A Novel Architecture and Medium Access Control Protocol for WDM Networks

Eytan Modiano, Richard Barry and Eric Swanson
MIT Lincoln Laboratory
244 Wood St.
Lexington, MA 02173

Abstract

We describe an architecture and Medium Access Control (MAC) protocol for WDM networks. Our system is based on a broadcast star architecture and uses a single transceiver per node. The system uses an unslotted access protocol and a centralized scheduler to efficiently provide bandwidth-on-demand in WDM networks. To overcome the effects of propagation delays the scheduler measures the delays between the terminals and the hub and takes that delay into account when scheduling transmissions. Simple scheduling algorithms, based on a look-ahead capability, are used to overcome the effects of head-of-line blocking. Lastly, our system is extended to Metropolitan Area Networks (MANs) with a layered architecture that uses synchronization between Local Area Network (LAN) hubs, while terminals remain unsynchronized.

Keywords: 060.4250 networks; 999.9999 wavelength division multiplexing; 999.9999 medium access control

I. Introduction

In recent years there has been a wave of research toward the development of WDM-based Local Area Networks (LANs) [1-10]. Most of the proposed protocols and architectures are based on a broadcast star network architecture. Some of the protocols are based on random access and consequently result in low throughput due to contention [3,4]. Other protocols that attempt to minimize contention, through the use of some form of reservations, require that the system be synchronized and slotted and many require multiple transceivers per node [5-8]. Despite the added complexity of these systems, most still fail to achieve high levels of utilization due to the use of inefficient scheduling schemes that often fails to deal with receiver contention, or ignore the effects of propagation delays. A comprehensive survey of WDM multi-access protocols and their properties is presented in [1,2].

The purpose of the system described in this paper is to achieve good throughput delay characteristics, while maintaining simple user terminals. Previous efforts to simplify user terminals involved protocols that use fixed tuned receivers or transmitters [9-10]. However, those protocols limit the number of users to the number of available wavelengths and are hence not scaleable. Also, protocols using only a single fixed tuned device are often limited to the use of a random access protocol, that results in low channel utilization.

The architecture and protocol described in this paper eliminate the need for slotting and synchronization, uses one tunable transceiver per user terminal, yet results ın high utilization in both the LAN and the MAN. In the LAN the system is a simple broadcast-and-select Star network. Each user terminal consists of a single transmitter and receiver, both of which are tunable over all data wavelengths and one control wavelength. The proposed system consists of 32 wavelengths operating at 10 Gbps each. The system is extended to the MAN with a layered architecture, where LAN hubs are interconnected through a MAN hub. The MAN hub can be as simple as another broadcast star, a wavelength router, or a fully configurable frequency selective optical switch.

Our system is novel in a number of ways. First, it uses an unslotted MAC protocol, yet results in high efficiency even in high latency environments. The choice of an unslotted protocol is driven by a desire, for simplicity, to eliminate the requirement to maintain slotting in the network. Unfortunately, unslotted MAC protocols such as CSMA result in very low utilization in high latency. Alternatively, high latency protocols such as unslotted Aloha are limited in throughput to less than 18% [3,5]. Another novelty of our system is that it uses a centralized master/slave scheduler which is able to schedule

OSA TOPS Vol. 20 Optical Networks and Their Applications
Richard A. Barry (ed.)

transmissions efficiently. To overcome the effects of propagation delays the scheduler measures the delays between the terminals and the hub and takes that delay into account when scheduling transmissions. Lastly, our system is extended to MANs with a layered architecture that uses synchronization between LAN hubs, while terminals remain unsynchronized.

II. LAN Architecture

In the LAN, optical terminals (OTs) are connected via a simple broadcast star located at a hub. As shown in Figure 1, each OT is connected to the star using two fibers, one in each direction. Transmissions from all OTs on all wavelengths are combined at the star and broadcast to the OTs on the downlink fibers. Each OT is equipped with a single transmitter and receiver, both of which are tunable to all wavelengths, as shown in figure 2. All OTs send their requests to the scheduler on a dedicated control wavelength, λ_C, using a random access protocol. The scheduler, located at the star, schedules the requests and informs the OTs on a separate wavelength, $\lambda_{C'}$, of their turn to transmit. Upon receiving their assignments, OTs immediately tune to their assigned wavelength and transmit. Hence OTs do not need to maintain any synchronization or timing information. By measuring the amount of time that OTs take to respond to the assignments, the scheduler is able to obtain an estimate of each OT's round-trip delay to the hub. This delay information is then used by the scheduler to overcome the effects of propagation delays.

