S' = \{\Phi^{S'}\}$$

$$S \rightarrow_{\tau_A(i,j)} S'' \text{where } S'' = \{\Phi^{S''}\}$$

Departure Transition

We say that the departure transition $\tau_D(i, j)$ from state S to state S' exists if the following three conditions are met:

1. $\Phi^S(i,j) = 1$

2. $\Phi^{S'}(i,j) = 0$

3. $\Phi^S(k,l) = \Phi^{S'}(k,l) \forall k,l, 1 \leq k,l \leq N, k \neq i, l \neq j$

The first condition ensures that the call from i to j exists in the initial state S. The second condition ensures that in the final state, the call has departed. Finally the third condition forces all other state variables to be the same in the two states. The rate of the departure transition $\tau_D(i,j)$ is evidently μ corresponding to the service rate of all calls.

Arrival Transition

The arrival transitions conditions are simply the inverses of the departure ones for this system,

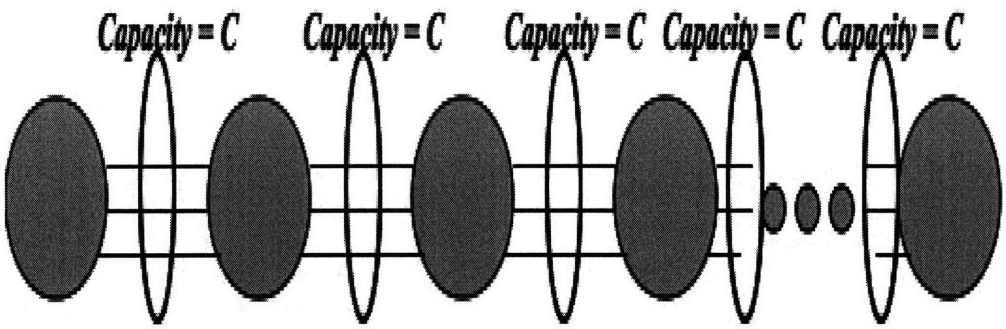
1. $\Phi^S(i,j) = 0$

2. $\Phi^{S''}(i,j) = 1$

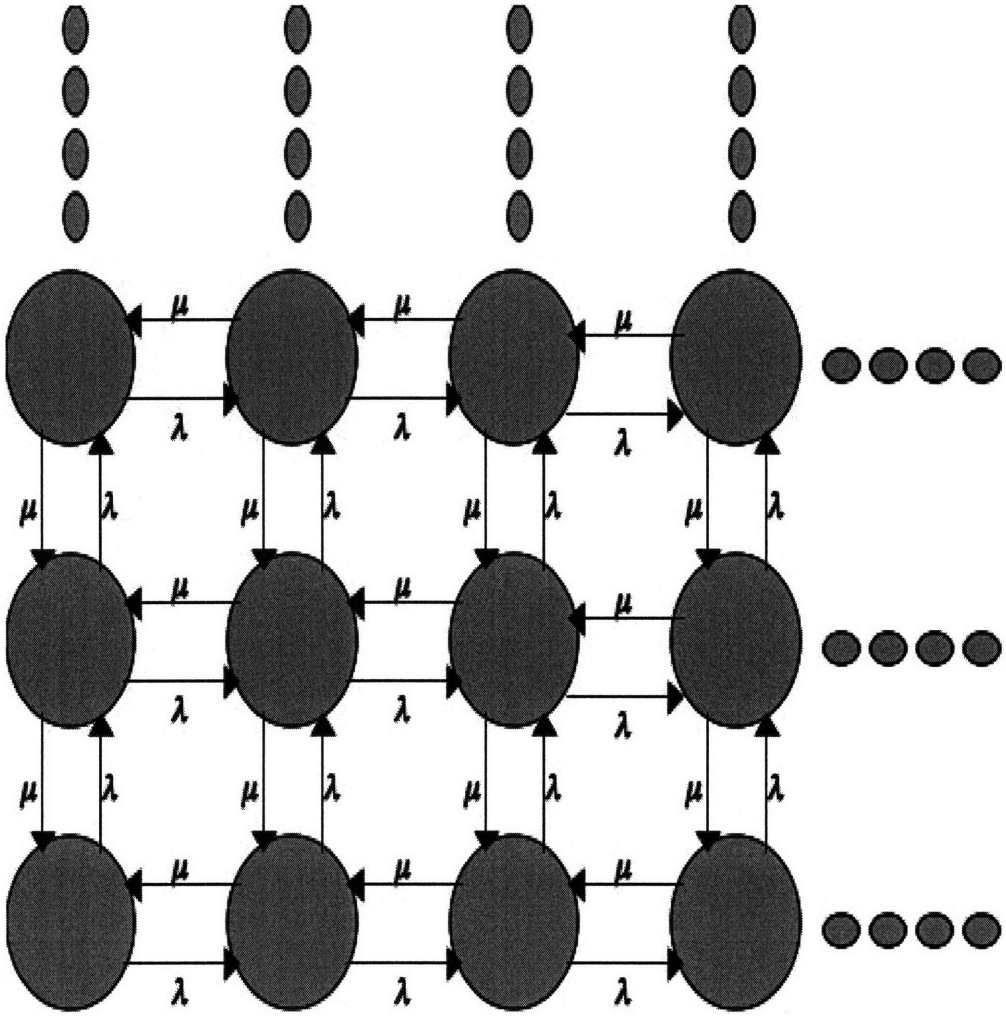
3. $\Phi^S(k,l) = \Phi^{S''}(k,l) \forall k,l, 1 \leq k,l \leq N, k \neq i, l \neq j$

4. $\Phi^{S''}$ admissible

The first condition ensures that the call from i to j does not exist in the initial state S. The second condition ensures that in the final state, the call does exist. The third condition forces all other state variables to be the same in the two states. The final condition ensures that the final state is admissible as described earlier in the chapter The rate of the arrival transition $\tau_A(i,j)$ is evidently $\Lambda(i,j)$, corresponding to the arrival process of calls from node i to node j. Note that in our definitions, a state transition can only involve a single event, be it an arrival or a departure.



(a) Circuit-switched network example



(b) Infinite 2D Markov chain example

Figure 6-2: Network and Markov chain illustrations for analysis development

6.1.3 Analysis Description

The state model described in subsection 6.1.3 is a single-channel line network model that is Markov, due to the memoryless properties of the transition distribution functions (exponential). The model is also that of a circuit switched network. [4] provides insight and analysis of this model and provides a computationally efficient method of calculating its stationary distribution. We now describe this analysis, and prove several of its properties. This analysis will yield our results for the single-channel line network analysis and will also be useful for more complex models we present later in this chapter. At the end of the development, we convert the analysis results in [4] to our notation, so that we can calculate our results.

Figure 6-2(a) shows an example of a circuit-switched network. Each link has capacity C and we assume that there are $|R|$ routes in the network. The routes overlap at various links, and there fore dependences are present between the number of calls on each route in steady state. We assume arrival processes of calls and call service processes at each route r_i to be exponentially distributed with rates λ_{r_i} and μ_{r_i} respectively. Note that our single channel line network takes this form with capacity $C = 1$. We seek the stationary distribution of the Markov process that defines the state of calls in the network.

To proceed with the analysis, temporarily assume that the link capacities are infinite, that is, $C = \infty$. Then the route-to-route overlaps become irrelevant, the number of calls active at any route is mutually independent of any other route. In this case, the routes behave as mutually independent $M/M/\infty$ queues [7]. Given the arrival and service processes of calls, the state of the network is described by a multi-dimensional ($|R| - dimensional$) Markov chain. Figure 6-2(b) is an example of such a chain, of dimension 2. The chain represents the state of the system, with transitions for arrivals (of rate λ) and departures (of rate μ) to each of the active routes. There are several important claims on this Markov process which we note here:

- The Markov chain is a multi-dimensional birth-death process, and each dimen-

sion represents a single-dimensional birth-death process. Also, each dimension is mutually independent. Justification: This follows from the fact that each dimension of the process represents a route and each route is mutually independent.

- The Markov chain satisfies the detailed balance equations. Justification: Define i, j to be states in the system, P_{ij}, P_{ji} to be the transition rates between the two states. Finally, define π_i, π_j to be the stationary probability of the two states. Then, the detailed balance equations state:

$$P_{ij}\pi_i = P_{ji}\pi_j \quad \forall \text{ states}$$

  This follows from the fact that birth-death processes satisfy the detailed balance equations [7].

- The Markov chain is time-reversible. Justification: This follows from the fact that a Markov process that satisfies the detailed balance equations is said to be a reversible Markov process or reversible Markov chain [8].

Based on these justified claims, we can analyze the stationary distribution of the Markov chain.

First, we define the following terms:

- $|R|$ - The number of active routes in the system.
- $\overline{r}$ - A vector of routes active in the system,
- $\overline{n}$ - A vector of non-negative integers representing the number of calls active at each route in the system. This can also be termed a *state* of the system.
- r_i - Element i of $\overline{r}$.
- n_i - Element i of $\overline{n}$.
- λ_i - The arrival rate of calls to route i.

- μ_i - The service rate of calls on route i.

We have stated that each route behaves as an $M/M/\infty$ queue, so the stationary distribution $P_{r_i}(n)$ of having n_i calls active on route r_i , with arrival and service rates λ_{r_i}, μ_{r_i} is:

$$P_{r_i}(n_i) = \frac{p_{r_i 0} \times (\frac{\lambda_{r_i}}{\mu_{r_i}})^{n_i}}{n_i!}$$

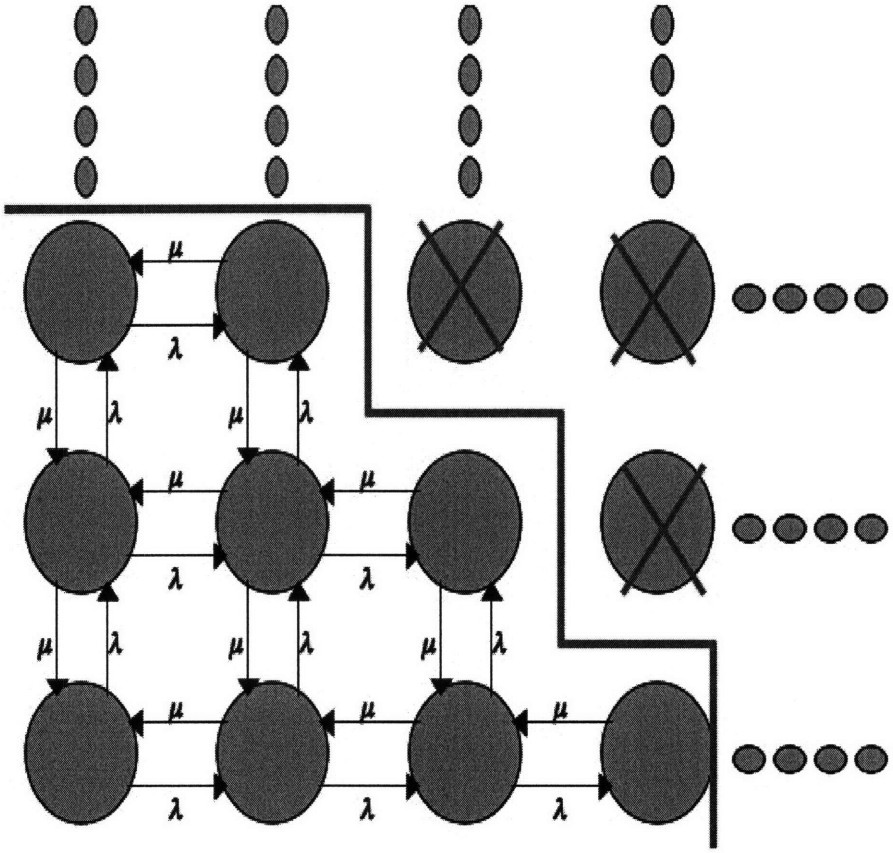
Here. $p_{r_i 0}$ is the probability of zero calls active at route r_i. This formula comes directly from fundamental results on queuing results in [7].

Because of independence of routes, the stationary distribution of calls active in all of the states will take on a product form, with one term for each of the active routes present in the model. Define $P_{\overline{r}}(\overline{n})$ to be the stationary distribution of the assignment of calls $\overline{n}$ to $\overline{r}$. Note that the elements n_i of the vector $\overline{n}$ can take on any value zero or greater, because the capacity of routes is infinite:

$$P_{\overline{r}}(\overline{n}) = \prod_{i=1}^{|R|} \frac{p_{r_i 0} \times (\frac{\lambda_{r_i}}{\mu_{r_i}})^{n_i}}{n_i!}$$

This development applies to the infinite capacity case. What remains is to apply capacity constraints to the network and therefore the objective Markov chain. Figure 6-3(a) shows visually the results of applying capacity constraints to the network. The result is a *truncated* Markov chain, partitioning the original state space, or set of $\overline{n}$, into *admissible* and *inadmissible* states w.r.t. the capacity constraints placed on the links. Admissible states are defined to be states that obey the capacity constraints, i.e. all links have occupancy less than C. Otherwise the state is inadmissible. In Figure 6-3(a) uses a thick red line to demarcate the border between admissible and inadmissible state. Note that all transitions between inadmissible and admissible states have been removed. In the truncated chain, the stationary probability of inadmissible states is clearly zero, i.e. those states cannot be entered since they violate capacity constraints.

We now analyze the stationary probability of the admissible states of the truncated chain. We first need to define a quantity G that represents the sum of the stationary



(a) Truncation of of infinite Markov chain using capacity constraints

Figure 6-3: Network and Markov chain illustrations for analysis development

probabilities of the admissible states in the infinite chain.

$$G = \sum_{\overline{n} \in admissible} \prod_{1 \leq i \leq |R|} \frac{p_{r_i 0} \times (\frac{\lambda_{r_i}}{\mu_{r_i}})^{n_i}}{n_i!}$$

Here, the sum is over all admissible states, and the thing being summed is the stationary probability of each of those states, as derived above. Again, $p_{r_i 0}$ is the probability of zero calls active at route r_i. The insight provided by [4] is that the stationary probability of admissible states is the probability of the state in the non-truncated chain, scaled by the factor G defined above. We first present the formula for the distribution and then immediately prove that it is true:

$$P_{\overline{n} \in admissible} = \frac{\prod_{i=1}^{|R|} \frac{p_{r_i 0} \times (\frac{\lambda_{r_i}}{\mu_{r_i}})^{n_i}}{n_i!}}{G} \qquad (1)$$

$$P_{\overline{n} \notin admissible} = 0$$

Note that in this formula, the $p_{r_i 0}$ terms will cancel, so we need not be concerned with them in our analysis. In order to show that this is indeed the distribution, we need to show two things. First, the distribution must sum to unity which is shown by the following:

$$\sum_{n \in admissible} \frac{\prod_{i=1}^{|R|} \frac{p_{r_i 0} \times (\frac{\lambda_{r_i}}{\mu_{r_i}})^{n_i}}{n_i!}}{G} = \frac{G}{G} = 1$$

Second, we show that the new distribution satisfies the detailed balance equations, and thus is the stationary distribution of the truncated chain.

Consider two admissible states n^i, n^j and the associated transitions between them P_{ij}, P_{ji}. Further, define π^i, π^j to be the stationary probability of the states n^i, n^j in the original (infinite) Markov chain. Define $\pi^{i\prime}, \pi^{j\prime}$ to be the stationary probability of the states n^i, n^j in the truncated chain (recall that both states are admissible). By our formula (1), above, we know that $\pi^{j\prime} = \frac{\pi^j}{G}$ and $\pi^{i\prime} = \frac{\pi^i}{G}$

We know from earlier analysis that the infinite chain satisfies the detailed balance equations, that is:

$$P_{ij}\pi^i = P_{ji}\pi^j \quad quad(2)$$

What we need to show is that the stationary probabilities of the truncated chain satisfy the detailed balance equations. This is shown as follows, first we divide both sides of (2) by G:

$$P_{ij}\frac{\pi^i}{G} = P_{ji}\frac{\pi^j}{G}$$

This shows:

$$P_{ij}\pi^{i\prime} = P_{ji}\pi^{j\prime}$$

Or that the detailed balanced equations hold for the newly formulated stationary distribution. Note that this development depended on the fact that $G > 0$, but if $G = 0$, there are no admissible states with positive probability, so we preclude this degenerate case.

In showing that the truncated chain satisfies the detailed balance equations we have also shown the following two lemmas which will be useful for subsequent analysis:

Lemma 6.1: *The Markov chain induced by a circuit-switched network is time reversible*

This follows directly from the fact that the Markov chain obeys the detailed balance equations. We can therefore consider the original process which operates in forward time and the *reversed process* that which operates in reverse time.

Lemma 6.2: *The state of the system is independent of the arrival process to the routes, which are governed by the* λ_i

By Lemma 6.1 (reversibility), the future arrivals of the reversed process is the departure process of the original process. Looking at reversed time, the state of the system at any time t is independent of any future arrivals. Now looking at the original process, this fact shows that the state at time t is independent of previous arrivals. This shows Lemma 6.2.

6.1.4 Single Channel Line Network Analysis

We now turn our attention to the single channel line network as described in Section 6.1. This network is actually a circuit switched network with capacity $C = 1$ therefore, it is amenable to the analysis in the previous subsection, with notational changes. In our single channel model, we defined a state (see list above) as $\Phi(t, i, j)$ and routes were defined by i, j, or the originating and terminating nodes, respectively. The arrival rates to the route defined by nodes i, j is $\lambda_{i,j}$ and the service rate of calls (flows) was fixed at μ. Also the number of nodes was defined to be N. Substituting this notation into the formulas above we get the following formulas for the line network model:

$$G = \sum_{\Phi(t,i,j) \in admissible} \prod_{1 \leq i \leq N, i \leq j \leq N} \frac{p_{0i,j} \times (\frac{\lambda_{i,j}}{\mu})^{\Phi(t,i,j)}}{\Phi(t,i,j)!}$$

The admissible states then retain retain product form steady state probabilities as follows (note that the $p_{0i,j}$ terms will cancel):

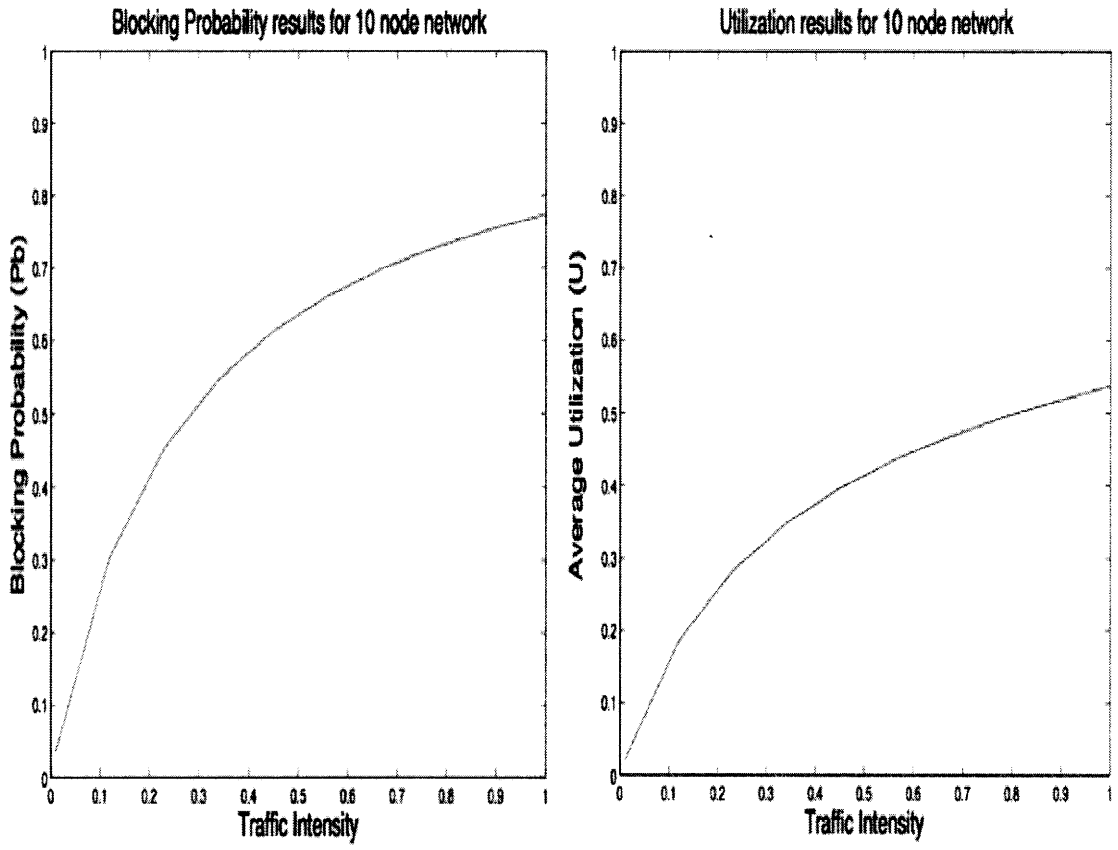
$$p(\Phi(t,i,j) \in admissible) = \frac{\prod_{1 \leq i \leq N, i \leq j \leq N} \frac{p_{0i,j} \times (\frac{\lambda_{i,j}}{\mu})^{\Phi(t,i,j)}}{\Phi(t,i,j)!}}{G}$$

$$p(\Phi(t,i,j) \notin admissible) = 0$$

Where $p(\Phi(t,i,j))$ is the stationary probability of the state $\Phi(t,i,j)$. We use these formulas to calculate the results of the following subsection.

