Minimizing Delay and Packet Loss in Single-Hop Lightwave WDM Networks Using TDMA Schedules

George N. Rouskas
Department of Computer Science
North Carolina State University
Raleigh, NC 27695-8206

Mostafa H. Ammar
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332-0280

Abstract

We consider packet-switched lightwave WDM networks with stations equipped with tunable transmitters and fixed receivers. Access to each of the available channels is controlled by a *weighted* TDMA scheme, whereby the channels are not necessarily shared equally among the various sources. In this paper we study the problem of designing TDMA frames to minimize the mean packet delay, as well as the mean packet loss probability given a finite buffer capacity. We develop optimization methods which, for non-uniform communication patterns common to parallel and distributed computations, represent a significant improvement over I-TDMA*. Furthermore, the margin of improvement increases with the size of the network. Our main contribution is to present relatively simple media access control schemes which, in the general case (i.e., non-uniform traffic), achieve good performance in terms of delay, throughput, and packet loss.

Keywords: Single-hop optical networks, Wavelength Division Multiplexing (WDM)

1 Introduction

Wavelength division multiplexing (WDM) is currently believed to be the most promising technology for implementing a new generation of computer communication networks that fully exploit the vast information-carrying capacity of single-mode fiber [12]. By carving the bandwidth of the optical medium into multiple concurrent channels, WDM has the potential of delivering an aggregate throughput that can be in the order of Terabits per second. At the same time, WDM has introduced a new set of media-access problems, on which a great deal of recent research has been devoted. In this paper we focus on one of the candidate network architectures, namely the single-hop systems (see [20] for an overview).

As their name implies, single-hop networks provide one hop communication between any source-destination pair, by allowing the various stations to select any of the available channels for packet transmission/reception. Access to the channels can be based on a reservation scheme that requires the use of one [13, 19, 5, 6] or more [16] separate control channels (the last reference also contains a performance comparison of some of the schemes that have appeared in the literature). Alternatively, a hybrid time-wavelength division multiple access (T-WDMA) approach may be employed, in which case the bandwidth of each channel may be preallocated to each of the sources by means of a transmission schedule that indicates the slots in which the various stations may access the available channels [7, 3].

This work explores the delay and packet loss probability behavior of transmission schedules. In particular, we are interested in developing schedules that will have good performance under the (potentially non-uniform) traffic patterns one expects to encounter in realistic parallel and distributed computing environments. Previous work by the same authors [24, 23] focused on the throughput behavior of T-WDMA schedules; our main new contribution in this work is a relatively simple media access control scheme which, in the general case (i.e., non-uniform traffic), has good performance not only in terms of throughput, but also in terms of delay and packet loss. A round-robin T-WDMA protocol (I-TDMA*) was studied in [3] and its delay characteristics under uniform traffic were obtained. Numerical results to be presented indicate that our approach achieves significant performance gains over I-TDMA*, especially as the size of the network grows.

As in [3, 13, 19, 6], we consider environments in which the latency of tunable transceivers is small with respect to the packet transmission times (more on this later). The work in [4, 1, 21, 25, 10], on the other hand, assumes that tuning latency is comparable to packet transmission time. In [4, 1, 21, 25] heuristics to construct schedules that hide the tuning latency are developed under various traffic assumptions. The approach taken in [10] is somewhat different. Time is divided into transmitting and tuning epochs; during the latter, no packets are transmitted, but rather, transceivers are retuned to be ready for the next transmitting epoch. The goal of the design is to minimize the overall length of the schedule.

The remainder of the paper is organized as follows. In Section 2 we present a model that captures the salient features of our system, and in Section 3 we show how to select the minimum frame length to insure stability. A heuristic for the problem of minimizing the mean packet delay across the network is developed in Section 4. Section 5 presents a dynamic programming approach to optimally allocate the finite buffer capacity at each station so that the packet loss probability is minimized. We present numerical results in Section 6, and summarize our work in Section 7.

2 System Model

We consider a network of N stations, each equipped with one receiver and one transmitter, interconnected through an optical broadcast medium that can support C wavelengths, $\lambda_1, \lambda_2, \cdots, \lambda_C$. Each wavelength can be considered as a channel operating at a data rate accessible by the electronic interfaces at each station. In order to access the channels, the stations must employ tunable transmitters and/or receivers. For simplicity we only consider systems with tunable transmitters and fixed receivers; our results can be easily extended to tunable-receiver systems.