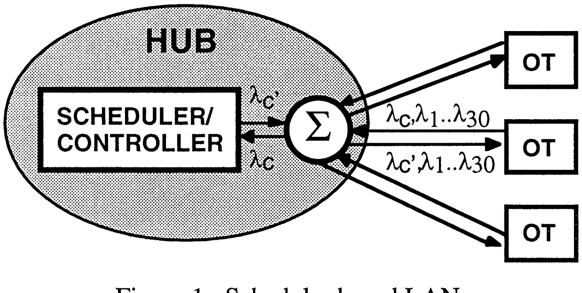
HUB
SCHEDULER/ CONTROLLER
$\lambda_{c'}$
λ_c
Σ
$\lambda_c,\lambda_1..\lambda_{30}$
$\lambda_{c'},\lambda_1..\lambda_{30}$
OT
OT
OT

Figure 1. Scheduler based LAN.

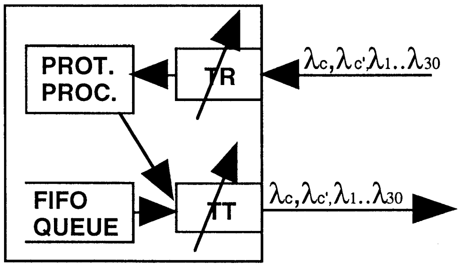
PROT. PROC.
TR
$\lambda_c,\lambda_{c'}\lambda_1..\lambda_{30}$
FIFO QUEUE
TT
$\lambda_c,\lambda_{c'},\lambda_1..\lambda_{30}$

Figure 2. Optical terminal (OT).

III. Access protocol

Our proposed protocol is based on a simple master/slave scheduler as was shown in figure 1. All OTs send their requests to the scheduler, which schedules the requests and informs the OTs when and on which wavelength to transmit. Upon receiving their assignments, OTs immediately tune to that wavelength and transmit. Hence OTs do not need to maintain any synchronization or timing information. There are three major aspects to the protocol. First, the protocol uses ranging to overcome the effects of propagation delays. Second, the protocol uses random access for the control channel and third, the protocol uses a simple scheduling algorithm with First-come-first-serve (FCFS) input queues and a look-ahead window to overcome Head-of-line (HOL) blocking. These are described in more detail below.

A. The use of ranging

The protocol is able to overcome the effects of propagation delays by measuring the round-trip delay of each OT to the hub and using that information to inform the OTs of their turn to transmit in a timely manner. For example consider figure 3, in order for OT B's transmission to arrive at the hub at time T, the scheduler must send the assignment to OT B at time T-τ, where τ is OT B's round-trip delay to the hub (including tuning delays). In this way the transmissions of different terminals can be scheduled back-to-back, with little dead-time between transmissions.

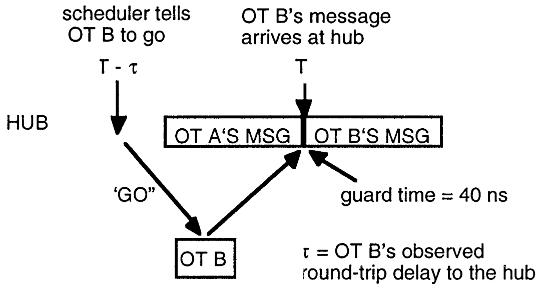
scheduler tells OT B to go
OT B's message arrives at hub
T - τ
T
HUB
OT A'S MSG
OT B'S MSG
'GO"
guard time = 40 ns
OT B
τ = OT B's observed round-trip delay to the hub

Figure 3. Use of ranging to overcome propagation delays.

An important and novel aspect of this system is the way in which ranging is accomplished. Unlike other systems where terminals need to range themselves to their hubs in order to maintain synchronization [11], here we recognize that it is only the hub that needs to know this range information. Hence ranging can be accomplished in a straightforward manner. The scheduler, ranges each terminal by sending a control message telling the terminal to tune to a particular wavelength and transmit. By measuring the time that it takes the terminal to respond to the request the scheduler can obtain an estimate of the round trip delay for that terminal. This estimate will also include the tuning time delays. Furthermore the scheduler can repeatedly update this estimate to compensate for fiber inaccuracies. These measurements can also be made by simply monitoring the terminals response to ordinary scheduling messages. The significance of this approach is that terminals are not required to implement a ranging function, which simplifies the OTs.