6.1.5 Single-channel Line Network Results

For this model, using the above described techniques, we performed numerical analysis using the MATLAB programming tool. Using this analysis setup, we can numerically calculate the steady state distribution for a line network with N nodes. From this distribution, we can calculate P_b and U as defined earlier. Figure 6-4(a)(b) show blocking and utilization results for an N=10 node line network. In this case μ is fixed at 1 and $0.1 \leq \lambda \leq 1$ is varied to increase overall traffic intensity along the Y axis



(a) Blocking results

(b) Utilization results

Figure 6-4: Results of analysis for single-channel line network

(recall traffic intensity is $\frac{\lambda}{\mu}$. As expected, U and P_b continue to increase as traffic intensity increases, but remarkably for a traffic intensity of one, P_b is slightly under 0.8, while utilization is slightly above 0.5. This shows that even with a high arrival rate, optical resources in the network are under-utilized while incoming arrivals are rarely admitted. This poor performance situation is unacceptable if OFS is to be efficient. This issue is also independent of control signaling scheme used for optical connections for unscheduled connections.

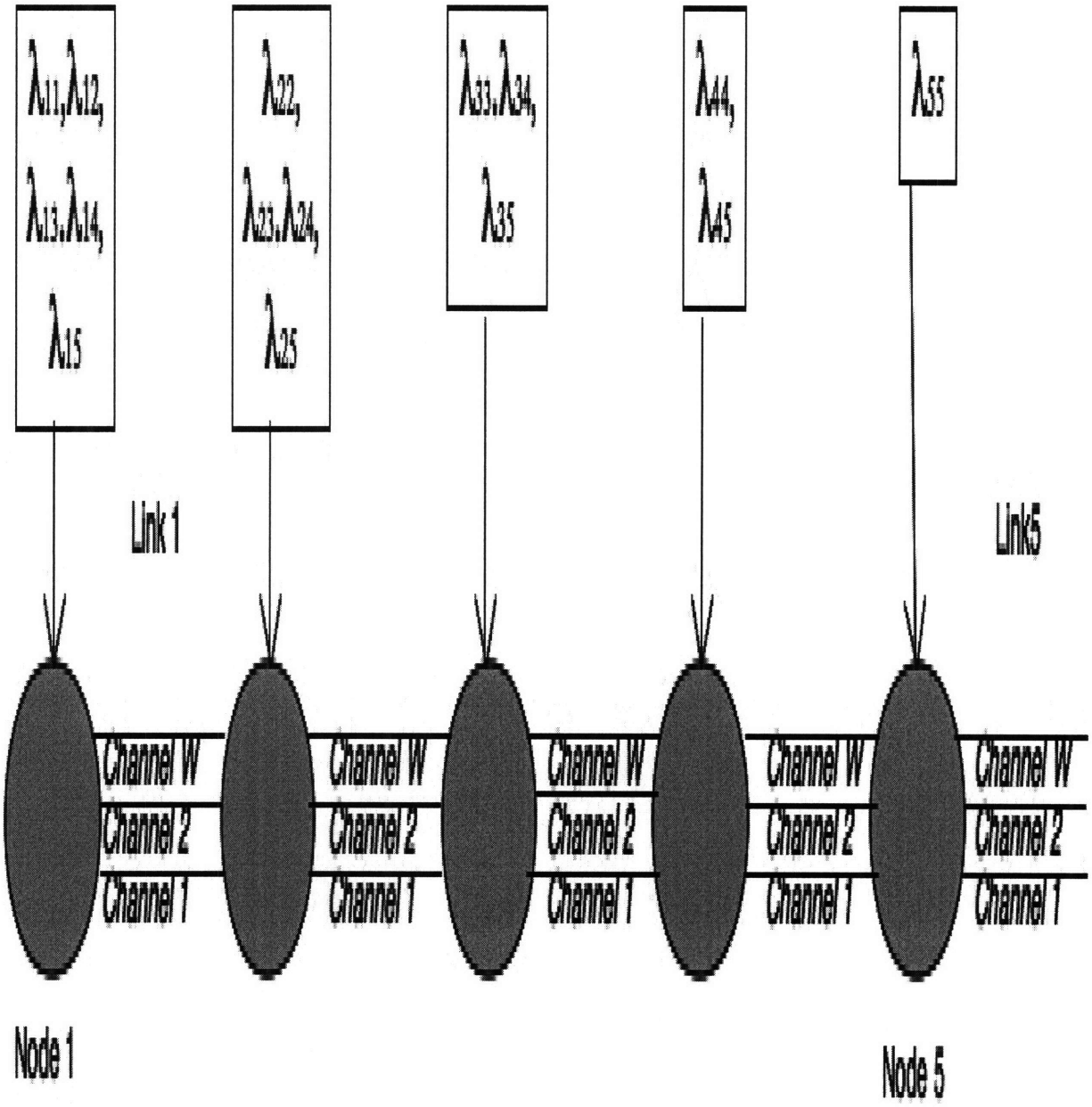
6.2 OFS in a Multiple-Channel Line Network

Qualitatively speaking, we see that the performance of the OFS line network with a single-channel assumption does not appear to be acceptable in either the sense of user blocking or network component utilization. In order to investigate this further, we take away the single-channel assumption for the next OFS model to be studied. It is possible that multiple channels may provide some measure of performance improvement, but note that the advent of multiple channels increases network capacity and resources needed to implement the network. We account for this in the multi-channel analysis that follows.

Figure 6-5(a) illustrates a multiple-channel line network. It is similar to the previous line network model except that there are W channels (e.q. wavelengths) available for OFS calls. In the figure, the channels are labeled from 1 to W from bottom to top. As we shall see, we will define the same call arrival and departure process for this network as for the single channel line network. The key difference in the two models is that the multi-channel model must have a *wavelength assignment* approach to determine which wavelength each flow is assigned to. We now proceed to the model description.

6.2.1 State Space Model

The multiple-channel OFS state model will be comprised of a set of single-channel state variables. These variables will all be time-dependent as before. Recall that



(a) Five node multi-channel line network

Figure 6-5: Multi-channel line network model, with W channels

for an individual channel, we can represent the call occupancy using the $\Phi(t,i,j)$ matrix. In the multiple-channel model, we define one of these matrices per channel and label it with the index. Therefore the occupancy of the ith channel in the network is represented by the variable $\Phi_i(t,i,j)$. Given this definition, we can now define a state of a W-channel line network model:

$$S(t) = \{\Phi_1(t,i,j), \Phi_2(t,i,j), \Phi_3(t,i,j)...\Phi_W(t,i,j)\}$$

The indices of the state variables correspond to the channel number they represent. The arrival processes to the system remain defined by the Λ matrix as defined earlier.

6.2.2 Transition Model

Arrival Transition

In defining an arrival transition to the multiple channel line network OFS model, we recognize that an arrival can be admitted to any channel whose links corresponding to the arrivals route are free. Thus, in order to define arrival transitions we must perform *wavelength assignment.* We choose the First-Fit (FF) wavelength assignment strategy to be applied to arrivals to the system [42]. This assignment strategy assigns a flow to the lowest-index channel that contains all the free links required for the flow. If no such channel exists in the network the call is discarded and considered blocked. In other words the channel space in the network is searched in increasing order for a channel that accommodate the flow. If the Wth channel is searched and there is still not the resources needed, the call is blocked. Note that this strategy enforces wavelength continuity as each admitted flow must continue on the same channel on all the links it occupies.

More formally, assume that we have a flow arrival to the system from link i to link j. We say that this arrival will be admitted to channel q and cause the system to go from state S to S'. This represents a valid arrival transition $S \rightarrow_{\tau_A(i,j)} S'$ iff all of the following conditions are met:

1. $\Phi_q^S(i,j) = 0$

2. $\Phi_q^{S'}(i,j) = 1$

3. $\Phi_q^S(k,l) = \Phi_q^{S'}(k,l) \forall k,l, 1 \leq k,l \leq N, k \neq i, l \neq j$

4. $\Phi_q^{S'}$ admissible

5. $\Phi_u^{S'}$ inadmissible $1 \leq u < q$

6. $\Phi_u^{S'} = \Phi_u^S \forall 1 \leq u \leq W, q \neq u$

The first three conditions above ensure that the can be fit into channel q in state S and that it is active in state S'. The fourth condition states that the arrival is admissible into channel q. The fifth condition states that channel q is the lowest index channel that the flow is admissible for. Finally, the sixth condition states that all other channels state do not change.

Departure Transition

Assume a departure of a flow from node i to j in channel l, and define it to be $S \rightarrow_{\tau_{D(l)}(i,j)} S''$.

This transition from a channel with index l looks very similar to that of the single channel model with the added index. We list the existence conditions for this transition here:

1. $\Phi_l^S(i,j) = 1$

2. $\Phi_l^{S''}(i,j) = 0$

3. $\Phi_l^S(k,l) = \Phi_l^{S''}(k,l) \forall k,l, 1 \leq k,l \leq N, k \neq i, l \neq j$

4. $\Phi_q^{S''} = \Phi_q^S \forall 1 \leq q \leq W, l \neq q$

The final condition states that all other channel's state variables are constant through the state transition.

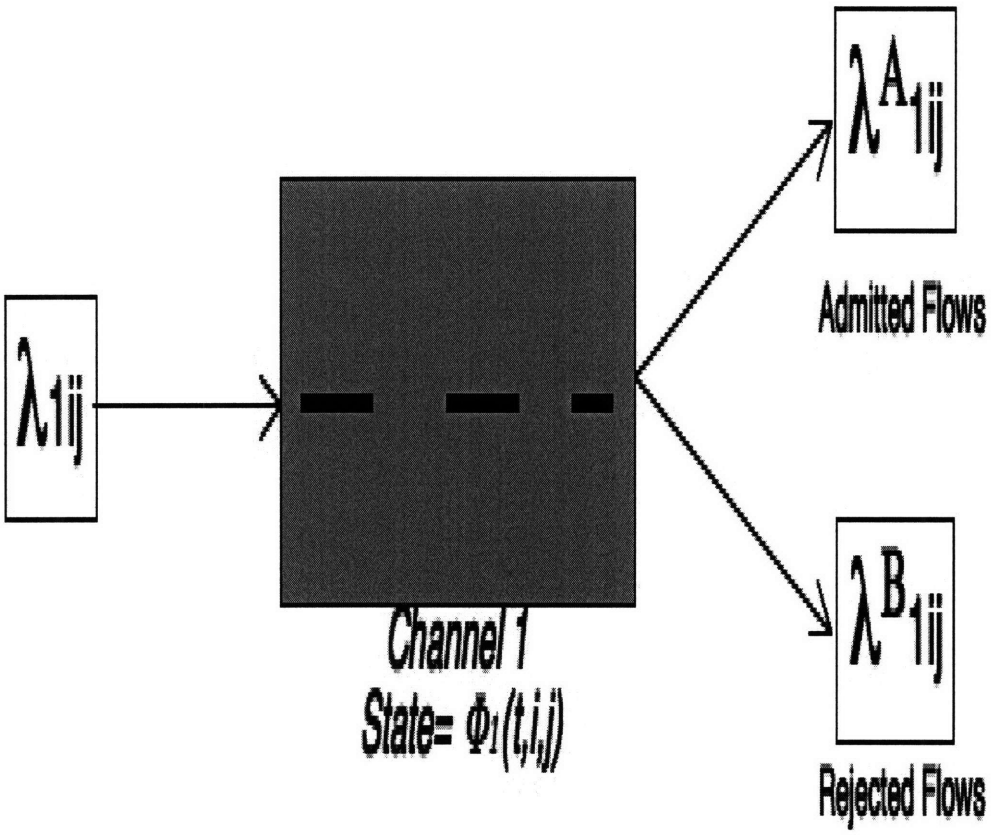
6.2.3 Analysis Description

We have defined a Markov Chain for the multi-channel line network OFS model using FF wavelength assignment. This model is more complex and expansive than the single channel line network, because of the added dimension of the channel selection. Solving for a stationary distribution of such a model explicitly is daunting, except for very small cases.

We have designed analysis that will yield the stationary distribution of this model, without resorting to solving the chain explicitly. It builds on the analysis of the previous section for the single channel line network.

The analysis proceeds in an iterative fashion according to the actions of the FF algorithm. We know that in the FF algorithm, all arrivals will attempt to occupy their intended route in channel 1 first before proceeding to 2, then to 3 and so on. Now, consider an arbitrary, non-zero element of $\Lambda(i,j)$ and call it λ_{1ij}, where the 1 indicates that arrivals arrive at channel 1 first. This arrival process was defined to be independent of the state of the network. Reversibility results show that the state of the system is independent of the arrival process, making the process and system state mutually independent. Considering an arbitrary arrival generated by λ_{1ij} it may be blocked or admitted to the network according to the arrival transition rules presented earlier for the multi-channel OFS system. This will depend on the time-varying state of channel one or $\Phi_1(t,i,j)$. Again, λ_{1ij} and $\Phi_1(t,i,j)$ are independent, so this is an *independent split* of λ_{1ij}, more details about this phenomenon are contained in [7]. Figure 6-6 illustrates the situation. The arrival process λ_{1ij} is split into two independent processes, which we will call the *admitted* and *blocked* process with rates λ_{1ij}^A and λ_{1ij}^B, respectively.

We proceed to show that the blocked and admitted processes are Poisson. We examine Channel 1, and recall Lemma 6.2. Channel 1 is clearly a circuit-switched network of capacity $C = 1$, and is therefore amenable to the analysis described in Subsection 6.1.3, and therefore Lemma 6.2 applies. Lemma 6.2 states that the state of the system (in this case, Channel 1) is independent of the arrival process to it,



(a) Split of arrival process

Figure 6-6: Illustration of First-fit arrival process

defined by the rate λ_{1ij}. In steady state, for a given flow arrival to Channel 1, there is some probability that it will be blocked and depending on the stationary probability of the channel state, $\Phi_1(t,i,j)$. This is therefore an independent split of the Poisson arrival process to Channel 1, resulting in the two processes admitted and blocked, represented by the rates λ^A_{1ij}, λ^B_{1ij}. By theory of split Poisson processes, these two process are also Poisson, and are independent.

The rates of these processes can be calculated as follows. We assume that we have the stationary distribution of Channel 1, i.e. $p(\Phi_1(t,k,l)$, where we have used k,l just for notational convenience. Then the rates of the resultant processes are:

$$\lambda^A_{1ij} = \lambda_{1ij} \sum_{\Phi_1(t,k,l) s.t. ADM(\Phi_1(t,k,l),i,j)=1} p(\Phi_1(t,k,l)$$

This expression sums the stationary probabilities of states of the system s.t. the state can admit the flow i,j, and multiplies the incoming arrival rate by this sum.

$$\lambda^B_{1ij} = \lambda_{1ij} \sum_{\Phi_1(t,k,l) s.t. ADM(\Phi_1(t,k,l),i,j)=0} p(\Phi_1(t,k,l)$$

This expression sums the stationary probabilities of states of the system s.t. the state cannot admit the flow i,j, and multiplies the incoming arrival rate by this sum. Therefore, after calculating the stationary distribution of Channel 1, we can calculate the rate of both the blocked and admitted process.

The above analysis suggests an iterative algorithm for calculating the stationary distribution of a multi-channel OFS line network model with W channels. Clearly, given $\Lambda(i,j)$ we can calculate the stationary distribution of Channel 1 via the same method outlined in Subsection 6.1.3. Once this distribution is obtained, we can determine $\Lambda_2(i,j)$ by calculating the blocked process of Channel 1 as outlined above. Given this arrival process we can calculate the stationary distribution of Channel 2. We can subsequently calculate each of the distributions of the channels up to W in this manner. The complexity of this algorithm is that of the single channel case, repeated W times, and splitting each of the arrival process matrices once. This is

much less computationally intensive than explicitly solving the Markov Chain for the entire monolithic model, which has a very large state space. .

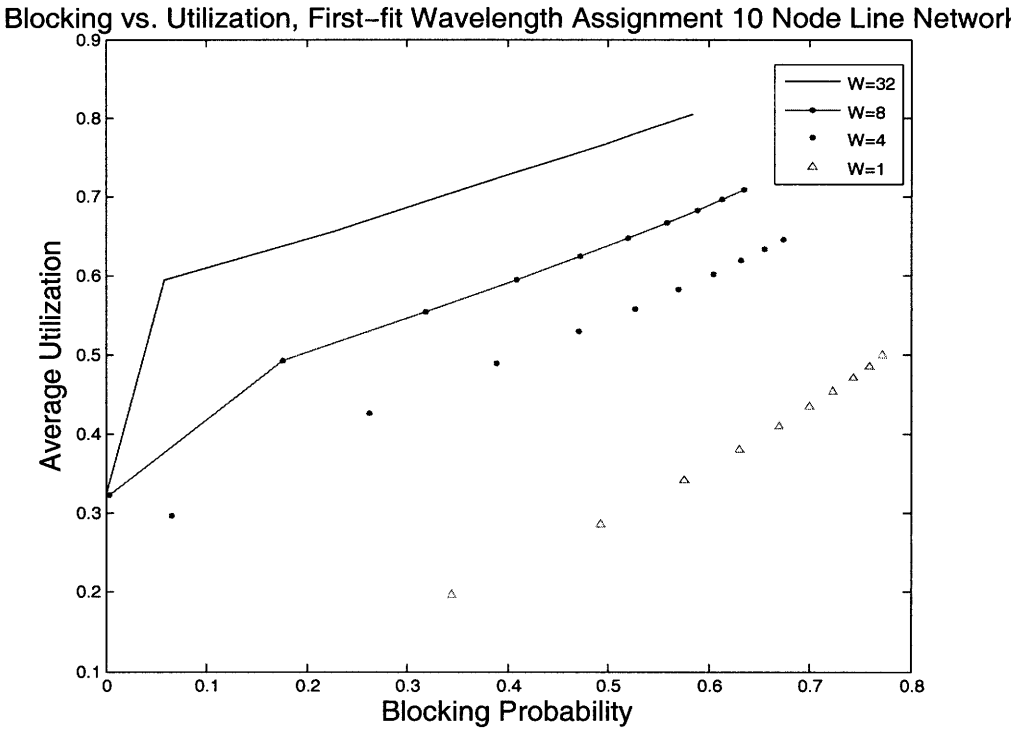
6.2.4 Multi-channel Line Network Results

Figure 6-7(a) shows the results of the analysis for a ten node line network with various numbers of channels, W. The traffic model arrival process for each of the routes in this network is a Poisson arrival process as defined earlier with the same parameter, λ, for all routes. The holding time parameter is μ for all flows as before. In the figure, the lines are created by varying λ in $0.1 \times W \leq \lambda \leq W$ in linearly equal intervals. The scaling factor of W accounts for the fact that as W increases, the network capacity increases by the same factor. Therefore, the traffic arrival rate should vary by the same factor for fair comparison.

The results in 6-7(a) confirm the single channel results shown earlier. That is, in the case W=1, the network suffers from low utilization even for a high blocking probability, just as we have seen in the previous section. The figure shows that increasing the number of wavelengths helps to alleviate this problem. The ideal location in the graph for performance is the upper-left corner where $P_b = 0, U = 1$. The results for W=4, W=8, and W=32 progress towards this in ascending order. The results for W=32 show a peak performance of approximately $P_b = .05$ while $U = .59$. At this performance level, calls are arriving at the rate $0.1 * 32 = 3.2$ per second with a holding time of 1 second. 32 wavelengths appear to be needed to support this relatively small rate of traffic, which seems excessive, given our goal of OFS taking up a small portion of the optical network resources.