The fixed receiver of station i is assigned wavelength $\lambda(i) \in \{\lambda_1, \cdots, \lambda_C\}$; in other words, i may only receive packets transmitted on channel $\lambda(i)$. The transmitters, on the other hand, are lasers that can be tuned to, and transmit on any and all wavelengths $\lambda_c, c = 1, \cdots, C$. An important parameter in such a system is the *tuning delay*, or the time it takes a transmitter

to tune from one wavelength to another. The tuning delay may vary with the transmitters and/or the wavelength pairs.

The network operates in a slotted mode, with a slot time equal to the packet transmission time plus the tuning delay. In addition, some padding (guard time) is necessary within each slot to counter the effects of dispersion and to synchronize the transmission of slots on the different wavelengths [26]; the size of this padding is very small for LAN/MAN distances [26], and will be neglected in our work. A *collision* occurs if two or more transmitters access the same channel in a given slot. All packets involved in a collision are lost; recovery is assumed to take place via a higher level protocol.

We define σ_i as the probability that a new packet arrives at station i during a slot time, and let p_{ij} denote the probability that a packet arriving at station i has station j as its destination, with $\sum_j p_{ij} = 1$. Thus, $\Sigma = [\sigma_i p_{ij}]$ is the matrix of externally offered traffic, in units of packets per slot.

2.1 Channel Sharing

If the number of wavelengths, C, is equal to the number of stations, N the fixed receiver at each station i is assigned a unique wavelength, $\lambda(i) \in \{\lambda_1, \cdots, \lambda_C\}$, or *home* channel. Consequently, total *optical self-routing* [3, 9] is achieved, as a station only receives packets destined to itself. While optical self-routing is clearly a desirable feature, technical considerations make it difficult to achieve. First, current WDM technology may support only a small number of wavelengths in a single mode fiber, making self-routing architectures unsuitable for anything but trivial networks. Second, in order to keep channel utilization at high levels, the maximum tuning delay should only constitute a small fraction of the slot time. However, there is a tradeoff between the tuning range and tuning speed in state of the art tunable lasers and optical filters [12]; that is, the fastest tunable transceivers may only tune over a small portion of the optical bandwidth. As a result, even if a large number of wavelengths could be supported within a fiber, because of tunability considerations, only a subset of these

channels would be usable. To allow for network scalability [1], all the techniques developed in this work are applicable in the general case, i.e., for a number of wavelengths less than or equal to the number of stations.

Whenever $C < N$, a number of receivers have to be assigned the same wavelength $\lambda_c, c = 1, \cdots, C$. We let $R_c \subseteq \{1, \cdots, N\}$, denote the set of receivers that share channel λ_c,

$$R_c \quad = \quad \{j \mid \lambda(j) = \lambda_c\} \quad c = 1, \cdots, C \tag{1}$$

Intuitively, sets R_c should be constructed so that the traffic load be balanced across the various channels. Load-balancing is a well-known and widely-studied NP-complete problem, and various heuristics and approximation schemes have been developed for it [11, 8]. If *slowly-tunable* receivers are available, it may be desirable to periodically reconfigure the network by retuning the receivers, to make sure that the connectivity keeps up with dynamically changing traffic demands [2]. Since reconfiguration is expected to be a relatively infrequent event, for the remainder of this paper we assume that sets R_c are given, and that they have been determined so as to evenly spread the load over all channels.

Note that a station $j \in R_c$ will receive all packets transmitted on channel λ_c, and will have to filter out those addressed to stations $j' \in R_c, j' \neq j$. Thus, only *partial* self-routing is achieved. We now define

$$q_{ic} \quad = \quad \sigma_i \sum_{j \in R_c} p_{ij} \quad i = 1, \cdots, N, \ c = 1, \cdots, C \tag{2}$$

as the probability that a packet with destination $j \in R_c$ arrives at i within a slot. Each station has C buffers, one for storing packets that need to be transmitted on each channel. Having C separate queues eliminates the head-of-line effects of a single buffer. The buffer for channel λ_c at station i has a capacity of L_{ic} packets. Packets arriving to find a full buffer are lost.

[1] The maximum number of stations in the network *is* limited by the power budget [14]; what is implied here is that it should not be wavelength-limited.