B. Access to the control channel

Reservations are made using a random access protocol to access the control channel where terminals send reservation requests periodically and update their requests after waiting a random delay. These reservation messages contain the state of the queues at the requesting terminal. For example, each reservation message can contain the destinations with which the terminal wants to communicate and the duration of the requested transmissions.[1]

Reservation requests are sent on the control channel at random, therefore it is possible for two or more terminals to send their request during overlapping time intervals. In which case their transmissions would "collide" and not be received by the scheduler. However, since reservation messages containing the state of the queue are sent periodically, all requests will eventually be received by the scheduler. As requests are answered by the scheduler, terminals update their requests to reflect the changes in their request queue.

In order to randomize transmissions on the reservation channel terminals wait a random exponentially distributed time, with an average duration $\overline{T}$, between successive transmissions of a reservation request[2]. With N terminals and an average rate of one request message every $\overline{T}$ seconds, requests arrive at a rate of N/$\overline{T}$ requests per second. When $\overline{T}$ is much larger than L, the duration of a reservation request, we can model the arrival of requests as Poisson. Therefore the probability of having n arrivals during a period of time Δ is given by,

$$P(n) = \frac{(N\Delta / \overline{T})^n e^{-(N\Delta / \overline{T})}}{n!}.$$

We are interested in computing the average amount of time that is takes a successful request to get through to the scheduler. If it were not for collisions each terminal would get a successful request every $\overline{T}$ seconds. However, due to collisions, some requests will fail and the average amount of time between successful requests will increase. With an unslotted protocol, a request will be successful if no other requests were made in the 2L time period before the end of the transmission. This will happen with probability, $P(0) = e^{-(2L(N-1)/\overline{T})}$, and the average number of transmission attempts per successful transmission is $e^{(2L(N-1)/\overline{T})}$. Therefore, on average, every terminal gets a successful request every Λ seconds, where Λ is given by,

$$\Lambda = \overline{T} e^{(2L(N-1)/\overline{T})}$$

We can now choose $\overline{T}$ to minimize the average access time to the control channel. This can be done by taking the derivative of Λ with respect to $\overline{T}$ and setting it equal to 0,

$$d\Lambda / d\overline{T} = e^{2L(N-1)/\overline{T}} - 2L(N-1)e^{2L(N-1)/\overline{T}} / \overline{T} = 0$$
$$\Rightarrow \overline{T} = 2L(N-1)$$

Hence, the value of $\overline{T}$= 2L(N-1) minimizes the access delay and the resulting access delay is $\Lambda_{\text{min}} = 2(N-1)Le$. For example, in a system with N=100 nodes, a transmission rate of 10 Gbps and a control message size of 100 bits, a terminal would send a reservation request on average every 2μs and the average access delay for a successful reservation would be about 5.5μs.

C. Scheduling algorithm

In order to simplify the design of the scheduler we use a slotted system where requests are made for fixed size slots and the scheduler maintains a slotted reservation system. However, it is important to note that the OTs remain unslotted and unsynchronized. All of the timing is controlled by the scheduler using the master/slave protocol described in the previous section.

In a WDM system with a single transmitter and receiver per node, scheduling is constrained by the number of wavelengths, W, which limits the number of requests served during a slot to W. It is also constrained by the fact that each node has a single transmitter and a single receiver.

[1] Since sending the complete state information may lead to very large reservation messages, reservation requests may contain only partial information (e.g., first ten requests).

[2] Notice that unlike a random backoff algorithm where information about the success or failure of a transmission is available. Here we do not rely on any such information but rather periodically send the state of the queue. Of course, the state of the queue changes as successful requests are answered by the scheduler.