6.3 OFS in a Multi-channel Mesh Network

We can use the analysis of the multi-channel network in the previous section to apply to a general mesh network such as the one shown in 6-1(b). This network is obviously more complex than the line network, and the state space is much less regular. Interactions between routes in such a network are more complicated, and not



(a) Multi-channel line network results, various W

Figure 6-7: Multi-channel network results employing First-Fit RWA strategy

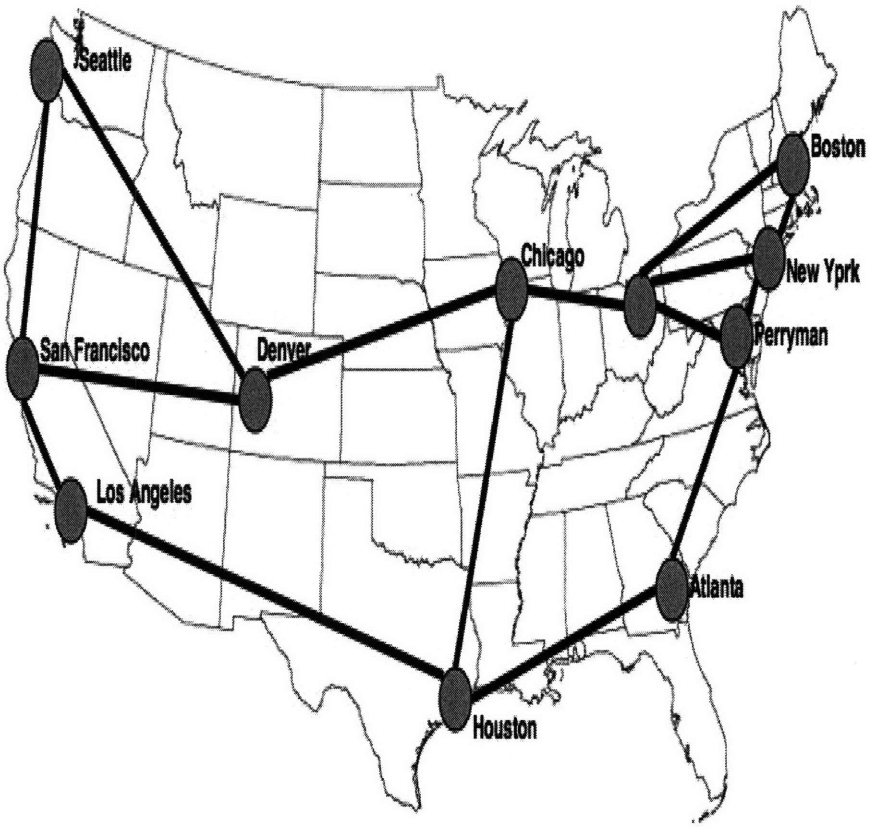
easily described. Nevertheless, the iterative form of the analysis of the multi-channel network using the first-fit wavelength selection strategy will apply to this model, and we will describe constructs that allow the analysis to proceed.

6.3.1 State Space Model

The state model for the mesh network will be a combination of the basic model described in chapter 4, and the model for a multi-channel line network. The descriptions presented here, as well as the results, will reference the mesh network shown in 6-8(a). This shows the node and links for the very high speed backbone network service (vBNS) network which is a wide-area network with nodes located in the Continental U.S. (CONUS). This links are high-rate optical links connected by IP routers at the nodes listed. The nodes are named for the cities that they are located in. We list the nodes with abbreviations and links here (reference Figure 6-8(a)), assuming that links are bidirectional.

Nodes:

- Los Angeles - LA
- San Francisco - SF
- Seattle - SEA
- Chicago - CHI
- Cleveland - CLE
- Boston - BOS
- New York - NY
- Perryman - PER
- Denver - DEN
- Atlanta - ATL



(a) vBNS CONUS topology

Figure 6-8: vBNS mesh network, connected by bidirectional fiber links

- Houston - HOU

Bidirectional Links:

- LA-SF
- SEA-DEN
- SF-DEN
- SF-SEA
- DEN-CHI
- LA-HOU
- HOU-CHI
- HOU-ATL
- CHI-CLE
- CLE-BOS
- CLE-NY
- CLE-PER
- BOS-NY
- NY-PER
- PER-ATL

We define a route set on this network similar to that of chapter 5, using this set of links to form it. Note that there are a large number of acyclic links between pairs of nodes in this network. However, our routeset will be the result of applying shortest path routing [43] to the vBNS network. We assume that each node has a non-blocking optical cross-connect with no wavelength conversion. As in chapter 5,

a route is described by a series of contiguous links, and a routeset is described by a set of routes. As an example, we list all the routes that originate from the node SEA in the vBNS network to all other nodes.

Routes (from SEA):

- SEA-SF
- SEA-DEN
- SEA-DEN-CHI
- SEA-SF-LA
- SEA-SF-LA-HOU-ATL
- SEA-SF-LA-HOU
- SEA-DEN-CHI-CLE
- SEA-DEN-CHI-CLE-PER
- SEA-DEN-CHI-CLE-BOS
- SEA-DEN-CHI-CLE-NY

This example route set consists of shortest path routes from SEA to all other nodes. Note that for particular destination node, more than one shortest path may exit; we have chosen one. A similar list can be constructed for every node in the network, and these together form an all-to-all shortest path routeset for the network.

Recall from chapter 5 that the state of a network with a defined routeset can be defined as a binary vector $S(t)$ with length $|R|$, the cardinality of the routeset. This applies to the network model here. As described in chapter 5, not all values of $S(t)$ are admissible, and our analysis will only deal with admissible states. In the case of a mesh network (as opposed to a line network), admissibility is more complex and harder to determine. So, we have a state of this network as discussed in chapter 5:

$$\overline{S}(t) =< b_1 b_2 b_3 ... b_{|R|} >$$

Similarly, we define arrival processes of OFS flows to these routes:

$$\overline{S}(t) =< \lambda_1 \lambda_2 \lambda_3 ... \lambda_{|R|} >$$

We define these vectors for the vBNS network shown in Figure 6-8.

For description purposes, we now assume a single channel in the network, we have designed a construct called a *conflict matrix*. This is a two dimensional binary matrix of dimensions $|R|x|R|$. An entry in this matrix (i, j) is one if and only if the two routes r_i, r_j conflict by sharing a directional link. For example, if we define r_i to be SEA-SF, r_j to be SEA-SF-LA and r_k to be SEA-DEN. Then the entry (i, j) would be set to one since they share the link from SEA-SF. The entry (i, k) would be zero since r_i and r_k do not share a link.

In Subsection 6.3.2, we will present a description of the technique for finding a conflict matrix and then admissible states of a mesh network efficiently. Note that this development will be for a single channel network, since for our results, we can use the iterative algorithm described in Subsection 6.2.3

Given that we can find the set of admissible states for the mesh network system with a single channel using the above outlined techniques, we can use the iterative approach to analysis of the first-fit RWA algorithm. This was outlined in the previous section. Here we simply add indexing to the state variables to produce per-channel state variables index with channel number (from $1...W$). Therefore for a channel i, we define a state variable $\overline{S}_i(T)$

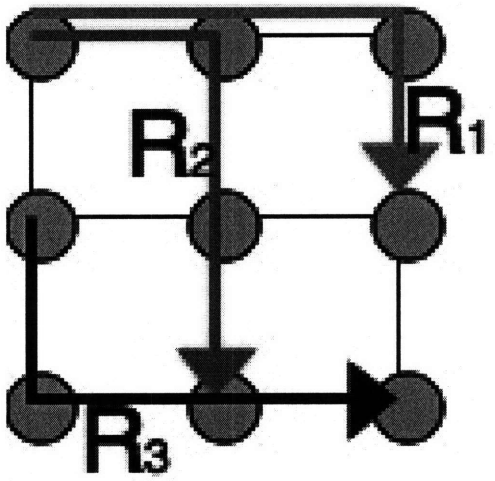
$$\overline{S}_i(t) =< b_1 b_2 b_3 ... b_{|R|} >$$

6.3.2 Algorithm Description

The problem we are faced with is finding all admissible states (of active flows) in a single-channel mesh network. It turns out that this problem is very computationally

difficult. For example, the if there are $|R|$ routes in the network, then there are a total of $2^{|R|}$ combinations of flows that can be assigned to them. Luckily not all these combinations of flows are admissible, otherwise the state space itself would be too large to do any analysis.

We have devised a computationally efficient approach to finding the admissible states for a problem problem as large as the vBNS network. It uses the concept of a *conflict matrix.*



(a) Example 3x3 Mesh Network with Three Routes

	R_1	R_2	R_3
R_1	X	1	0
R_2	1	X	0
R_3	0	0	X

(b) Resultant Conflict Matrix

Figure 6-9: Example of a Conflict Matrix

The goal is to find all of the combinations of routes from the routeset that do not conflict in a single link. Consider Figure 6-9(a)(b). Figure 6-9(a) shows a 3x3 mesh network with a single channel. In the figure, we are considering 3 of the routes as our routeset, labeled R_1, R_2, R_3. Based on the links that these flows occupy, we can define a conflict matrix as shown in Figure 6-9(b). This conflict matrix is a 2D matrix whose axes are both all the routes in the network. We place a 1 position R_i, R_j if the routes R_i, R_j conflict in any link, otherwise, we place a zero in the position. For example, in Figure 6-9(a), R_2, R_1 conflict on the link in the leftmost, top most link. Therefore, there is a 1 in the conflict matrix corresponding to R_2, R_1. Conversely R_3

does not conflict with any other route, so there are 0s in all of its rows and columns.

The conflict matrix is a key construct, because its construction is simple, and because it allows us to identify which routes conflict with the most other routes in the network. This is a simple matter of counting the number of 1s in each routes row (or column) and summing, to find which route has the greatest sum. Knowing this information suggests an efficient algorithm for finding the desired set of routes. We sketch the algorithm here with high level pseudo-code, given a mesh network of interest.

1. Given the mesh network and routeset, construct the conflict matrix.

2. If the problem is deemed to be small enough to solve directly, say if $|R|$ is less than some chosen constant, search the entire combination space for admissible combinations and return the resulting set. Otherwise, continue.

3. By counting the ones in each row of the conflict matrix, identify the "most conflicting" route in the set, call it R_i.

4. Solve the following two problems:

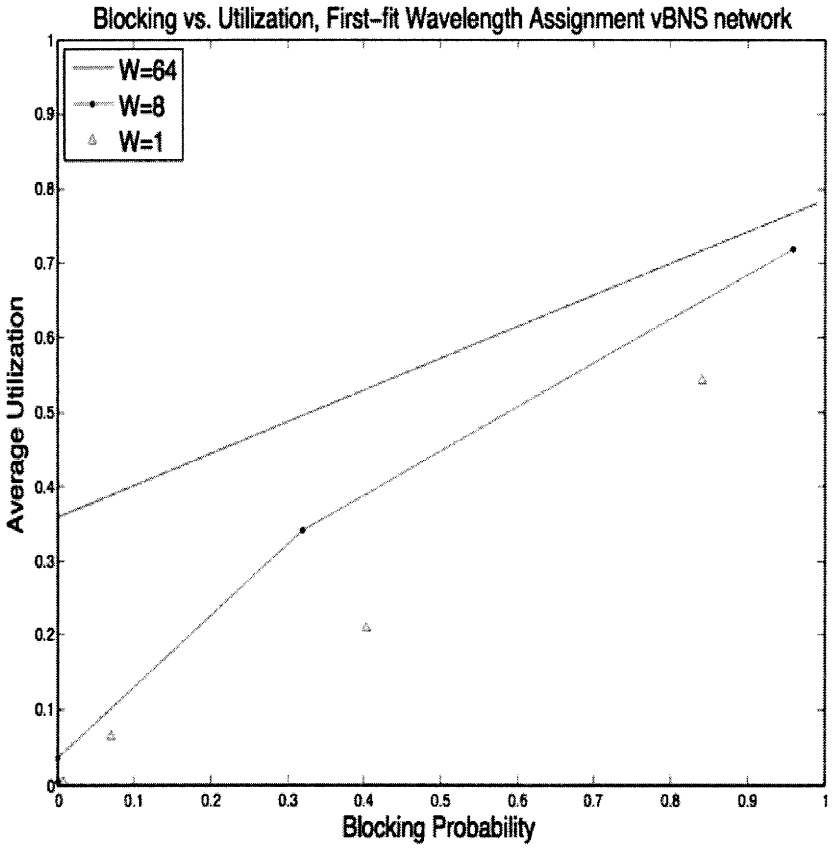
   (a) Assume that the route R_i is active. Eliminate all routes from the set that conflict with R_i. Also, eliminate all rows and columns in the conflict matrix in which R_i had a 1, including $R_i s$ row and column. Solve this new, reduced problem recursively.

   (b) Assume that the route R_i is not active. Eliminate R_i from the routeset, solve the reduced problem recursively.

5. When the two recursive problems return, their intersection should be null, since R_i is present in the combinations in one set and not the other; combine them and return the result.

We have implemented this algorithm in MATLAB to produce the results shown in the next subsection. The runtime for the vBNS network was more than two hours, but presumably much shorter than a brute force search of route combinations.

6.3.3 Multi-channel Mesh Network Results



(a) Results for multi-channel mesh network, varying arrival rates

Figure 6-10: vBNS mesh network multi-channel results

Figure 6-10 shows the results of the analysis on the vBNS network. The graph shows curves for blocking vs. utilization performance for several cases in terms of W, the number of channels per link, ranging from 1 to 64. For these results μ, the service rate of flows was fixed at 1, and λ ranged between $.01 \times W \leq \lambda \leq W$ geometrically evenly spaced.

The curves show blocking vs. utilization results for each channel number scenario. The curves look very similar to the single channel results we have seen earlier in Figure 6-4(a). The system fails to reach high utilization for a low blocking probability (upper left corner of figure). It appears that increasing the number of available channels helps

this problem somewhat, but blocking remains high when utilization is $\geq 30\%$.

These results are qualitatively very different from those of the single channel line network that we analyzed earlier in this Chapter. Recall that we modeled the single channel network with Poisson arrival processes to *each* of the routes in the network. In the line network, the routes conflict in a very regular, predictable way. In the mesh network, the external (i.e. exterior) and internal (i.e. interior) nodes see a very different set of arrival processes. This is due to blocking and occupancy of exterior links before calls can arrive at internal nodes. In particular, interior nodes of the mesh see arrival processes that are do not resemble Poisson processes. The presence of these interior nodes makes the performance of the two networks very different, and shows clearly that the line network is a very rough approximation of a route within a mesh network.

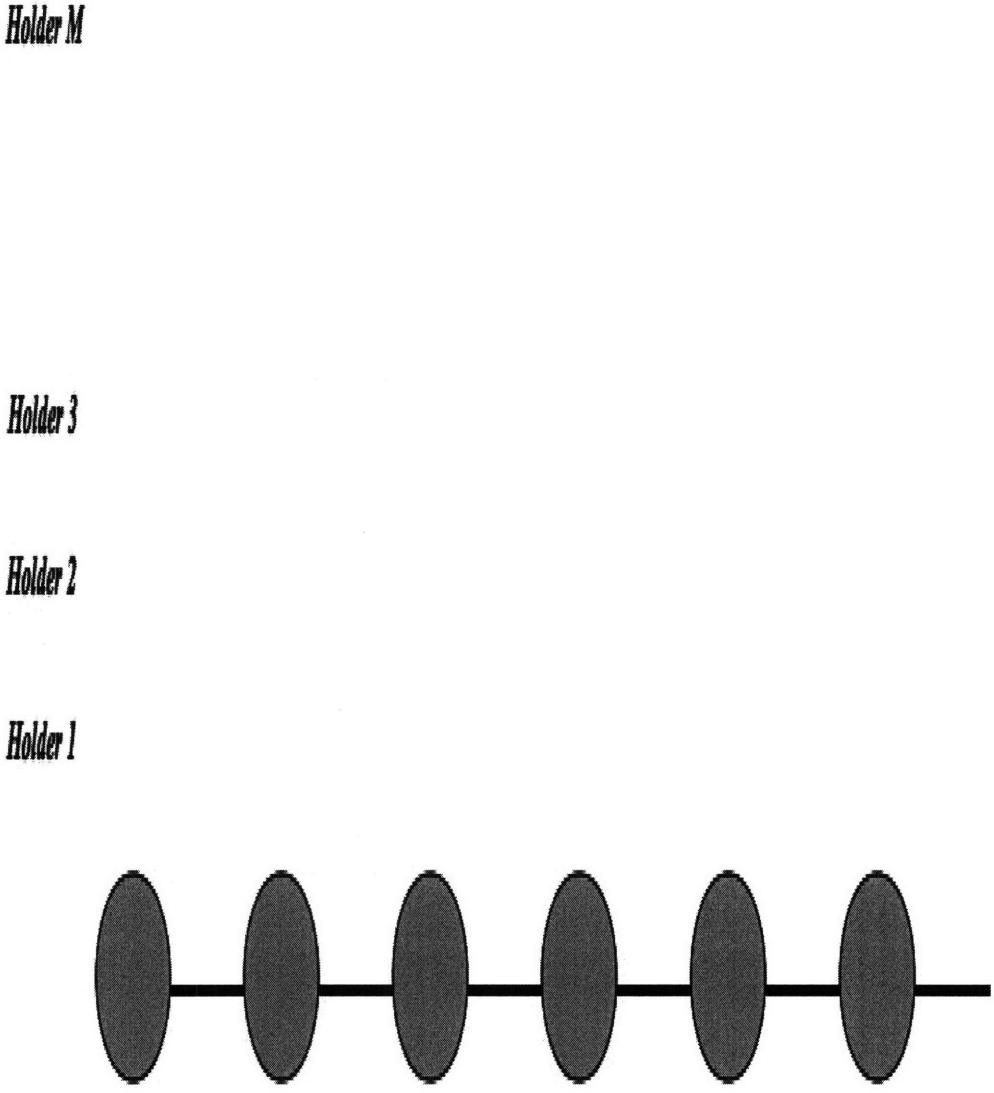
6.4 Scheduled OFS

In this section, we present numerical results for a Scheduled OFS system using a line network, similar to the single channel line network defined earlier. We show that performance is greatly enhanced by this simple scheduling scheme, at a cost of no more optical resources. Finally we perform delay analysis on the scheduled system to show that it is numerically close to the size of the scheduling horizon calls on average. For this thesis we define a scheduled approach is one that uses precise network-wide timing information to make decisions about optical resource allocation to requests.

6.4.1 Single Channel Network

In this section, we study all optical connections in a line network assuming ideal, instantaneous connection setup.

The model for study is as shown in Figure 6-11(a). The figure shows a line network similar to the single channel line network we have analyzed earlier in this Chapter. However, to the left of the figure, above the network, we notice M scheduling *holders*. These holders are used to hold incoming flow requests until resources in the network



(a) Scheduled OFS model with holders

Figure 6-11: Scheduled OFS model for analysis

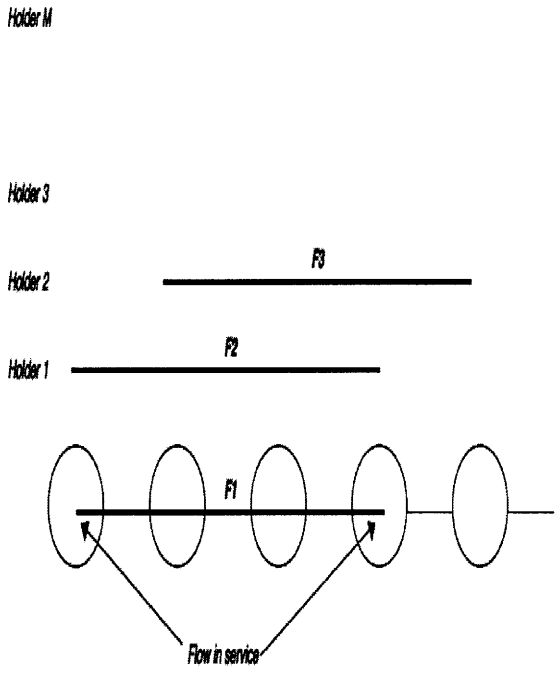
become available due to departures. Arrival and departure processes are defined as before, with exponentially distributed interarrival times and service times.