Therefore, during a given slot , each node can be scheduled for at most one transmission and one reception. This, in fact, is a very similar problem to that of scheduling transmissions in an input queued switch. In the case of an input queued switch it is known that when a First-Come-First-Serve service discipline is employed, under uniform traffic, throughput is limited to $2-\sqrt{2}=0.585$ [13]. This throughput limitation is due to the head-of-line (HOL) blocking effect, where transmissions are prevented because the packet at the head of the queue cannot be scheduled due to a receiver conflict. It is also known that if nodes are allowed to look-ahead into their buffers and transmit a packet other than the one at the head of the queue, the effect of HOL blocking can be significantly reduced [14]. Scheduling algorithms based on bipartite graph matching algorithms have been proposed that achieve full utilization under uniform and non-uniform traffic conditions [15,16]. However, it is also known that these algorithms are computationally intensive and require $O\left(M^{2.5}\right)$ operations to be implemented, where M is the number of input and output ports on the switch [17].

The network in this paper is being developed to support an enormous traffic volume. For example, with 30 data wavelengths operating at 10 Gbps each and an average slot size of 10,000 bits, 30 million slots have to be scheduled every second. This requirement makes the implementation of a complicated scheduling algorithm impractical with present technology. We therefore resort to a simpler, although sub-optimal, algorithm.

Our scheduling algorithm is based on input queues. The algorithm is made efficient through the use of a "look-ahead" window that allows the scheduler to look-ahead into each input queue and schedule requests that are not necessarily at the head of their queue. A look-ahead capability of k, allows the scheduler to look as far as the kth request in the queue. The algorithm is implemented on a slot-by-slot basis to form a schedule for the given slot. The algorithm works by maintaining N request queues, each containing the transmission requests from one of the N nodes in the network. The algorithm visits every node in some order (perhaps random) and starting with the first request in the queue it searches for a request that can be scheduled. That is, it searches for a request for a transmission to a receiver that has not been assigned yet. The algorithm searches the queue until depth k has been reached. If a request has been found, a wavelength is assigned to it. This process is continued until either all of the request queues have been visited or all W wavelengths have been assigned. During the next slot, the algorithm starts anew with the first request in each queue.

Figure 4 shows an example of the scheduling algorithm with three nodes. Shown in the figure is the destination of each request. After the first request in queue 1 is selected, the second request in queue 2 is selected leaving no available receivers for node 3 to communicate with. Notice, that the algorithm is clearly not maximal in the sense that there are other possible scheduling assignments that would allow all three nodes to transmit (e.g., 1-to-2, 2-to-3, and 3-to-1). Nonetheless, this algorithm improves considerably over an algorithm that looks only at the request at the head of the queue, and is only slightly more complicated to implement. In fact, it is clear from the description of the algorithm that the algorithm can be implemented in O(kxN) operations. A significant reduction in the number of operations compared to the graph matching algorithms.

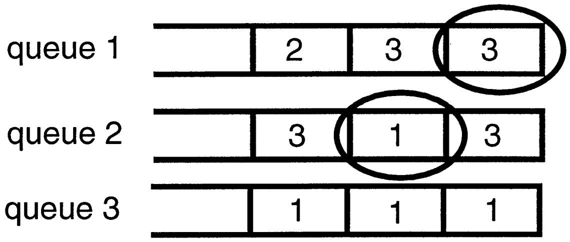

Figure 4. Example of the scheduling algorithm.

We analyze, through simulation, the maximum throughput that this algorithm can achieve. Table 1 shows the maximum achievable throughput under uniform traffic, with 30 data wavelengths. When the number of nodes is equal to the number of channels and no look-ahead is employed (i.e., k=1), HOL blocking limits throughput to 59% as predicted in [13]. However, a look-ahead window of just 4 packets can increase throughput to over 80%. As the number of nodes exceeds the number of channels the effect of HOL blocking is drastically reduced. This is due to two factors; first, the probability that multiple nodes have a packet at the HOL to the same destination is reduced due to the increase in the number of destinations, and second, with fewer channels than nodes the algorithm has many more requests from which to choose a schedule of W transmissions. As can be seen from the table, the combination of more nodes than channels and a look-ahead window of 4 or 5 packets virtually eliminates the effects of HOL blocking on throughput, under uniform traffic.

N	k=1	k=2	k=3	k=4	k=5	k=6	k=7
30	0.59	0.71	0.77	0.81	0.83	0.85	0.86
35	0.69	0.83	0.90	0.94	0.96	0.98	0.99
40	0.79	0.95	0.99	0.99	0.99	0.99	0.99
45	0.89	0.99	0.99	0.99	0.99	0.99	0.99
50	0.96	0.99	0.99	0.99	0.99	0.99	0.99
60	0.99	0.99	0.99	0.99	0.99	0.99	0.99

Table 1. Achievable throughput for a system with 30 wavelengths, N nodes and a look-ahead window k.

Scheduling multicast traffic in a WDM broadcast-and-select system is even more of a challenge because multicast messages have multiple intended receivers and trying to schedule transmissions in order to avoid receiver conflicts can be very inefficient. A simple and efficient multicast algorithm, based on random scheduling, is presented in [12].

IV. Analysis of Queueing Delay

In order to analyze the average queueing delay in this system we assume that packets arrive to each of the N nodes according to a Poisson random process of rate λ. We again assume that all packets are of the same length and take 1 slot to transmit and that the scheduler uses the slotted scheduling algorithm described in the previous section and that all transmissions are scheduled to occur at the beginning of a time slot.

Clearly in this system the queues at each on the N nodes are dependent on one another which makes analysis of the system difficult. This system can be analyzed using an N^2-dimensional, discrete-time, infinite Markov chain representing the number of requests (packets) between each of the N^2 source/destination pairs[3]. However, obtaining closed form expressions for the steady-state behavior of interacting queues is generally very difficult. Even numerical evaluation can be computationally complex [18]. An approximate analysis for this system, based on an independent approximation is presented in [19]. Here, for brevity, we present simulation results.

Shown in figure 5 is the simulated delay for a system with 100 nodes and 30 wavelengths. Notice that with these values the arrival rate of new packets to a user cannot exceed 0.3 due to the channel constraint. Furthermore, the maximum throughput may be decreased due to the HOL blocking effect, but as can be seen from table 2, the HOL blocking effect on maximum throughput is minimal for these values of N and W. Hence we expect that the maximum achievable arrival rate per node will be close to 0.3. Also notice from the figure that a look-ahead of just 2 packets can significantly help in reducing delays. However, a larger look-ahead window does not reduce delay any further because for these values of N and W, a look-ahead of just two packets essentially eliminates the HOL blocking effect.

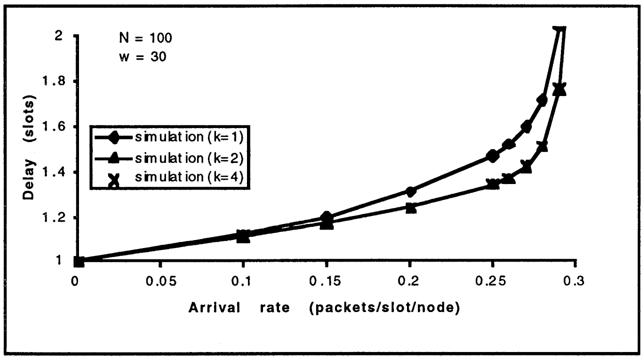

Figure 5. Delay vs. load for a system with 100 nodes and 30 wavelengths and a look-ahead capability (k).

V. Extension to MANs

This system is extended to Metropolitan Area Networks (MANs) with a layered architecture, shown in figure 6, where LAN hubs are interconnected through a MAN hub. The physical architecture of this network is similar to the one used in the All Optical Network (AON) testbed [11]. The MAN hub can be as simple as another broadcast star, a wavelength router, or a fully configurable optical switch. In this system, operation within a single LAN hub remains as described previously. For communication between LANs OTs send their requests to the MAN hub. The MAN hub computes the transmission schedules and forwards them to the respective LAN hubs which notify the OTs when and where to transmit. Of course, since the LAN hub consists of a passive device, certain wavelengths (denoted by λ_{LAN}) will have to be assigned for use within a LAN and the rest of the wavelengths (denoted by λ_{MAN}) will be used for communication between LANs. In order to simplify the task of scheduling, LAN hubs are synchronized to the MAN hub clock. However, OTs remain unsynchronized.

[3] Keeping track of queue sizes only is not sufficient because of the receiver contention problem.

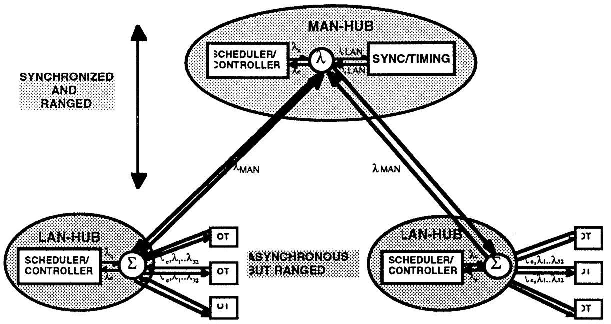

Figure 6. Extension to MANs.