In the scheduling algorithm we use for the results of this section, arriving flows first attempt to occupy the single channel network. In the previous unscheduled OFS model, if the flow did not find network resources (i.e. links) available along its path, it was lost (i.e. blocked). In the scheduled system, however, the flow will be admitted to the lowest numbered holder that has space available for it. The holders are in this sense are virtual copies of the line network, that can hold flows for future admission into the network. Since there are a finite number of holders, it is possible that an arriving flow cannot be admitted into the network or any of the M holders. In this case it is lost (i.e. blocked), so the blocking event applies to the scheduled system as well.

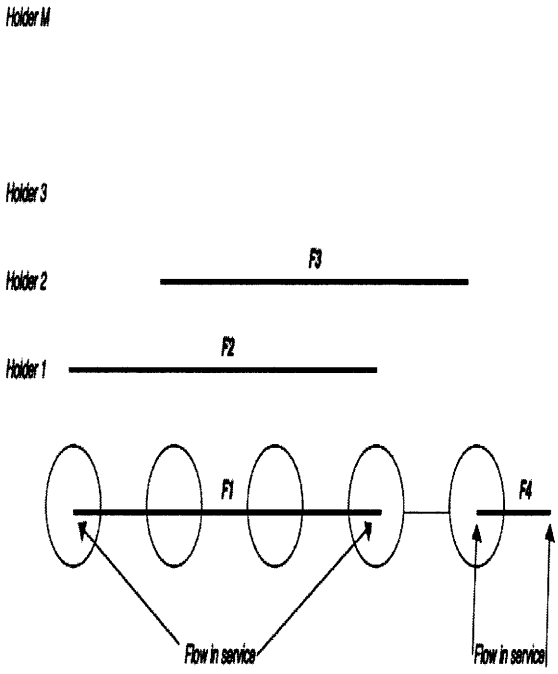
In the scheduling scheme we employ, upon departure of a flow from the network, the resources it previously held are released and we search the holders in sequential order (lowest to highest) for flows that can re-occupy the newly freed resources. Note that for a flow to be admitted to a vacated segment of the network *all* its resources must be available. This scheme is similar to FIFO, but it gives a slight advantage to flows that demand fewer resources, or equivalently have a shorter hop-count. This scheme was chosen because it is provides the benefits of scheduling but is also conceptually simple and amenable to implementation and numerical analysis. Note that holders are similar to the network in that they can only hold one networks worth of traffic. No two flows that use the same link can simultaneously occupy the same holder.

Figures 6-12(a)(b)(c)(d) provide an example of flows arriving and departing the scheduled OFS system.

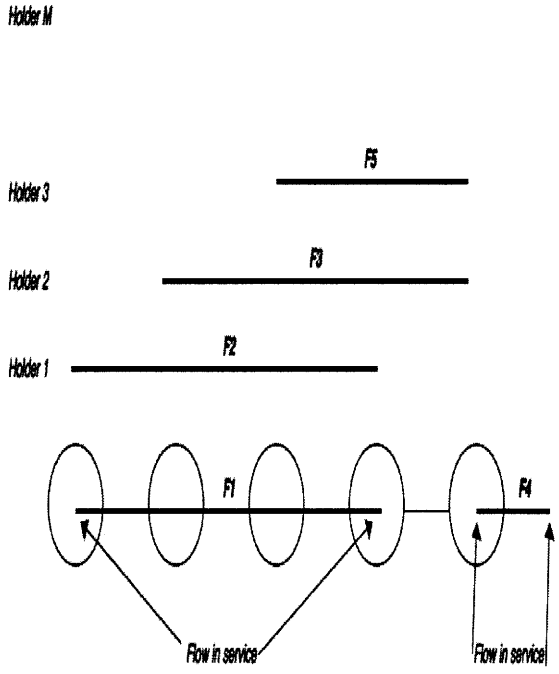
The idea of scheduling is presented visually by Figures 6-12(a)(b)(c)(d). Figure 6-12(a) shows the initial state of the line network, similar to Figure 5-1(b), with a flow occupying the first four nodes. However, Figure 6-12(a) differs in that it shows M *scheduling holders* above the network. We say that this network has a *scheduling horizon* of M.



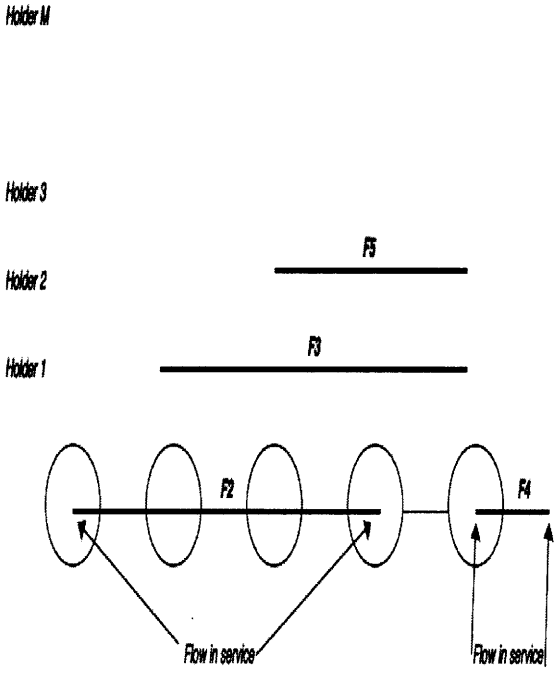
(a) Initial state



(b) Step 2: arrival of flow C4



(c) Step 3: arrival of flow C5



(d) Step 4: departure of flow C1

Figure 6-12: Modified FIFO scheduling example

We can define the following variables to describe the state of the scheduled system. For this discussion define m=0 to be the network channel.

N : Number of intermediate nodes

μ : Service rate of all calls/flows

$\Lambda(i,j)$: NxN matrix, of arrival rates of flows from node i to node j

$\Phi_m(t,i,j)$: NxN matrix, element i,j = 1 if path from i to j occupied at t in holder m; 0 otherwise

As in an earlier section, we define a demand transformation, given $\Phi_m(t,i,j)$:

$D_m[\Phi_m] = D_m^1 D_m^2 ... D_m^N$: N-vector representing the number of calls active at each node in Φ_m

$$D_m^i = \sum_{1 \le j \le i} \sum_{i \le k \le N} \Phi_m(t,j,k)$$

Each of the Φ_m describe the state of one of the holders or channel in the system. We define the state of the system S at time t to be:

$$S(t) = \{\Phi_1(t,i,j), \Phi_2(t,i,j), ..., \Phi_{m-1}(t,i,j), \Phi_m(t,i,j)\}$$

We can now define admissibility of a state of the scheduled system:

$$\Phi_m : \text{admissible} \iff D_m^i[\Phi_m] = D_m^i \le 1 \quad \forall \quad 1 \le m \quad \forall \quad 1 \le i \le N$$

We will only consider admissible states of the network in the analysis of this section. The networks in this subsection consider networks with a positive number of holders. There are many admission disciplines that can be used for this scheduling system, ranging from simple to complex. In this work, we choose to use a modified FIFO scheduling scheme. The scheme places incoming arrivals in the network if

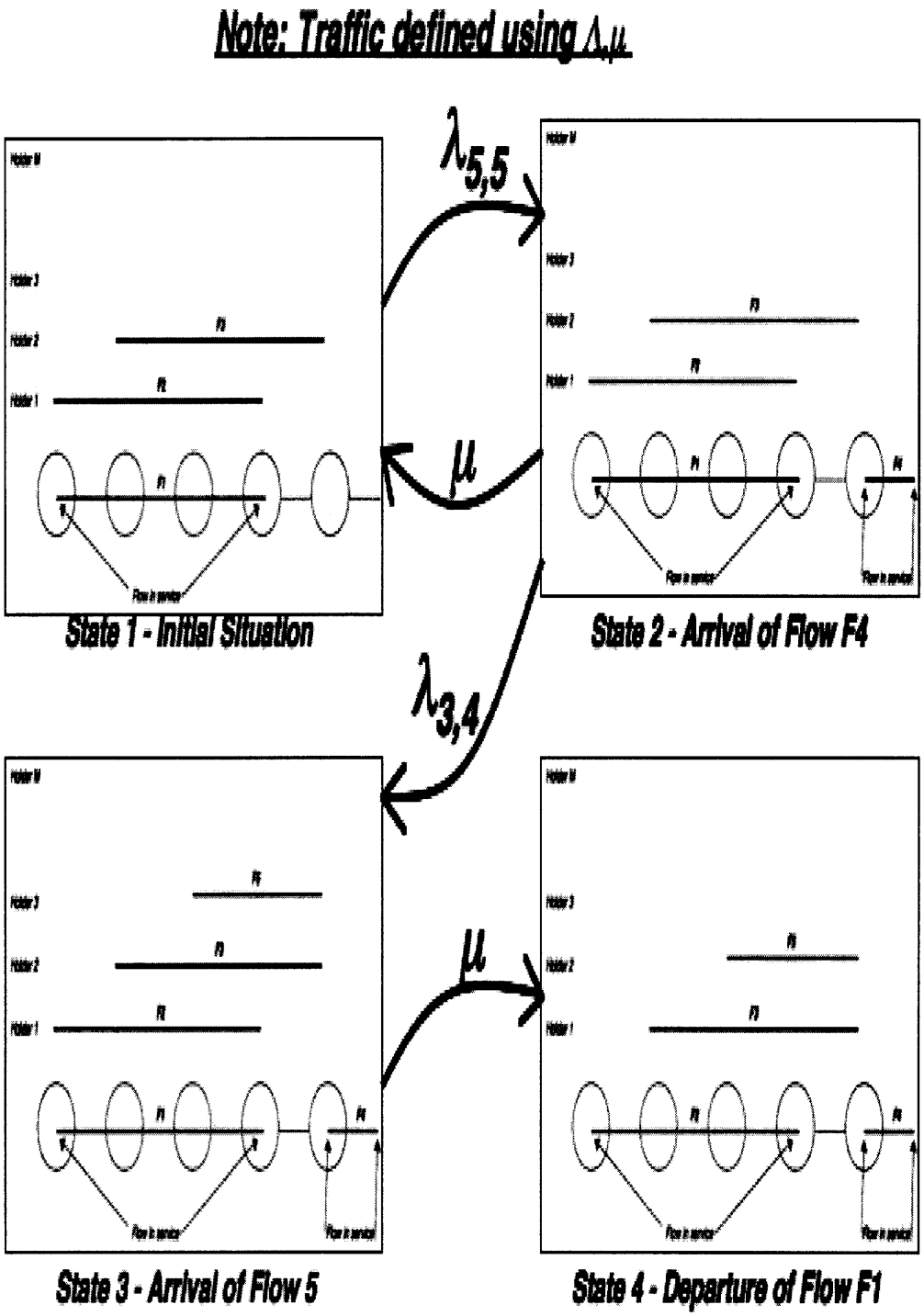
possible. If not, it places the arrival in the lowest numbered holder which has space available. Note that it is possible in this scheme for a later arriving flow to "pass" an earlier admitted flow, hence the name modified FIFO. It is reminiscent of the familiar $M/M/1/M$ queuing system, except that it is not *work conserving.* Several steps of the algorithm are illustrated in Figure 6-12. Figure 6-12(b) shows that in step 2, flow C4 arrived and is two nodes long consuming the last two nodes. Note that it has "passed" C2 and C3 both of whose resources are currently taken by C1. Figure 6-12(c) shows step 3 is the arrival of C5 which been placed in holder 3 due to resource conflicts in the network and holders 1 and 2. Finally in step 4, 6-12(d) shows that C1 has departed and C2 has moved into the network for service. C3 and C5 have advanced to holders 1 and 2 respectively.

This scheme is similar to FIFO, except that on occasion it allows later arrivals to pass an earlier one. We choose this scheme because it is tractable to analyze. We intend to show that even this simple scheduling approach can greatly improve both blocking and utilization performance. A more complex scheduling scheme may perform better.

Given our earlier definitions for the arrival and service models, that is Λ and μ, we can model the scheduled network as a Markov chain. Each state of the Markov chain is a set of occupancies of the network and the scheduling holders. Figure 6-13 shows a fragment of a Markov chain that is induced by the example given in Figure 6-12 For any finite M, this is a positive recurrent Markov chain because it is finite and aperiodic by definition. Therefore it must have a steady-state distribution P. Define the stationary probability of any particular state S to be P_S. Note that under the definition of our modified FIFO algorithm, certain states are invalid, and are not part of the chain. For example, there cannot exist a valid state with no flows in service and some flow in a scheduling holder.

We can formally define this Markov chain as follows. Define a given state of the system as:

$$S = \{\Phi_1^S \Phi_2^S \Phi_3^S ... \Phi_M^S\}$$



(a) Initial state

Figure 6-13: Markov fragment of scheduled OFS model corresponding to example

Thus, each of the Φ_i^S represents the state of holder i, (or the channel if i=0); Consider i and j to be two integers such that $1 \leq i \leq N$ and $i \leq j \leq N$.

Define two state transitions exiting the state S, $\tau_D(i,j)$ which is a departure transition and $\tau_A(i,j)$, an arrival transition as follows:

$$S \rightarrow_{\tau_D(i,j)} S' \text{where } S' = \{\Phi_1^{S'}\Phi_2^{S'}\Phi_3^{S'}...\Phi_M^{S'}\}$$

$$S \rightarrow_{\tau_A(i,j)} S'' \text{where } S'' = \{\Phi_1^{S''}\Phi_2^{S''}\Phi_3^{S''}...\Phi_M^{S''}\}$$

Departure Transition

We say that the departure transition $\tau_D(i,j)$ from state S to state S' exists if the following two conditions are met:

1. $\Phi_0^S(i,j) = 1$
2. We can construct state S' as follows:

   Algorithm 6.4.1: A(b)

$$\begin{cases} \Phi_0^S(i,j) = 0 \\ S' = \{\} \end{cases}$$

$$\textbf{for } k \leftarrow 0 \textbf{ to } M-1$$

$$\textbf{do} \begin{cases} \textbf{for } l \leftarrow k+1 \textbf{ to } M \\ \quad \textbf{do} \begin{cases} \textbf{for } kk \leftarrow 1 \textbf{ to } N \\ \quad \textbf{do} \begin{cases} \textbf{for } ll \leftarrow kk \textbf{ to } N \\ \quad \textbf{do} \begin{cases} \textbf{if } ((\Phi_l^S(kk,ll) == 1) \textbf{ and } (ADM(\Phi_k^S, kk, ll))) \\ \quad \textbf{then} \begin{cases} \Phi_k^S(kk,ll) = 1 \\ \Phi_l^S(kk,ll) = 0 \end{cases} \end{cases} \end{cases} \end{cases} \\ S' \leftarrow S' \bigcup \Phi_k^S \end{cases}$$

$$S' \leftarrow S' \bigcup \Phi_M^S$$

This pseudo-code constructs the target state S' for $\tau_D(i,j)$ coming from state S.

Arrival transition

For the description of the arrival transition that follows, we need to define the following two constructs:

$$One(i,j) = \text{NxN matrix with a 1 at element i,j and 0 elsewhere}$$

$$ADM(S,i,j) = 1 \text{ if } \exists \Phi_k \in S \text{ s.t. } \Phi_k + One(i,j) \text{ admissible 0 otherwise}$$

We can now define an arrival transition $\tau_A(i,j)$ from state S to state S'' as existing if the following two conditions are met:

1. $\exists k$ where $1 \leq k \leq M$ s.t. $\Phi_k^{S''} - \Phi_k^S = One(i,j)$
2. $k =$ min s.t. $ADM(\Phi_k^S, i, j) = 1$

Condition 1 states that exactly one call has been added to S to form S'' and that that call is from node i to node j. Condition 2 ensures that the call from i to j is places in the lowest numbered holder (or the channel) possible. These are in accordance with our scheduling discipline described pictorially above. From these two definitions, we can construct a Markov chain representing the evolution of a system given a set of input parameters

Unfortunately, the state space for the scheduled number of possible system states has order $O((N^2)^M)$. However, since OFS routes are expected to be tens of nodes long, it is instructive to solve a modest size problem, and compare it to the unscheduled approach. Two numerical methods of solution were used to solve for our results. The first is the the Power Method described in [5]. In brief, the transition rates of the irreducible, continuous-time Markov chain are used to generate a state transition rate matrix $\mathbf{Q}$. By design, this $\mathbf{Q}$ matrix is guaranteed to have an eigenvalue of zero. We therefore solve the equation:

$$\vec{v} \times Q = \vec{0}$$

This yields an eigenvector $\vec{v}$ that, when normalized to sum component-wise to one, holds the steady state probabilities of the chain. More details are available in [5] and [6].

When the scale of the state space grew to be greater than 100,000 states, computational resources for the Power Method proved to be too great. We therefore employed a Markov Monte Carlo technique to solve for the stationary distribution of these larger chains. This technique is described in detail in [44]. The basic idea is to simulate the continuous time Markov chain, using the fact that time averages equal sample averages for a finite, aperiodic Markov chain. This method is also amenable to parallelization and is distinct from other simulation techniques in that it has convergence checking. We have compared the results from Markov Monte Carlo and the Power Method for representative cases and found nearly perfect agreement.

Results for this study are in terms of two quantities: Blocking probability (P_b) and node utilization (U). The quantities are of interest to the users and designers of the network respectively. The can be defined formally as follows. Given P, the stationary distribution of a scheduled OFS system:

$$U = \sum_{\text{all states S}} P_S(\sum_{i=1}^{N} D^i(\Phi_0^S))$$

This states that U is the occupancy of the N nodes averaged over all the states in the Markov chain. In order to formally define P_b we again need the $One(i,j)$ and $ADM(S,i,j)$ constructs which we re-state here:

$$One(i,j) = \text{NxN matrix with a 1 at element i,j and 0 elsewhere}$$

$$ADM(S,i,j) = 1 \text{ if } \exists \Phi_k \in S \text{ s.t. } \Phi_k + One(i,j) \text{ admissible 0 o.w.}$$

In addition, we define number of different types (node, length combinations) of arrivals to the system at any given time as:

$$A = \sum_{1}^{N}(N - i + 1) = \frac{N^2 + N}{2}$$

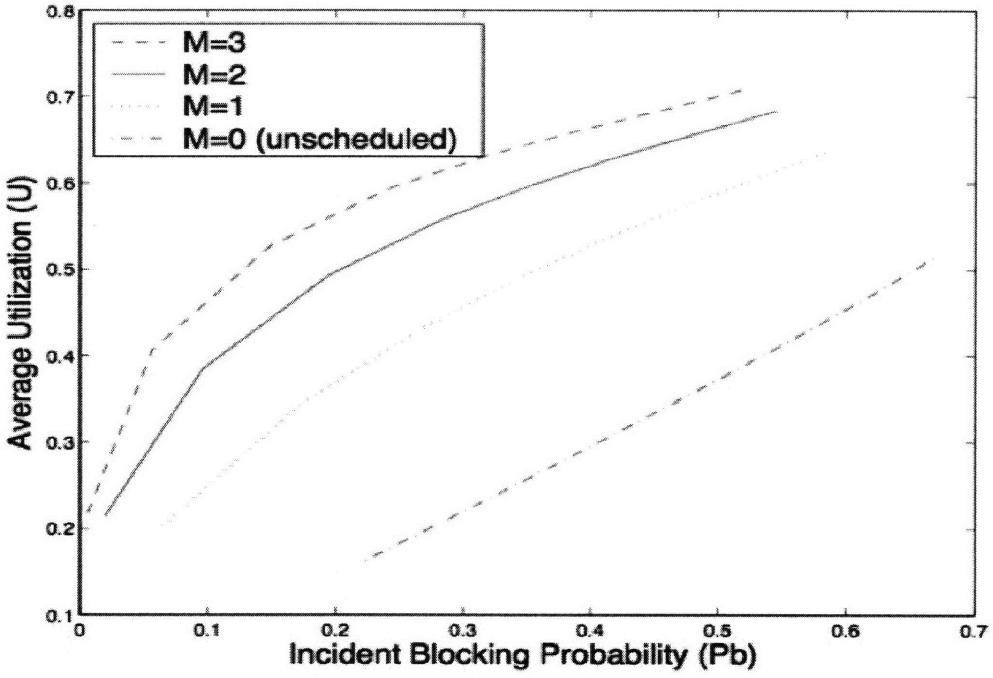
In the expression for A, we sum over i ranging from 0 to N the number of nodes. For any node i, the number of routes that originate from it is $N - i + 1$, recalling that all routes in the network move from left to right.

We assume that in steady-state, arrivals to any path are equally likely. Note that while this does not correspond to the Λ vector of arrival rates, this assumption can be modified to reflect the true arrival rates.