VI. Conclusions

This paper describes an architecture and MAC protocol for providing bandwidth on demand in a WDM system. A driving principle in the design was to minimize the cost of the user terminal. To that end, our system uses a single transceiver per node and does not requires terminals to be slotted or synchronized. Transmissions are efficiently scheduled using a simple master/slave scheduler located at a hub node. The scheduler is also able to overcome the effects of propagation delays by taking propagation delays into account in the scheduling of transmissions.

This novel system is applicable to high performance local area networks where multi-gigabit per second transmission can be achieved. Another important application area for this system is in optical access networks, where a WDM Passive Optical Network (PON) can be used to provide connectivity between the customer premise and a central office. This MAC protocol, with a scheduler located at the central office, can be used to allow users to share wavelengths over the PON.

References

[1] B. Mukherjee, "WDM-Based Local Lightwave Networks Part I: Single-Hop Systems," IEEE Network, May, 1992.

[2] G. N. M. Sudhakar, M. Kavehrad, N.D. Georganas, "Access Protocols for Passive Optical Star Networks," Computer Networks and ISDN Systems, pp. 913-930, 1994.

[3] N. Mehravari, "Performance and Protocol Improvements for Very High Speed Optical Fiber Local Area Networks Using a Passive Star Topology," Journal of Lightwave technology, April, 1990.

[4] M.S. Chen, N.R. Dono, R. Ramaswami, "A New Media Access Protocol for Packet Switched Wavelength Division Multiaccess Metropolitan Network," JSAC, August, 1990.

[5] H. B. Jeon and C. K. Un, "Contention Based Reservation Protocols in Multiwavelength Protocols with Passive Star Topology," ICC'92.

[6] I. Chlamtac and A. Ganz, "Channel Allocation Protocols in Frequency-time Controlled High-Speed Networks," IEEE Transactions on Communications, April, 1988.

[7] F. Jia and B. Mukherejee, "The Receiver Collision Avoidance (RCA) Protocol for Single hop WDM Lightwave Networks," ICC'92, Chicage, June, 1992.

[8] I. M. I. Habib, M. Kavehrad, C.-E. W. Sundberg, "Protocols for Very High-Speed Optical Fiber Local Area Networks Using a Passive Star Topology," J. Lightwave Technology, December, 1987.

[9] G.N.M. Sudhakar, N.D. Georganas and M. Kavehrad, "A Multi-channel Optical Star LAN and its Application as a Broadband Switch," ICC '92, June, 1992.

[10] P.Dowd, "Random Access Protocols for High Speed Interprocess Communications Based on a Passive Optical Star Topology," Journal of Lightwave Technology, June, 1991.

[11] I.P. Kaminow, et. al., "A Wideband All-Optical WDM Network", IEEE JSAC, June, 1996.

[12] Eytan Modiano, "Unscheduled Multicasts in WDM Broadcast-and-Select Networks," Infocom '98, San Francisco, CA, March, 1998.

[13] M.J. Karol, M.G. Hluchyj and S.P. Morgan, "Input Versus Output Queueing in a Space-Division Packet Switch," IEEE Transactions on Communications, December, 1987.

[14] M.G. Hluchyj and M.J. Karol, "Queueing in High-Performance Packet Switching," JSAC, December 1988.

[15] K. M. Sivalingam and J. Wang, "Media Access Protocols for WDM Networks with On-Line Scheduling," Journal of Lightwave Technology, June, 1996.

[16] N. McKeown, V. Anantharam and J. Walrand, "Achieving 100% Throughput in an Input Queued Switch," Infocom '96, San Francisco, CA, April 1996.

[17] J.E. Hopcroft and R.M. Karp, "An $n^{5/2}$ Algorithm for Finding Maximal Matching in Bipartite Graphs," Society for Industrial and Applied Math. Journal of Computing, Feb. 1973.

[18] E. Modiano and A. Ephremides, "A Method for Delay Analysis of Interacting Queues in Multiple Access Systems," INFOCOM 93, San Francisco, CA, March, 1993.

[19] E. Modiano, "Design and Analysis of a WDM Network Using a Master/Slave Scheduler," in preparation.