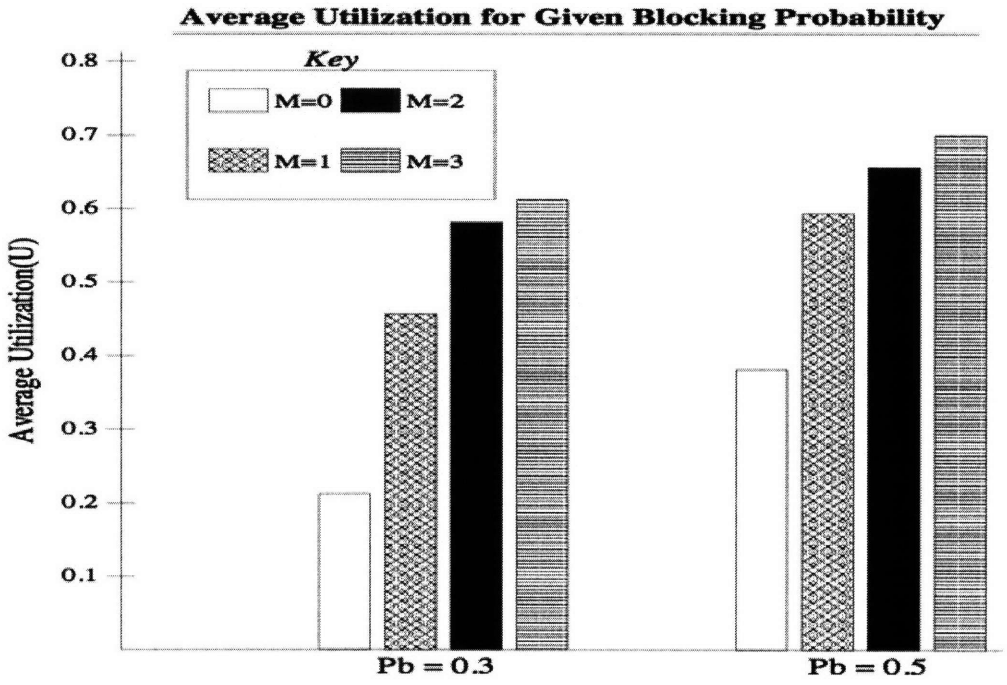
In order to derive the blocking probability of the system, we need sum over all states S and all routes in the network. In the line network, it suffices to sum over variables i, j, where $1 \leq i \leq N$ and $i \leq j \leq N$. Since each type of arrival is equally likely, $\frac{P_S}{A}$ is the probability contribution of any route to P_b, given that an arrival to the route in state S is inadmissible. For any i, j, $1 - ADM(S, i, j)$ is the indicator variable of inadmissibility of the route from i to j in S. Therefore, the expression for the blocking probability is as follows:

$$P_b = \sum_{\text{all states S}} [\sum_{i=1}^{N}(\sum_{j=i}^{N}(1 - ADM(S, i, j)) \times \frac{P_S}{A})]$$

Figure 6-14(a)(b) show the results of the numerical calculations. Figure 6-14(a) Figure 6-14(b) shows results for a 5 node network with arrivals restricted to be greater than one node in length. Each curve is a result of the traffic intensity $\frac{\lambda}{\mu}$ ranging from 0.1 to 1 linearly (μ fixed at 1). The graph is arranged to facilitate comparison of utilization for a given blocking probability for the four systems. Figure 6-14(b) shows utilization results in detail for two given blocking probabilities 0.3 and 0.5. For $P_b = 0.3$, the unscheduled network yields a utilization about .22, one holder yields slightly over .45, two holders .57, and finally three holders .62. This is almost a threefold gain in utilization from the unscheduled network to M=3. Utilization increases similarly for $P_b = 0.5$, attaining nearly a twofold increase. Although not explicitly shown, it also appears that the traffic intensity needed to produce a given blocking probability increases with increasing M as well. Both graphs show that the



(a) Results for 5 nodes, L>1



(b) Utilization Comparison

Figure 6-14: Numerical results for various M,M

relative improvement diminishes with higher M but invariably continues to improve. Overall, these results suggest that scheduling improves both blocking and utilization performance.

6.4.2 Delay Analysis

The introduction of scheduling to the OFS network also introduces delay, for accepted requests. This is because requests that are placed into the scheduling place-holders must await the freeing of resources (a channel).

While other statistics involved in the model can be calculated directly from the stationary distribution, waiting time cannot. This is because it is unclear how many paths through the Markov chain there are and what the expected delay will be. Even if this was enumerable, there do exist infinite paths through the chain (following cycles) so the direct calculation of waiting time seems difficult.

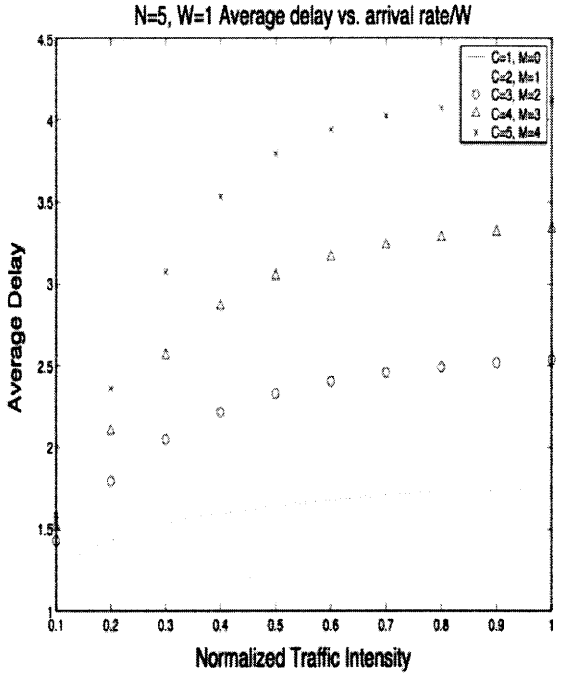
Instead we use Littles Theorem, which states:

$$E(N) = E(\Lambda) \times E(W)$$

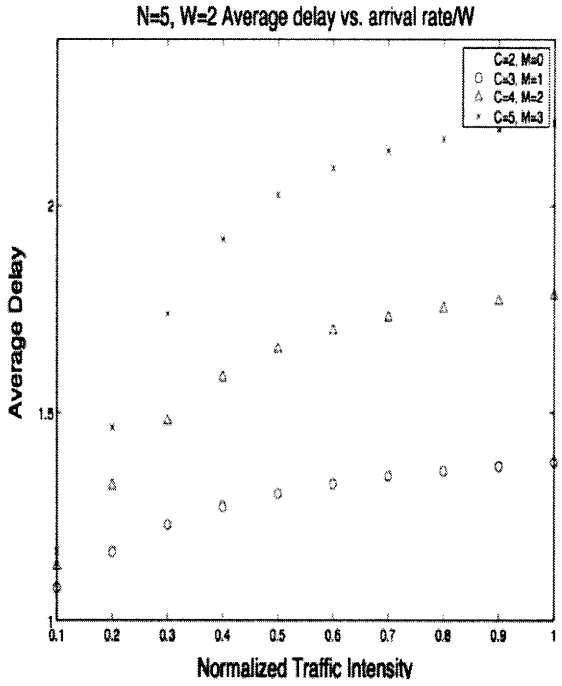
Here, $E(N)$ is the average number of flows in the system. This quantity is readily calculable from the stationary distribution of the system, and knowledge of the number of flows per state.

The remaining issue is that of Λ which is the average *arrival* rate to the system. For the purposes of this discussion we will define two quantities. The first is the *external* arrival rate to the overall system, we will call this Λ_e. The second is the actual or admitted arrival rate to the system, called Λ_a. Λ_e is the sum of N arrival processes at each of the N nodes. Λ_a is a more detailed calculation involving the stationary distribution and the admissible arrivals per state.

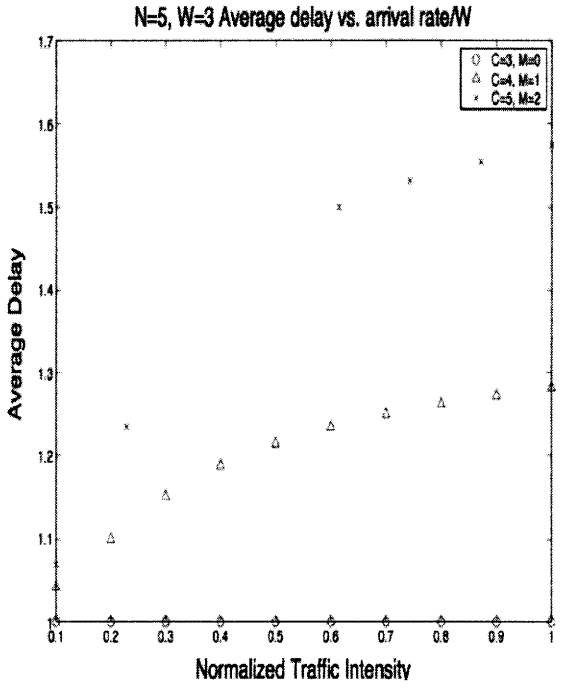
We have verified the conditions that need to be satisfied for Littles Theorem to apply to our system. 1) it is a renewal process, since renewals occur when a flow arrives at an empty system. The system is guaranteed to reach the empty state because the empty state has non-zero probability in the Markov formulation. 2)



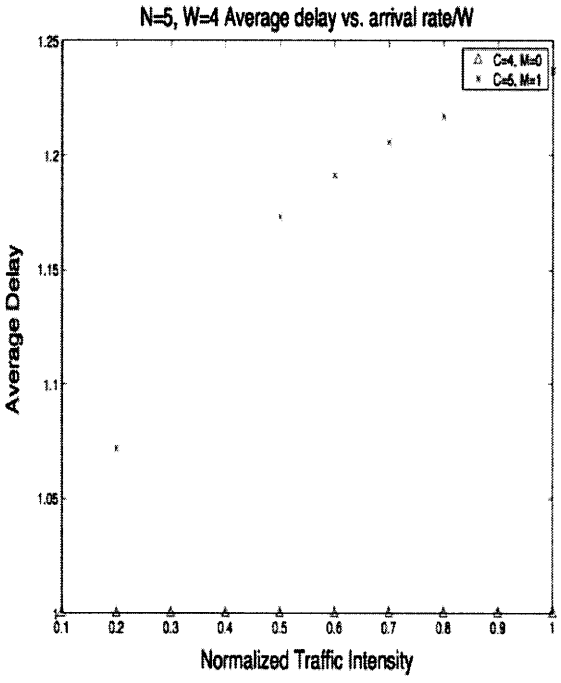
(a) Average Request Delay W=1



(b) Average Request Delay W=2



(c) Average Request Delay W=3



(d) Average Request Delay W=4

Figure 6-15: Average Request Delay for Various W,M Values

average inter-renewal time is finite, which is verified because the system is completely specified using a finite, ergodic Markov process. Therefore, the system must reach the renewal state with certainty.

The remaining question is which arrival rate to use. The proof of Littles Law states that it applies "without regard to service discipline", however experimentation showed that the external arrival rate results in incorrect calculation. It is therefore appropriate to use Λ_a for our results. Given a state S, and its stationary probability P_S, we calculate its contribution to Λ_a as follows. We scale the rates of the arrival transitions leaving state S by P_S. Summing the results of this scaling over all the arrival transitions in the Markov process will yield the overall rate of the admitted arrival process Λ_a. Formally we define:

$$\Lambda_a^S = P_S \times \sum_{\text{all states R}} l_{SR}$$

Here, l_{SR} is the rate of transition from state S to R, due to an *arrival only.* Given this calculation, Λ_a is:

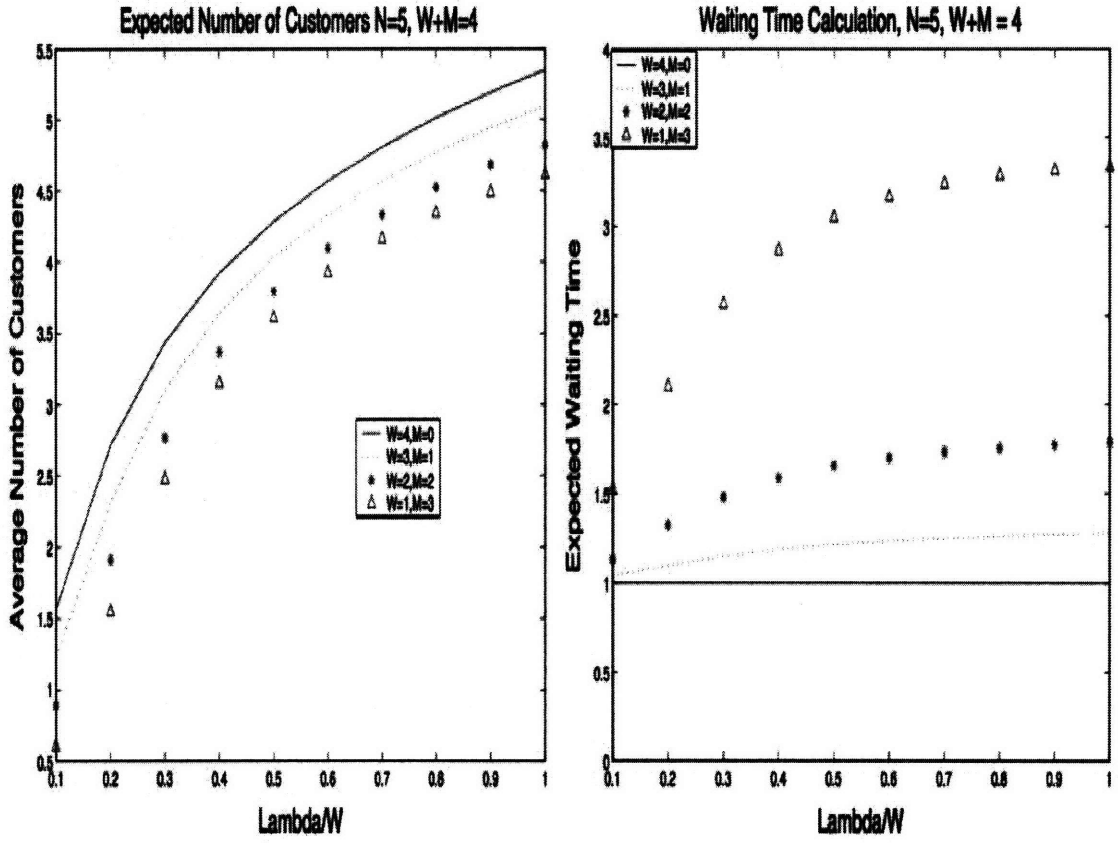
$$\Lambda_a = \sum_{\text{all states s}} \Lambda_a^s$$

As described above. We use Λ_a in Littles Theorem along with the experimentally found value of N, the number in the system to calculate W, the average delay of flows.

This is essentially the expected arrival rate to the system calculated over the stationary distribution of the associated Markov process. Experiments have verified that this is the correct arrival rate to use for Littles Theorem to apply.

Using the above calculations for $E(N)$ and the correct Λ, we can use Littles Theorem to calculate the average waiting time (including service) for various values of W and M. Results for both average number of flows in the system ($E(N)$) and waiting time ($E(W)$) are shown in Figures 6-16(a)(b).

Figure 6-15 (a-d) show the delay analysis results for a five node network. Again, each graph plots results for a fixed W, with M varying within each graph such that



(a) Average Number of Flows

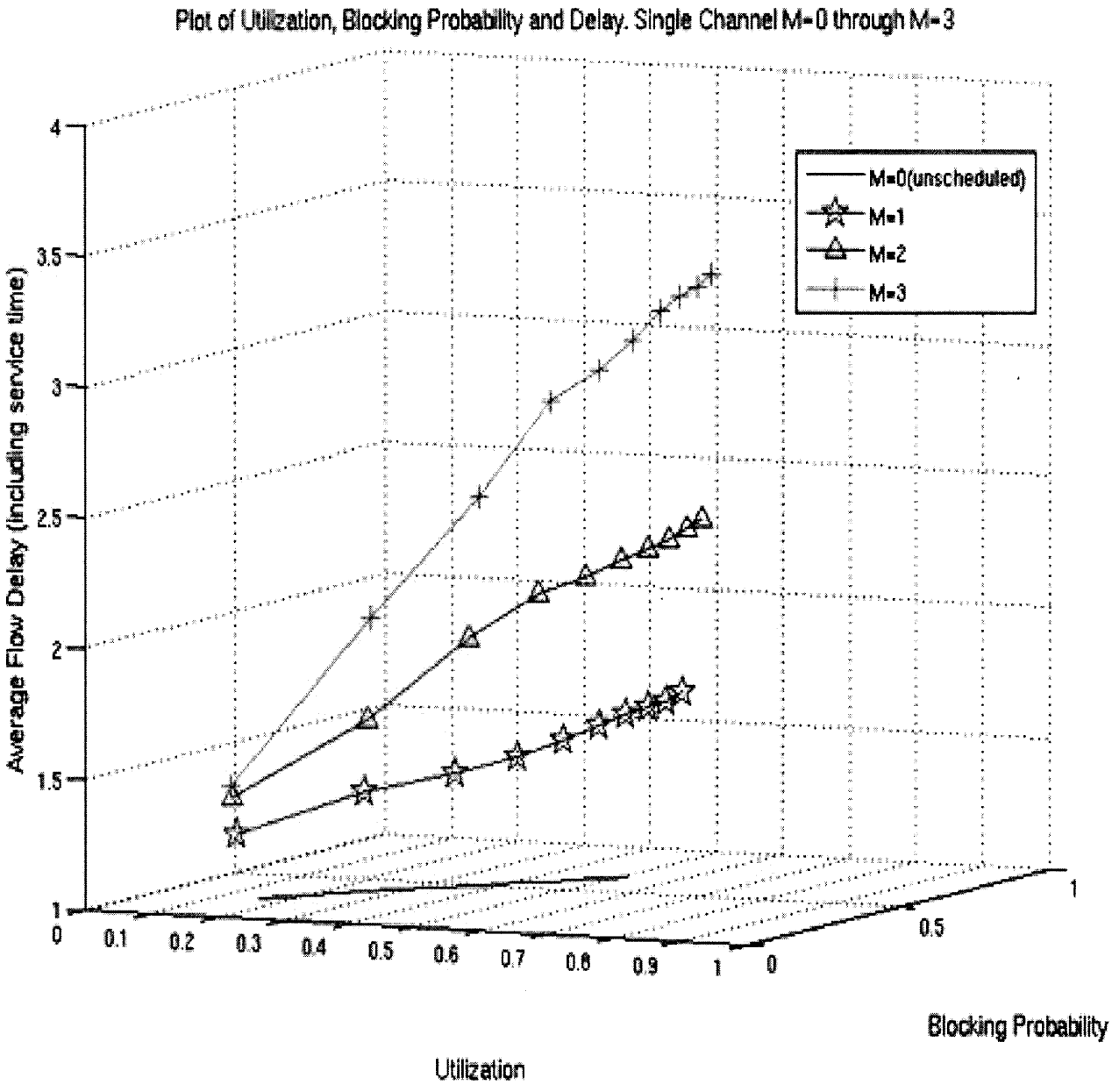
(b) Average Waiting Time

Figure 6-16: Delay Analysis Results

W+M does not exceed 5.

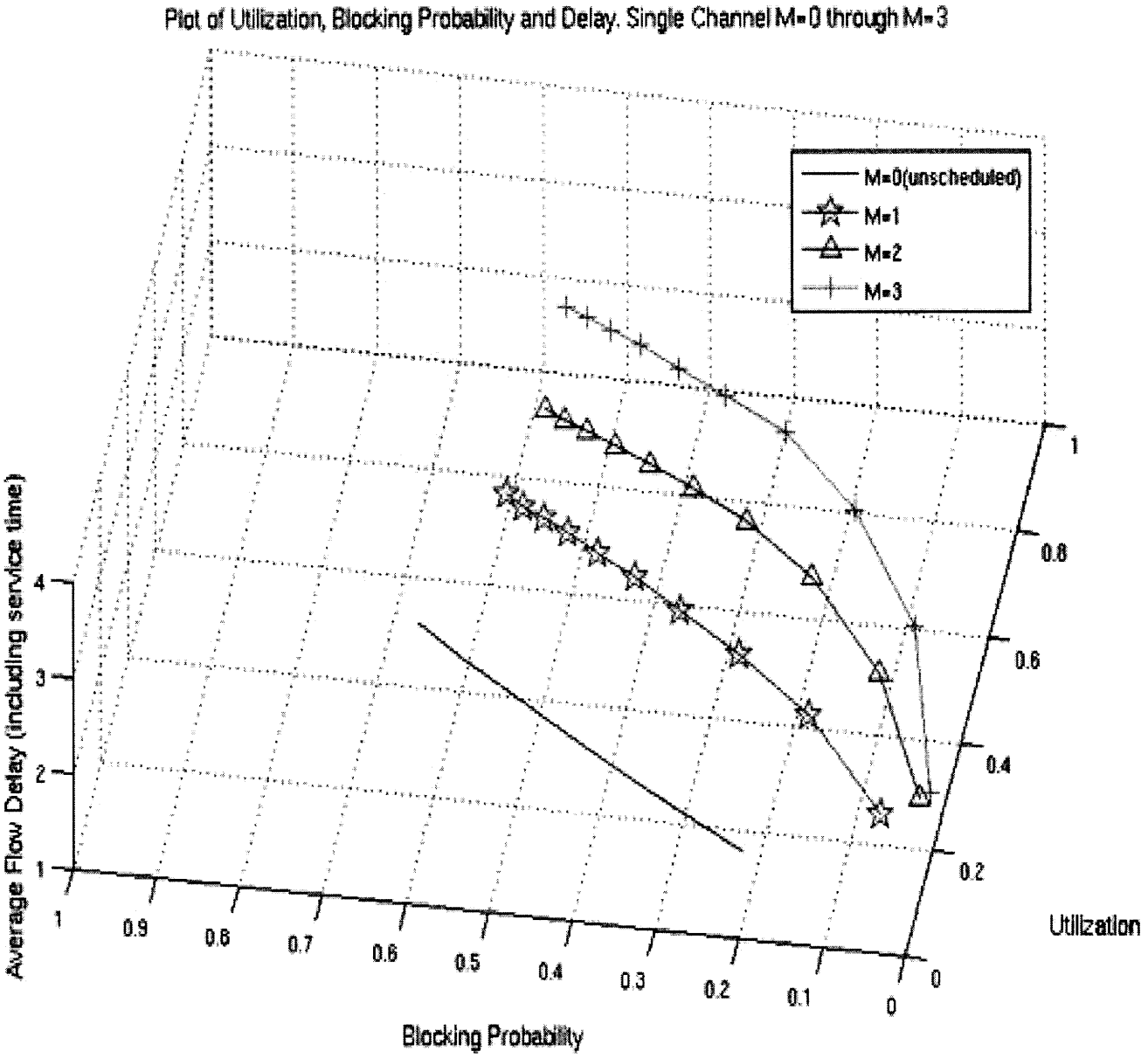
Delay seems to be on the order of M for smaller M values. Note that with the modified FIFO scheduling approach we are using, delay for larger (longer in terms of hop-length) request can be extreme. This is because a flow requesting more resources (nodes) will compete against a larger number of flows for resources. Overall, the delay is proportional to M which is a relief since, there are scenarios with non-zero probability (in the Markov model) that have very very long delay times. The concern of long average delay is mitigated somewhat by the fact that the system has a finite scheduling horizon, M. However examination of the modified FIFO scheduling scheme shows that it gives preference to calls with shorter hop duration, since these are allowed to "pass" larger flows as resources become available. However the delay results show that the additional delay to long hop-length flows does not cause the average to rise too much higher than the scheduling horizon M, in all cases we examined.

Figures 6-17(a) and 6-18(a) show a combined plots of Utilization, Blocking Probability and Average Delay. These two plots are of the exact same data, but different views. Figure 6-17(a) shows a view that focuses on the height, of the curve, namely the delay axis. Figure 6-18(a) shows a view that focuses on the Utilization vs. Blocking Probability aspect. Note that delay in the figures includes service time of the flow, so the unscheduled case has an average of 1, as expected. The curves in the figure range from unscheduled (M=0) to having a scheduling horizon of M=3. The delay view shows that the delay is approximately linear with scheduling horizon in a line network. With the highest intensity, a scheduling horizon of 3 yields a delay of 3.3 seconds, so this means that an admitted flow must wait for slightly over two flow times to acquire its needed resources. Further analysis shows that flows that request fewer resources have a much lower average delay than those that request more, given the definition of the scheduling algorithm. Figure 6-18(a) shows that the utilization performance for a given blocking probability increases with increasing M. These plots illustrate the benefits of scheduling, while showing that delay is acceptable.



(a) 3D Plot of OFS Results - View of delay axis

Figure 6-17: Utilization vs. Blocking Probability vs. Delay



(a) 3D Plot of OFS Results - View of Utilization, Blocking Probability axes

Figure 6-18: Utilization vs. Blocking Probability vs. Delay

Chapter 7

Summary

Optical Flow Switching (OFS) is an approach that provides short duration, all-optical connections to network users in a Metro or Wide area network. This technology provides benefit to both the users of the network as well as to the providers of the network. For network users, it provides a high-rate connection from end-to-end that is reserved for the users transaction. Other benefits for the users are transmission at optical rates, transparency (given optical limitations), and simplified transaction time calculation for real-time or quasi-real-time applications. For the network providers, OFS has the opportunity to provide *optical bypass* of electronic Internet routers. Transactions sent via OFS do not use resources on intermediate electronic routers, because they are sent all-optically. This provides resource savings in router port usage, router computation, and router memory.

There are a number of issues with implementation of OFS in a MAN or WAN network. WDM technology is maturing to the point where hardware is available to implement the approach. The issues that face an implementation include:

- Control Plane Implementation - Signaling, messaging and monitoring is needed to setup, teardown and field user requests for connections. This typically needs to be high performance if not real time, since OFS flows are short-lived and efficiency is important.

- Transport Layer Issues - Once connections are established, utilization of the

channel for application data depends largely on the performance of the underlying transport layers. We believe that there is an appropriate choice for transport layer for OFS that will maximize the efficient of OFS conduit usage, which is fundamentally different from traditional networks.

- Optical Signal Management - In a dynamic optical network like OFS, management of optical signal management. This basically comes down to signal-to-noise management, and involves issues of amplifier placement, regeneration and fiber distances.

- Application/Network Interface (User-Network Interface) - In order to use OFS, applications at end nodes need to have a method to request OFS connections on-demand. This involves engineering of a driver located at the end-station that provides a usable interface to application designers. Applications should remain agnostic to how the network provides OFS services.

From an implementation standpoint, OFS can be designed to be *scheduled* or *unscheduled*. For the purposes of this thesis, we defined scheduled OFS as using global network timing information to make decisions about allocation of network resources and time to send a flow. Unscheduled OFS does not make use of timing information.

In this thesis, we described two studies that explore the viability and performance of OFS:

- ONRAMP OFS Demonstration - This successful demonstration of scheduled OFS shows the viability of OFS using COTS hardware and firmware. The results show that the main obstacle to further efficiency is robustness of the transmission cards to a dynamic optical network.

- OFS Numerical Analysis - For a simple network, this analysis shows that scheduled OFS provides increased utilization and flow blocking probability when compared to an unscheduled approach. The delay introduced by a simple scheduling scheme is both acceptable, approximately linear with scheduling horizon. We

compare both to a single channel approach and to a WDM multichannel approach, and find better overall performance for scheduled OFS.

Appendix A

Analytical Results

This chapter presents theoretical analysis of the scheduled OFS system and related systems. Closed form analysis of the system is difficult, since most results require intimate knowledge of the stationary distribution of the induced Markov process. We present analysis of related systems which are tractable to analyze, and also strong numerical evidence that certain theorems hold for the general scheduled OFS model as defined in Chapter 5.

A.1 M/M/m/K Queuing System

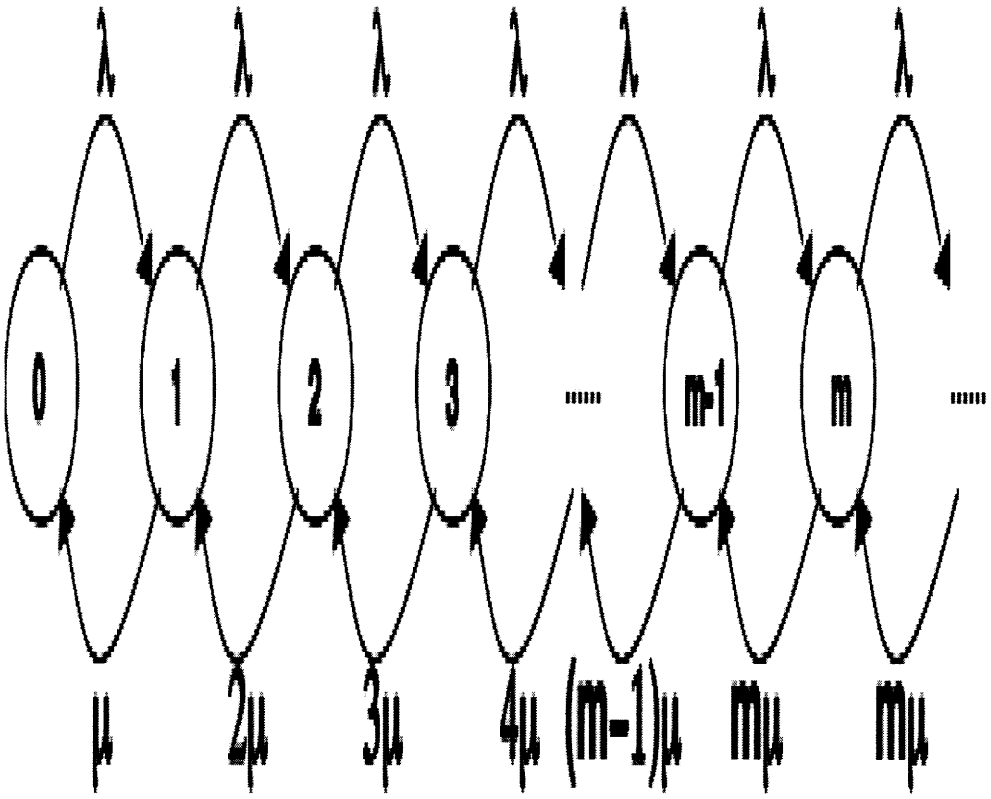
In this section we examine the properties of the M/M/m/K system. This queuing system contains m servers and K total number of spaces for customers including servers and buffer spaces, where $K \geq m$. Therefore the number of buffer spaces in the system is $K - m$. This system is related to the scheduled OFS system in that the variable K determines the the number of holders in a way analogous to the holder variable M. Our goal is to show that the M/M/m/K system obeys certain monotonicity properties similar to those that we wish to prove for the scheduled OFS system.

A.1.1 Background

In this section, we prove that for the M/M/m/K queue where $K > m$, the following two theorems hold for steady state.

Monotonicity of P_b with increasing K. *The probability of blocking of customers arriving at an arbitrary time (P_b) decreases as K increases and all other system parameters are held constant*

Monotonicity of U with increasing K. *The average utilization of servers U, defined to be the expected number of busy servers in steady state, increases as K increases and all other system parameters are held constant*



(a) Discretized Markov Chain for M/M/m queue

Figure A-1: Infinite Markov Chain

In order to prove these two theorems we require some results from queuing theory about the $M/M/m$ queue (K=∞). [7] provides analysis of the Markov chain shown in Figure A-1. This chain represents the a discretization of the Continuous Time chain induced by the M/M/m queue. The following results from the development of the $M/M/m$ queue are needed for our proofs. We have taken them from the development in [7].

$$p_n = p_0 \frac{(m\rho)^n}{n!} \qquad n \leq m$$

$$p_n = p_0 \frac{m^m \rho^n}{m!} \qquad n > m$$

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

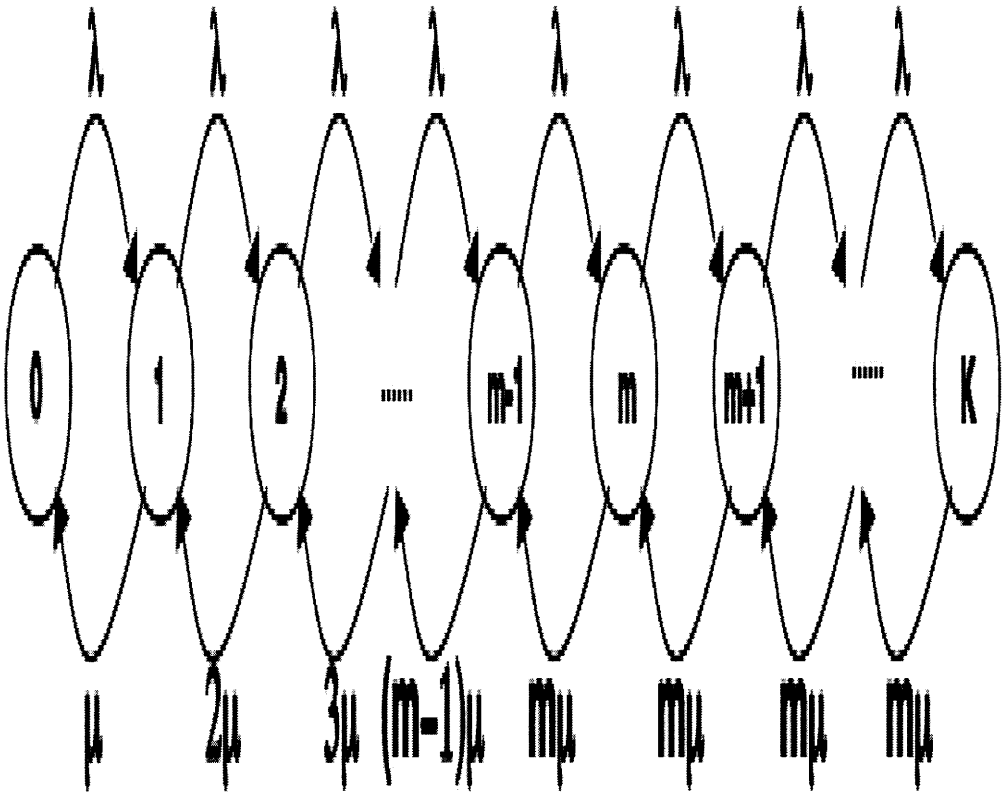
In these results, n is the occupancy of the queue p_n is the probability of an occupancy of n, and m is the number of servers. λ, μ are the average interarrival and service times of the system, respectively. ρ is defined to be the familiar load:

$$\rho = \frac{\lambda}{m\mu} < 1$$

A.1.2 Proofs

The Markov model for the M/M/m/K is related to the M/M/m birth-death chain by a *truncation*. As shown in Figure A-2, the truncation involves making the arrival transmission of the Kth state a self-transition and eliminating all states greater than K. This is justified by the fact that the arrival of a customer in the Kth state does not change the state, and states with an occupancy greater than K are not possible.

We have shown by earlier analysis in Subsection 6.1.3 using the work of Kelly [4] that such a truncation results in the remaining states retaining the same steady-state probabilities, with the sum normalized to unity. Occupancy probabilities of the M/M/m/K queue are therefore the same p_n defined above, each normalized by a



(a) Discretized Markov Chain for M/M/m/K queue

Figure A-2: Finite Markov Chain

factor G, defined as follows:

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

That is G is the sum of the remaining states' steady-state probabilities, which normalizes the sum of these probabilities to unity. Recall that K is the total number of spaces in the system, including buffers and servers. Given this model, we now prove the two theorems of interest, first for P_b:

Proof of Monotonicity of P_b with increasing K. From above, we know:

$$P_b = P_K = p_0\frac{\frac{m^m\rho^K}{m!}}{G}$$

There are two cases:

Case 1: $\rho \leq 1$ The numerator of P_K can be viewed as as a constant (in K) multiplied by ρ^K. The latter clearly decreases with increasing K therefore the overall numerator decreases with K. To prove the theorem we need to show that G increases with K, decreasing the overall blocking probability. From above G is:

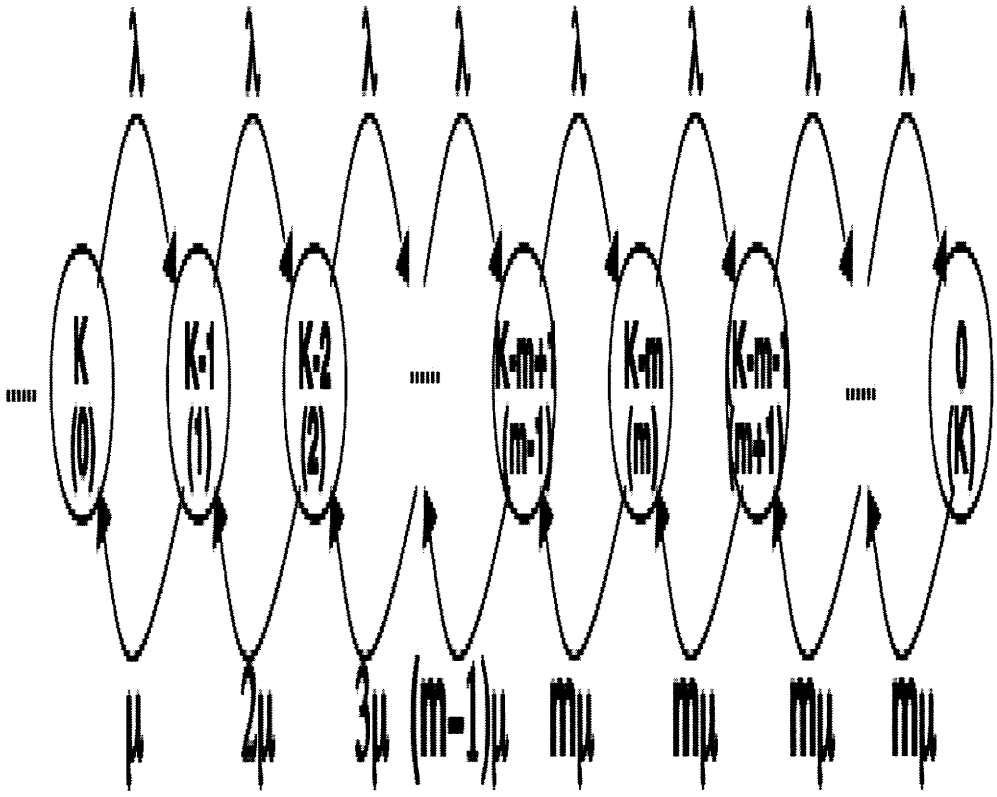
$$G = \sum_{n=1}^{m-1}[p_0\frac{(m\rho)^n}{n!}] + \sum_{n=m}^{K}[p_0\frac{m^m\rho^n}{m!}]$$

The first sum is invariant with K, the second sum can be rearranged to form:

$$\frac{m^m}{m!}\sum_{n=m}^{K}\rho^n$$

Which clearly increases with increasing K, since $\rho > 0$ except in a degenerate case, where $\lambda =$ and/or $\mu = 0$.

Case 2: $\rho > 1$ In order to prove monotonicity of P_b with $\rho > 1$, we cannot use the expression for P_b listed above, because the original Markov chain (pre-truncation) is unstable with $\rho > 1$.



(a) Discretized Markov Chain M/M/m queue stable for $\rho > 1$

Figure A-3: Infinite Markov Chain for $\rho > 1$

We therefore use the Markov chain pictured in A-3, which overlaps with the original chain in K states as shown. This chain extends to the left, and is stable for $\rho > 1$. The proof will proceed by finding the steady-state distribution of this chain truncated at state K. The resulting chain will looks exactly like the finite chain in Figure A-2, except for a re-numbering of the states. The probability of blocking (P_b) in the finite chain (Figure A-2) will be the same as the probability of being in the zero state (P_0) in the new finite chain (Figure A-3).

The proof will proceed by proving the result that the probability of being in the zero state in steady state P_0 increases monotonically as $K \leftarrow K + 1$, for a generalized chain. The generalized chain will have arbitrary arrival rates λ_i that satisfy the property $\lambda_i \geq \lambda_{i+1}$. We will then show that the truncated chain in Figure A-3 satisfies this property trivially thus the result applies to it. In order to complete the proof, we present an induction argument using the above property.

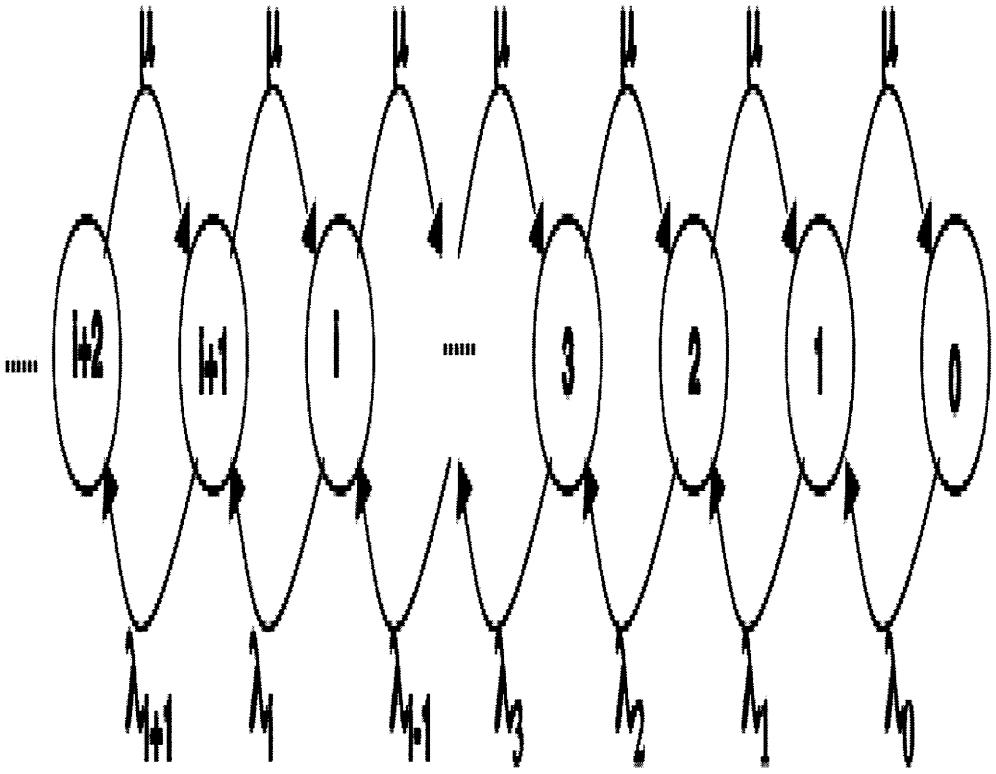
Figure A-4 shows a generalized version of the truncated form of the left-handed chain. The arrival rates λ_i are different for each state, while the service rates μ are fixed. We assume that $\lambda_i \geq \lambda_{i+1}$ for all states. The truncation of the chain dictates that the steady state probabilities of the a state m obeys the following proportionality:

$$P_m = \frac{\prod_{i=0}^{m-1} \frac{\lambda_i}{\mu}}{\sum_{l=0}^{K} \prod_{j=0}^{l-1} \frac{\lambda_j}{\mu}}$$

In particular, the expression for P_0 for a given K is:

$$P_0 = [\sum_{l=0}^{K} \prod_{j=0}^{l-1} \frac{\lambda_j}{\mu}]$$

Define $\lambda_i^{(K)} \quad 0 \leq i \leq K$ to be the arrival rate values for the chain of length K. If we define $\lambda_i^{(K+1)} \quad 0 \leq i \leq K+1$ to be the arrival rate values for the chain of length K+1. Then we can write the following relations between the two sets of



(a) Markov Chain with carious arrival rates, λ_i

Figure A-4: Markov Chain with arbitrary arrival rates

arrival rates as follows.

$$\lambda_0^{K+1} = \lambda_0^{(K)}, \lambda_1^{(K+1)} = \lambda_0^{(K)}, \lambda_2^{(K+1)} = \lambda_1^{(K)} ... \lambda_K^{(K+1)} = \lambda_{K-1}^{(K)}$$

Thus the expressions for P_0 for the two cases are (all rates are in terms of the K chain):

$$P_0^K = 1 + \frac{\lambda_0^{(K)}}{\mu} + \frac{\lambda_0^{(K)} \times \lambda_1^{(K)}}{\mu^2} + ... \frac{\prod_{l=0}^{K-1} \lambda_l^{(K)}}{\mu^K}$$

$$P_0^{K+1} = 1 + \frac{\lambda_0^{(K)}}{\mu} + \frac{\lambda_0^{(K)} times \lambda_0^{(K)}}{\mu^2} + \frac{\lambda_0^{(K)} \times \lambda_0^{(K)} \times \lambda_1^{(K)}}{\mu^3} + ... \lambda_0^{(K)} \times \frac{\prod_{l=0}^{K-2} \lambda_l^{(K)}}{\mu^K} ... \lambda_0^{(K)} \times \frac{\prod_{l=0}^{K-1} \lambda_l^{(K)}}{\mu^{K+1}}$$

Comparing the expressions term by term, we see that by the fact that $\lambda_i \geq \lambda_{i+1}$ for all i, the second sum is clearly greater than the first. This proves that P_0 is monotonically increasing with K, for this generalized chain. To prove the desired result, we observe that the arrival rates of our truncated original chain obey $\lambda_i \geq \lambda_{i+1}$. An inductive statement argument completes the proof:

(i) **Base case**: Using above argument show case for going from K=m+1 to K=m+2 for arbitrary positive m.

(ii) **Assumed case**: Assume theorem is true going from some K-1 to K.

(iii) **Inductive Case**: Use above argument to show the theorem going from K to K+1.

□

It is interesting to examine the limit of P_b as $K \rightarrow \infty$ for both cases involving ρ. For this, we use the expression:

$$P_b = \frac{\rho^K}{C + \sum_{n=m}^{K} \rho^n}$$

First, for the case when $\rho \leq 1$, we see that the numerator approaches zero as K increases. The denominator clearly is greater than zero, so the overall limit must be zero, as expected.

For the case where $\rho > 1$, we rearrange further, using the identity $\sum_{n=0}^{K} \rho^n = \frac{\rho^{K+1}-1}{\rho-1}$:

$$P_b = \frac{\rho^K}{C + \frac{\rho^{K+1}-1}{\rho-1} - \frac{\rho^m}{\rho-1}}$$

Dividing numerator and denominator by ρ^K yields:

$$P_b = \frac{1}{\frac{C}{\rho^K} + \frac{\rho-\rho^{(m-K)}}{\rho-1}}$$

Examining the denominator as $K \rightarrow \infty$, the terms $\frac{C}{\rho^K}$ and $\rho^{(m-K)}$ go to zero in the limit. This leaves a limit of $\frac{\rho-1}{\rho}$. This limit is sound from a probabilistic analysis standpoint in the sense that it is a valid probability that goes to 1 as $\rho \rightarrow \infty$. However, the definition of the model suggests that when $K = \infty$ we expect the blocking probability to be zero, not some non-zero value. This apparent contradiction is likely due to the instability of the M/M/m/∞ queue with $\rho \geq 1$.

Proof of Monotonicity of U with increasing K. We proceed by showing that the average utilization of the M/M/m/K+1 queue is larger than or equal to that of the M/M/m/K queue. This admits an inductive proof with a trivial base case. For the proof we make the following definitions:

$$U_K = 0 \times \frac{p_0}{G_K} + 1 \times \frac{p_1}{G_K} ....m \times \frac{p_m}{G_K} ....m \times \frac{p_K}{G_K}$$

Thus the utilization is the expected number of busy servers averaged over all states in steady-state. For the purposes of the proof, we define:

$$p_n^K = p_n / G_K$$

$$p_n^{K+1} = p_n / G_{K+1}$$

These definitions allow us to define:

$$U_K = \sum_{n=0}^{m-1} n p_n^K + m \sum_{n=m}^{K} p_n^K$$

$$U_{K+1} = \sum_{n=0}^{m-1} n p_n^{K+1} + m \sum_{n=m}^{K+1} p_n^{K+1}$$

We can also write:

$$p_n^{K+1} = p_n^K - \epsilon_n \qquad 0 \leq n \leq K$$

Where ϵ_n is a small probability. This is because we have earlier proved that G is monotonically increasing with K, and G is in the denominator of both p_n^K and p_n^{K+1}.

We also know that, since both are valid steady-state distributions:

$$\sum_{n=0}^{K} p_n^K = 1$$

$$\sum_{n=0}^{K+1} p_n^{K+1} = 1$$

We can write the latter sum as:

$$\sum_{n=0}^{K} [p_n^K - \epsilon_n] + p_{K+1}^{K+1}$$

The latter three facts show that:

$$p_{K+1}^{K+1} = \sum_{n=0}^{K} \epsilon_n$$

We may now re-write the average utilization formulas:

$$U_K = 0 \times (p_0^{K+1} + \epsilon_0) + 1 \times (p_1^{K+1} + \epsilon_1) ... + m \times (p_K^{K+1} + \epsilon_K)$$

$$U_{K+1} = 0 \times (p_0^{K+1}) + 1 \times (p_1^{K+1}) ... + m \times (p_K^{K+1}) + m \times (p_{K+1}^{K+1})$$

Rewriting the last term as above:

$$U_{K+1} = 0 \times (p_0^{K+1}) + 1 \times (p_1^{K+1}) ... + m \times (p_K^{K+1}) + m \times \sum_{n=0}^{K} \epsilon_n$$

We can write U_K in a similar form:

$$U_K = 0 \times (p_0^{K+1}) + 1 \times (p_1^{K+1}) ... + m \times (p_K^{K+1}) + \sum_{n=0}^{m-1} n\epsilon_n + m \times \sum_{n=m}^{K} \epsilon_n$$

We can see that all terms except the summation terms in the U_{K+1} and U_K formulas are the same. We can also see that the last term (sum term) of U_{K+1} is always greater than or equal to that of U_K for $m \geq 1$. This proves the result for an arbitrary step from K to K+1.

To complete the proof we state an inductive form with each step using the arguments above.

(i) **Base case**: Using above argument show case for going from K=m+1 to K=m+2 for arbitrary positive m.

(ii) **Assumed case**: Assume theorem is true going from some K-1 to K.

(iii) **Inductive Case**: Use above argument to show the theorem going from K to K+1.

□

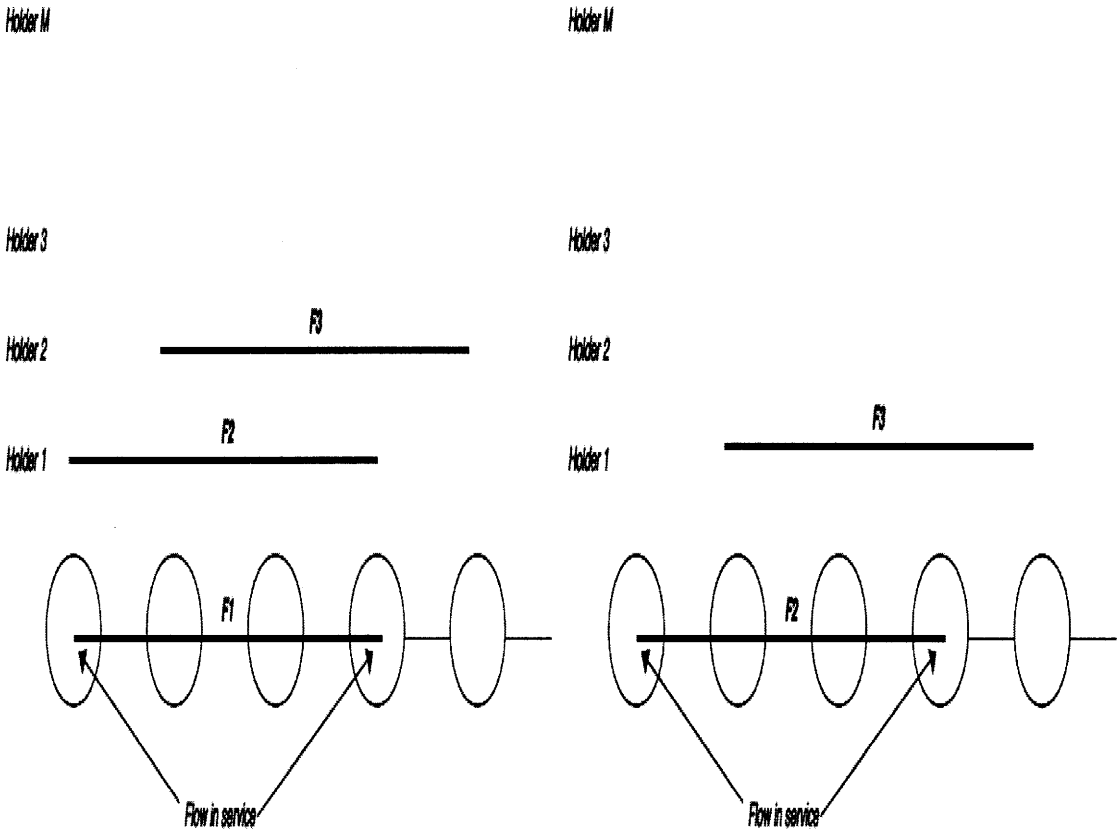
A.1.3 Discussion

The two theorems proved here show that an increased buffer for a finite multi-server queue makes for better blocking and utilization behavior. Another fact follows from these proofs: For a given blocking probability, a queue with more buffer will always have better utilization performance. We hope to apply similar reasoning to the case where there are inter-dependences between servers with respect to which customers they serve, such as in a scheduled OFS network.

A.2 Modified Scheduled OFS System

In this section, we present closed form analysis of a system that is closely related to the scheduled OFS system, called the Modified Scheduled OFS System or modified system for short. This system is similar to the original system, except that it is work-conserving. We prove two monotonicity theorems about the modified system, relying on the results of the previous section.

A.2.1 Background



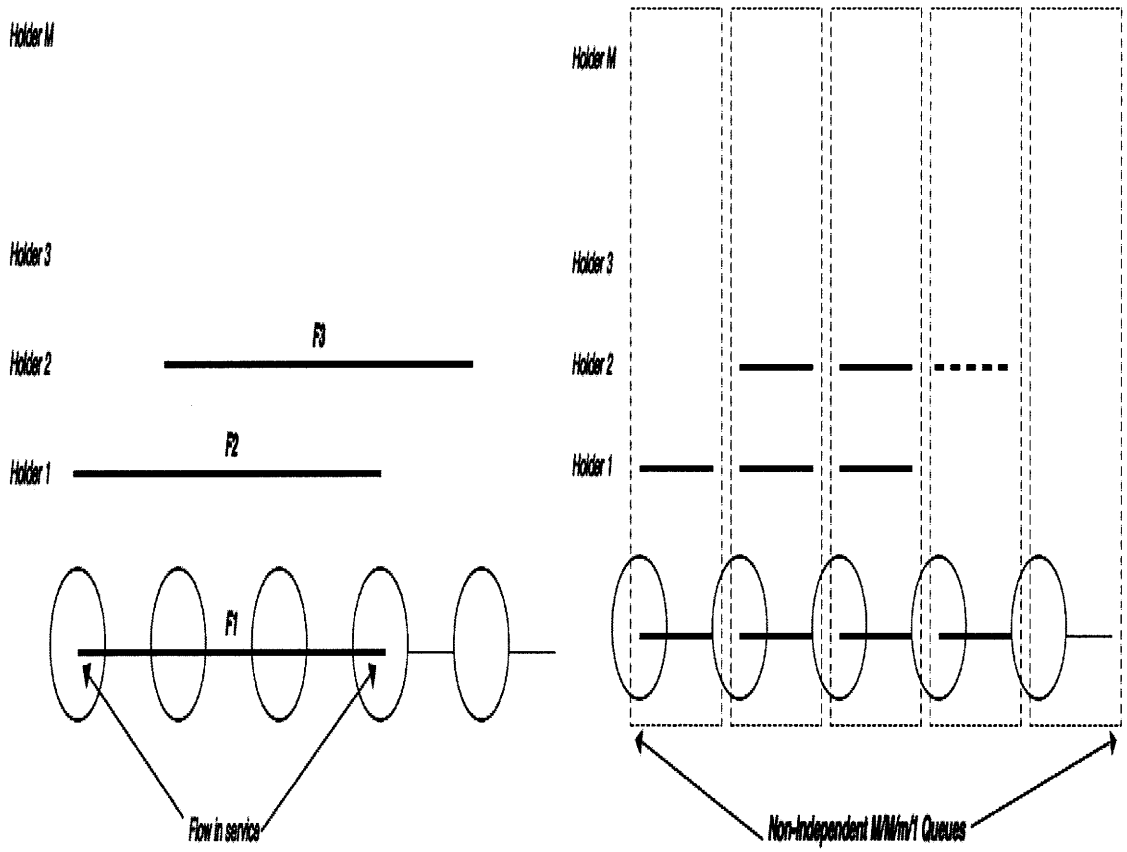
(a) Scheduling Scheme Diagram

(b) Post-departure Situation

Figure A-5: Original Scheduled OFS System

Figure A-5(a) demonstrates the idea of scheduling in an OFS system. It is identical to the scheduled OFS system discussed in Chapter 5. In the figure, there is a single

optical channel between the nodes and M scheduling holders. When requests for connections arrive, if the resources are available in the channel, they are granted. Otherwise, the request is placed in the lowest numbered holder possible that has space, and if no such holder exists, the request is dropped.



(a) Original System (b) Modified System

Figure A-6: Comparison of Original and Modified System

Figure A-6 shows a modified version of the OFS system. The key difference in the modified system is that arrivals of length n>1 are treated as n separate length-1 arrivals. That is a request for a connection between nodes i and j where i<j is treated as j-i length-1 arrivals, one to each link connecting nodes i and j. This modification results in the removal of coscheduling constraints for a single arrival. That is, each one link segment that comprises the arrival is served individually, independent of others. This makes the system *work conserving* since if there is any flow that requires

a particular link in the system, the link cannot remain unoccupied.

In this report, we prove the following theorems hold for the modified system with a scheduling horizon of M holders:

Monotonicity of P_b with increasing K. *The Union Bound on the probability of blocking of flows arriving at an arbitrary time (P_b) decreases monotonically as M increases and all other system parameters are held constant*

Monotonicity of U with increasing K. *The average utilization of the optical links U, defined to be the expected number of busy optical links in steady state, increases monotonically as M increases and all other system parameters are held constant*

We define blocking for the modified system below.

A.2.2 Model

As shown in the Figure A-6(b), arrivals will still have hop lengths but will enter the system as a set of independent length-1 customers. Therefore, the steady-state probability can be calculated by treating each link as a M/M/1/M queue, with arrival rates adjusted as per customer hop lengths. The situation then reduces to a N M/M/1/M queues with various arrival rates, which we quantify later in this section. Note that the arrival processes at the queues are still Poisson, but the queues are not necessarily independent because of the bulk arrivals induced by the length of arrivals being generally greater than one. For example, if a length 3 arrival occurs at node 1, then this tells us that an arrival is guaranteed to occur at nodes 2 and 3 at that time. This shows that the arrival processes are not independent.

The dependence of the arrival processes will not affect the proof of the utilization result as we will show. For the blocking probability, we will show that the Union Bound on the blocking probability is decreasing. This result will suffice for our purposes.

A.2.3 Proof and Transformation from Original to Modified System

The transformation from the original system to the modified system involves a calculation of arrival rate for each link based on the original arrival rate of customers to each node, and the length PMF. For this discussion, we assume that the latter is uniform as a function of node i, and is pre-set. Note that the calculation is valid for any parameterized PMF for lengths.

Recall that arrival rates for the OFS line systems are defined by a matrix of individual (node, length) arrival rates Λ where each value $\Lambda(i,j)$ provides the arrival rate to node i of length j. With this definition, the N queues in the modified OFS system have arrival rates that are the summation of a column of the matrix. That is for the queue corresponding to node i we have

$$\lambda_i = \sum_{k=1}^{N} \Lambda(k,i)$$

These rates are valid for any system OFS system that is transformed to a modified OFS system. This transformation of rates, along with the previous illustration in Figure A-6(a)(b) completes the transformation to the modified system. The links behave as M/M/1/M queues with the arrival rates listed above. The proof of the utilization result follows directly from the development for the M/M/m/K queue.

Proof of Monotonicity of U with increasing M. To begin, note that the nodes in Figure A-6(b) are numbered from 1 to N. The average utilization of the system can be viewed as the expected number of busy nodes in steady state. Define $\Phi(i,S)$ to be an indicator variable that is 1 iff node i is busy in a particular state S. The contribution of state U_S to the overall utilization is then:

$$U_S = \sum_{i=1}^{N} \Phi(i,S) \times \pi(S)$$

Where $\pi(S)$ is the steady state probability of state S. U is:

$$U = \sum_{\text{all S}} U_S$$

The sum U_S is a sum of expectation of the utilization of each individual queue. Since expectation is linear, this sum is the same whether the queues are independent or not. Hence, the sum of the expectations is equivalent to the expectation of the sum.

Each queue is M/M/1/K and earlier M/M/m/K results show that each queues' expected utilization increases with K individually. Therefore, both the components of $\sum_{\text{all S}} U_S$ and, in turn, U must increase monotonically with K, completing the proof. □

Proof of Monotonicity of the Union Bound of P_b with increasing M. Refer to Figure A-6(b). Recall that incident blocking probability is blocking of an arriving customer due to a full system. We assume any type of (node, length) arrival to be equally probable. For the modified system, blocking of an arrival of length L to a node i involves the L M/M/1/M queues (nodes) numbered i through i+L. We will say that if *any* of these queues are full at arrival time, then the arriving customer is blocked. Assume a length L customer is arriving at node i, and let *bi* be the event that the queue for node i is full. Then P_b for this (and all) arrivals is the union of L independent events as follows:

$$P_b = P(\bigcup_{k=i}^{i+L} bk)$$

In general this union is difficult to compute for the non-independent queues. We instead use the Union Bound, which states that for and collection of events e:

$$P(\bigcup_{\text{all e}} e) \leq \sum_{\text{all e}} P(e)$$

...with equality iff the events e are mutually disjoint. Using this we can write

$$P_b = P(\bigcup_{k=i}^{i+L} bk) \leq \sum_{k=i}^{i+L} P(bk)$$

We know from the M/M/m/K results that P(bi) for a node i decreases monotonically with increasing K. Therefore the Union Bound on P_b is monotonically decreasing, completing the proof.

□

A.2.4 Discussion

Note that we have only shown an monotonically increasing *upper bound* on P_b for this system. The results of this analysis show that for the work-conserving modified OFS system, both blocking and utilization are are improved by increasing scheduling horizon (K). This is also a proof of a decreasing lower bound for P_b and a monotonically increasing upper bound for U for the original system. Numerical calculation can examine how tight these bounds are. This is also evidence that the theorems may hold for the original Scheduled OFS system, which we address next.

A.3 Monotonicity Theorems for Scheduled OFS System

In this section, we study the Scheduled OFS system model from a theoretical perspective (first presented in Chapter 5). As earlier stated, the study of this system is difficult, since the Markov chain induced is provably non-reversible and is therefore not amenable to previously seen analysis on queuing type Markov systems. In this section, we present two monotonicity theorems, and short of proving them, we present strong numerical in the form of an example. We also reduce the proof of the two theorems to an intuitively plausible result, although we have not proven this result as of this writing.

We have seen that numerical results in Figure 6-14 strongly suggest that these theorems hold. However, the super-exponential growth of the state space of the

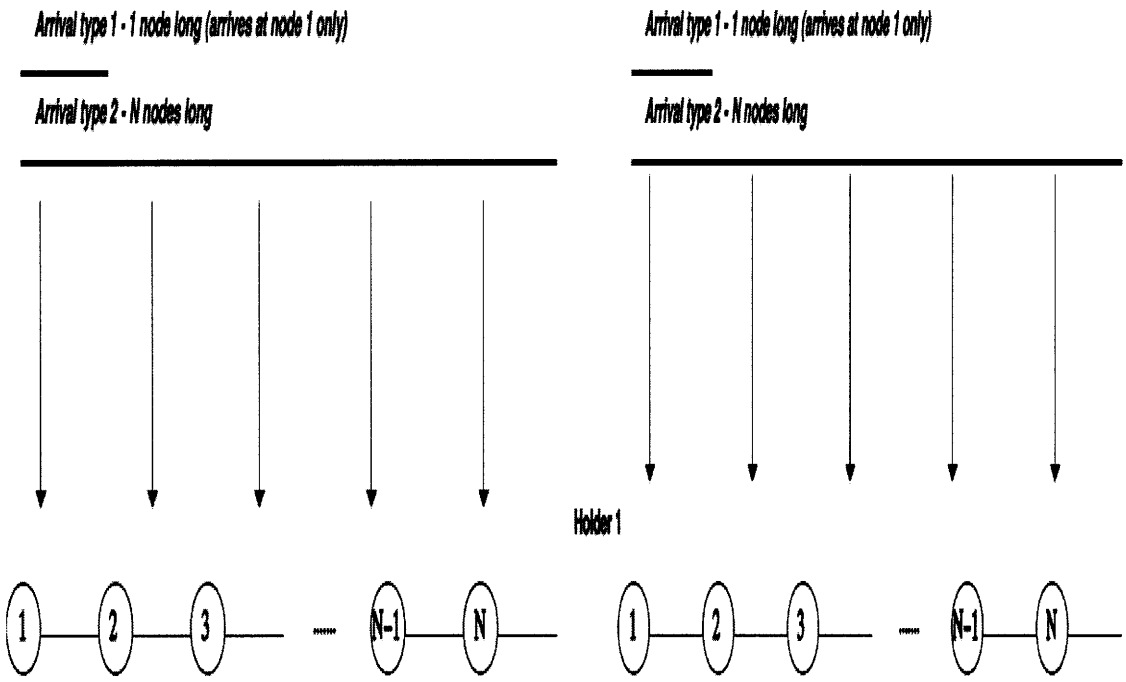
system makes numerical solution difficult for large size cases. Since proof is of the theorems is not obvious, we first pursue a counter-example to the above theorems. We choose a simple example which possesses properties that may cause it to violate the theorems. As it turns out, even this "counter"-example obeys the theorems.

A.3.1 Example

Figure A-7(a) shows the two systems (A,B) of interest. They are identical except that System B has a single scheduling holder (M=1) while A is an unscheduled system. There are only two types of customers arriving to the systems. Arrivals of type 1 are length 1 and arrive only at node 1, with an arrival rate λ_1 and an average service rate μ_1. Arrivals of type 2 are length N and have an arrival rate λ_2 and average service rate of μ_2. Our methodology is to vary the arrival rates and service rates of the two systems to see if we can arrive at some set of parameters where the systems violate the theorems. In order to build intuition, consider the sample path of arrivals illustrated by Figure A-7(b). The figure illustrates a sample path of arrivals of types 1 and 2 that can be applied to both systems. In particular, it shows that at some time after the origin, an arrival of type 2 arrives to the system, followed by a long duration arrival of type 1. After this, a series of type two arrivals occurs, completing the sample path. We can track the behavior of the two systems as the sample path is applied to them:

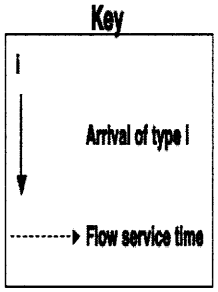
System A The first type 2 arrival is accepted into the network. The type 1 arrival is dropped due to insufficient resources. All subsequent type 2 arrivals are accepted as they arrive.

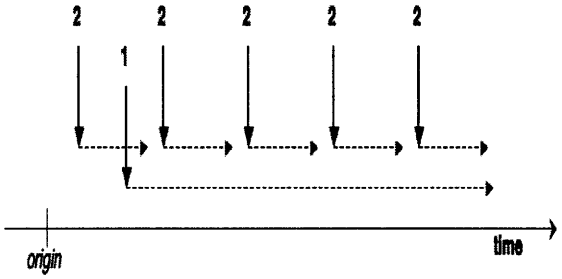
System B The first type 2 arrival is accepted into the network. The type 1 arrival is placed in Holder 1. Upon departure of the first type 2 customer, the type 1 customer moves from Holder 1 into the network. The subsequent type 2 arrival is placed in Holder 1 and all remaining arrivals are blocked, due to the type 1's long duration.



(a) System A for Example

(b) System B for Example





(c) Example Sample Path

Figure A-7: Counterexample Description

From these descriptions it is clear that for this particular sample path, system A has much better blocking and utilization performance than system B. Furthermore, the sample path appears to be not only plausible, but likely in the case $\lambda_2 >> \lambda_1$ and $\mu_2 >> \mu_1$. So this system appears to have a chance of violating the theorems if they do not hold in general.

We have solved for the utilization and the blocking probability in steady state in closed form, defined as follows:

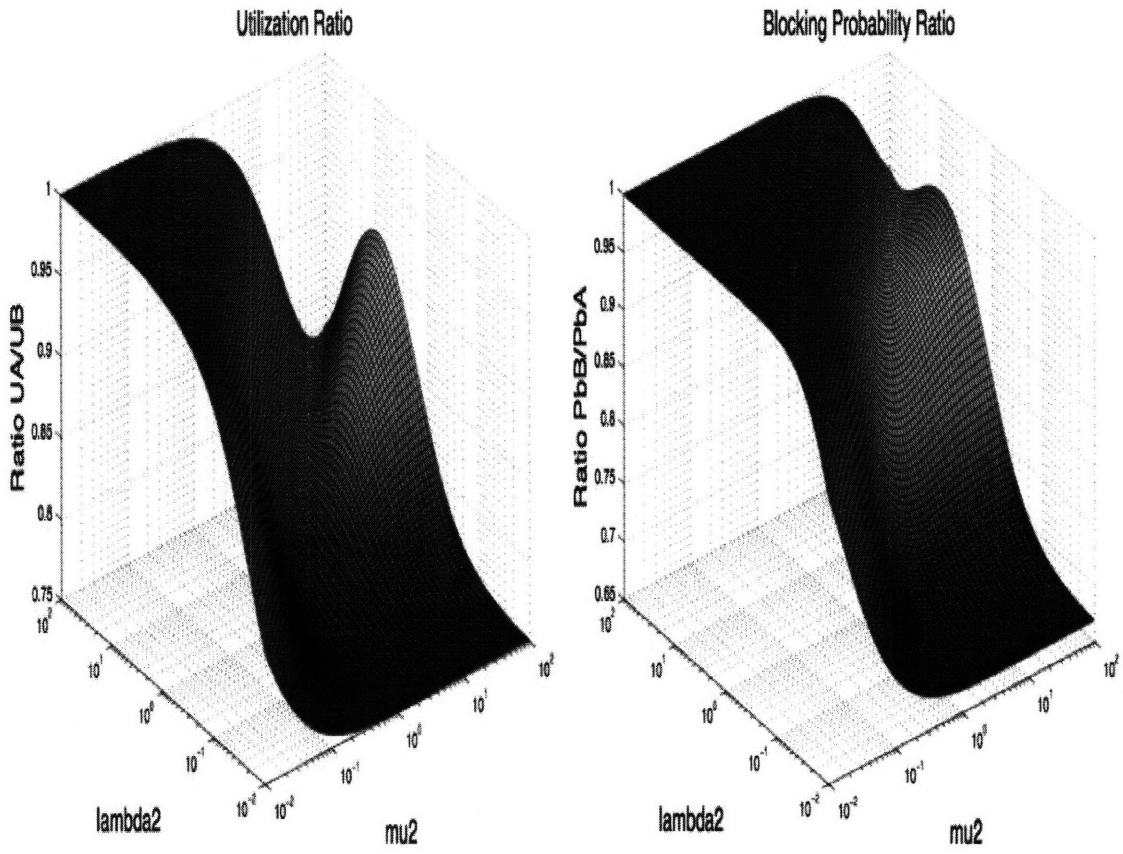
- U_A - The average utilization of the nodes in system A.
- U_B - The average utilization of the nodes in system B.
- P_{bA} - The average blocking probability of arrivals in system A.
- P_{bB} - The average blocking probability of arrivals in system B.

From these we can define the Utilization Ratio to be $\frac{U_A}{U_B}$, and the Blocking Probability Ratio to be $\frac{P_{bB}}{P_{bA}}$. If either of these ratios exceeds one for any choice of arrival and duration parameters, then the theorem is broken by the systems in the example. The solutions are parameterized by the arrival and service rates 1 and 2, but we have normalized λ_1 and μ_1 to be 1 since it is the relative rates that we wish to vary.

Figure A-8(a)(b) show the results. The figures show surface plots of the ratios $\frac{U_A}{U_B}$ and $\frac{P_{bB}}{P_{bA}}$ respectively versus the λ_2, μ_2 plane. There are several interesting aspects to the plots. First, there are local maxima and minima in both ratio plots. These are likely artifacts of the arrival processes we are using, since they are highly irregular. Second, neither of the plots exceed 1 so this is not a counter-example to the theorems even for large ratios of arrival rates and service rates. Even when the rate of arrival of arrivals of type 2 are very high, similar to our counter-example sample path in Figure A-7(c), the ratios do not exceed 1.

A.3.2 Discussion

OFS faces the potential problem of having low utilization of optical resources with poor customer blocking probability. We have presented numerical results that suggest



(a) Utilization Ratio for Example

(b) Blocking Probability Ratio for Example

Figure A-8: Example Results

that scheduling is a good solution to improve OFS performance. These results also suggest two theorems that appear to hold for the scheduled OFS system. These theorems were shown to hold for several related systems, and for an extensively studied example which intuitively appeared to be capable of violating the theorems but did not. All of these factors lead us to believe that the theorems do hold, but proofs have not been found as of this writing.

Two directions seem promising in pursuing the aforementioned proofs. First, extensive study of the example presented in this chapter may reveal properties or suggest other examples that may be helpful. Second, detailed analysis of the structure of the Markov chains induced by scheduled OFS need to be done in order to find out what properties are changing as the scheduling horizon is increased. If these are identified and characterized, a proof can potentially follow.

Bibliography

[1] Mark E Crovella and Murad S. Taqqu and Azer Bestavros, *Heavy-tailed probability distributions in the World Wide Web*, (A Practical Guide to Heavy Tails:Statistical Techniques and Applications), Birkhauser, Boston(1998), pp 3-25

[2] Xiaoyun Zhu and Jie Yu and John Doyle *Heavy Tails, Generalized Coding, and Optimal Web Layout*, (Proceedings of IEEE INFOCOM 2001), pp 1617-1626

[3] Bishwaroop Ganguly and Vincent Chan *A Scheduled Approach to Optical Flow Switching in the ONRAMP Optical Access Network Testbed*, Optical Fiber Communications 2002

[4] F.P. Kelly, *Blocking Probabilities in large circuit-switched networks*, Advances in Applied Probability 19, 1986, pp 474-505

[5] William J. Stewart, *Introduction to the Numerical Solution of Markov Chains*, Princeton University Press, 1994

[6] J.R. Norris *Markov Chains*, Cambridge University Press, 1997

[7] Dimitri Bertsekas and Robert Gallager, *Data Networks, second edition*, 1992, Prentice-Hall

[8] Robert Gallager, *Discrete Stochastic Processes*, 1996, Kluwer Academic Publishers

[9] Greg Bernstein and Bala Rajagopalan, *Introduction to optical control plane standards and technology:OIF UNI, GMPLS, G.ASON and all that*, (Optical Fiber Communication Short Course Notes), March 18, 2002

[10] Brett Schein and Eytan Modiano *Quantifying the benefit of configurability in circuit-switched WDM ring networks*, Proceedings of IEEE INFOCOM, 2001, pages 1752-1760

[11] Yousong Mei and Chunming Qiao, *Efficient Distributed Control Protocols for WDM All-Optical Networks*, (Proceedings of Sixth International Conference on Computer Communications and Networks), 1997, pp 150-153

[12] Bishwaroop Ganguly and Eytan Modiano, *Distributed Algorithms and Architectures for Optical Flow Switching in WDM networks*, in (Proceedings of ISCC 2000), pp 134-139.

[13] A. Mokhtar and M Azizoglu *Adaptive wavelength routing in in all-optical networks* In Sixth International Conference on Computer Communications and Networks, pages 150-153, 1997

[14] S. Subramaniam and R.A. Barry *Wavelength assignment in fixed routing WDM networks* In 1997 IEEE International Conference on Communications, volume 1, pages 406-410, 1997

[15] Vincent W.S. Chan et al., *Architectures and Technologies for High-Speed Optical Data Networks*, (Journal of Lightwave Technology), Vol. 15, No. 12, pp 2146-2168.

[16] D.L. Mills, *On the accuracy and stability of clocks synchronized by the Network Time Protocol in the Internet system*, (ACM Computer Communication Review), Vol. 20, No. 1, pp 65-75.

[17] G. Papadimitriou and Chrisoula Papazoglu and Andreas S. Pomportsis *Optical Switching: Switch Fabrics, Techniques and Architectures* Journal of Lightwave Technology, volume 21, No 2, pages 384-405, 2003

[18] R. Ramaswamy and K. Sivarajan *Optical Networks: A Practical Perspective* Morgan-Kaufmann Publishers, 2002

[19] T.E. Stern and K Bala *Multiwavelength Optical Networks*, Addison-Wesley, 1999

[20] B. Mukherjee *Optical Communication Networks*, McGraw-Hill, 1997

[21] R.A. Barry and P.A. Humblet *Models of Blocking Probability in All-Optical Networks with and Without Wavelength Changers* IEEE Journal on Selected Areas in Communications, vol. 14, no. 5, June 1996, pp. 858 - 867

[22] P. Bayvel *Wavelength Routing and Optical Burst Switching in the Design of Future Optical Network Architectures*, Proceedings 27th European Conference on Optical Networking, pages 616-619

[23] A. Zapata P. Bayvel *Dynamic Wavelength-Routed Optical Burst Switched Networks: Scalability Analysis an Comparsion of with Static Wavelength-Routed Optical Networks*, Optical Fiber Communications, Volume 1, 2003

[24] J. Li and C. Qiao and J. Xu and D. Xu *Maximizing Throughput for Optical Burst Switching Networks*, Proceedings of IEEE Infocom, 2004

[25] Z. Rosberg and H.L. Vu and M. Zukerman and J. White *Performance Analyses of Optical Burst-Switching Networks*, IEEE Journal on Selected Areas in Communications, Vol. 21, No. 7, September 2003, pages 1187-1197

[26] C.M. Gauger *Dimensioning of FDL Buffers for Optical Burst Switching Nodes*, Proceedings of the 6th IFIP Working Conference on Optical Network Design and Modeling, February 2002

[27] S Shen et al. *Benefits of advertising wavelength availability in distributed lightpath establishment*, Computer Networks, Volume 50, Issue 13, September, 2006

[28] J. Li et al. *Dynamic routing with inaccurate link state information in integrated IP-over-WDM networks*, Computer Networks, Volume 46, Issue6, December, 2004

[29] J.N. Turner *Terabit Burst Switching*, Journal of High Speed Networks, 1999

[30] A. Zalesky and H.L. Vu and Z. Rosberg and E.W.M. Wong and M. Zukerman *Modelling and Performance Evaluation of Optical Burst Switched Networks with*

Deflection Routing and Wavelength Reservation, Proceedings of IEEE Infocom, 2004

[31] J. Teng and G.N. Rouskas *Wavelength Selection in OBS Networks Using Traffic Engineering and Priority-Based Concepts*, IEEE Journal on Selected Areas in Communications, Vol. 23, N0. 8, August 2005, pages 1658-1669

[32] S. Lee and K Sriram and H. Kim and J. Song *Contention-Based Limited Deflection Routing Protocol in Optical Burst-Switched Networks*, IEEE Journal on Selected Areas in Communications, Vol. 23, No. 8, August 2005, pages 1596-1611

[33] H. Yang and S.J.B. Yoo *All-Optical Variable Buffering Strategies and Switch Fabric Architectures for Future All-Optical Data Routers*, Journal of Lightwave Technology, Vol. 23, No. 10, October 2005, pages 3321-3330

[34] A Rostami and S.S. Chakraborty *On Performance of Optical Buffers With Specific Numers of Circulations*, IEEE Photonics Technology Letters, Vol. 17, No. 7, July 2005, pages 1570-1572

[35] D. E. Comer *Internetworking with TCP/IP Volume 1: Principles Protocols, and Architecture*, Fifth edition, Prentice Hall, 2006

[36] Javvin Technologies *Network Protocols Handbook*, Second edition, Javvin Technologies Inc., January 2006

[37] P. Loshin *TCP/IP Clearly Explained*, First edition, Morgan Kaufmann, December 2002

[38] Ravi Malhotra *IP Routing*, First edition, O'Reilly, January 2002

[39] D Marquis *BoSSNET: An all-optical long haul networking testbed*, Technology Digest Lasers and Electro-Optical Society Annual Meeting, Vol 1, 2000

[40] R Gallager *Discrete Stochastic Processes*, First Edition, Kluwer Academic Publishers, 1996

[41] G Davies *Designing and Developing Scalable IP Networks*, First Edition, Wiley, 2004

[42] X Sun et al *Performance Analysis of First-Fit Wavelength Assignment Algorithm in Optical Networks*, 7th International Conference on Telecommunications, June 11-13, 2003

[43] T Cormen C Leiserson R Rivest *Introduction to Algorithms*, Second Edition, MIT Press, 2001

[44] W.R. Gilks et al *Markov Chain Monte Carlo in Practice*, First Edition, CRC Press, December, 1995