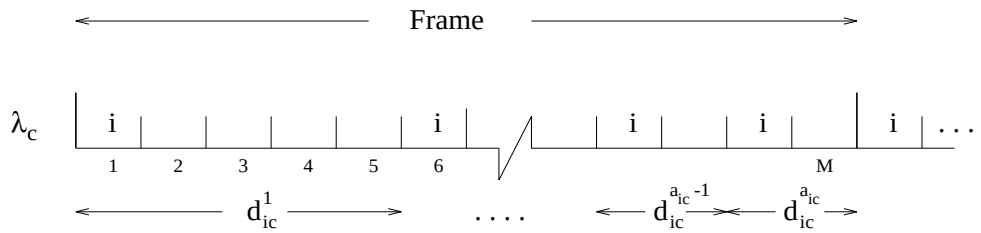


Figure 1: Definition of $d_{ic}^{(k)}$ for $k = 1, \cdots, a_{ic}(M)$

2.2 Transmission Schedules

The Interleaved TDMA (I-TDMA*) protocol [3] is an extension of time division multiple access (TDMA) over a multi-channel environment. In I-TDMA*, each station has exactly one chance per frame to transmit on each channel. I-TDMA* exhibits good performance under uniform traffic (i.e., when $\sigma_i = \sigma_k, p_{ij} = p_{kl}\ \forall\ i, j, k, l$), but will be shown to perform poorly under non-uniform loads one expects to encounter in realistic distributed and parallel computing environments. Here, we are concerned with *weighted* TDMA schemes, a generalization of I-TDMA*, whereby stations do not share the channels equally.

In a weighted TDMA scheme each frame consists of $M \geq N$ slots. Within each frame, source i is allowed to transmit on channel λ_c in exactly $a_{ic}(M)$ slots, $1 \leq a_{ic}(M) \leq M$. In these slots, i may transmit a packet to any station $j \in R_c$. A *transmission schedule* indicates, for all i and c, which slots within a frame can be used for transmissions from i on wavelength λ_c, and is described by variables $\delta_{ic}^{(t)}, t = 1, 2, \cdots, M$, called *permissions*, and defined as

$$\delta_{ic}^{(t)} = \begin{cases} 1, & \text{if station } i \text{ has permission to transmit on channel } \lambda_c \text{ in slot } t \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Then, $a_{ic}(M) = \sum_{t=1}^{M} \delta_{ic}^{(t)}$. For $k = 1, 2, \cdots, a_{ic}(M)$, we let $d_{ic}^{(k)}$ denote the distance, in slots, between the beginning of the k-th slot that i has permission to transmit on λ_c, and the beginning of the next such slot, in the same or the next frame (see Figure 1).

In particular, we are interested in developing schedules such that no collisions will ever occur. We then have the following definition:

Definition 1 *A schedule of frame length M provides full connectivity in the* **strong** *sense*

iff it satisfies the following three conditions:

$$q_{ic} > 0 \quad \Rightarrow \quad a_{ic}(M) \geq 1 \qquad \forall\, i, c \tag{4}$$

$$\sum_{c=1}^{C} \delta_{ic}^{(t)} \quad \leq \quad 1 \qquad \forall\, i, t \tag{5}$$

$$\sum_{i=1}^{N} \delta_{ic}^{(t)} \quad = \quad 1 \qquad \forall\, c, t \tag{6}$$

Condition (4) specifies that, if the traffic originating at station i and terminating at stations listening on wavelength λ_c is nonzero, then there is at least one slot per frame in which i may transmit on wavelength λ_c. This guarantees full connectivity among the network stations. Constraint (5) requires that each station be given permission to transmit on at most one channel within a slot t. Finally, constraint (6) implies that exactly one source may transmit on a given channel within a slot t. The last two constraints guarantee a collision-free operation. I-TDMA* is a special case of such a schedule with $M = N$ (for $C < N$) or $M = N \Leftrightarrow 1$ (for $C = N$), and $a_{ic}(M) = 1\ \forall\ i, c$.

By summing over all $t = 1, \cdots, M$, constraints (5) and (6) imply that:

$$\sum_{c=1}^{C} a_{ic}(M) \quad \leq \quad M \quad \forall\, i \tag{7}$$

$$\sum_{i=1}^{N} a_{ic}(M) \quad = \quad M \quad \forall\, c \tag{8}$$

i.e., that a source may not be given permission to transmit in more than M slots within a frame, and that exactly M slots contain permissions for transmission on each channel. In [22, Appendix B] it has been shown that (7) and (8) are also sufficient for constructing a schedule that satisfies (5) and (6), leading to the following corollary:

Corollary 1 *A schedule of frame length M providing full connectivity in the strong sense exists iff (4), (7), and (8) are satisfied.*

Figure 2 illustrates two schedules providing full connectivity in the strong sense for a network with $N = 4$ stations and $C = 2$ wavelengths. Channel λ_1 is shared by the receivers of stations 1 and 3 ($R_1 = \{1, 3\}$), while channel λ_2 is shared by the receivers of stations 2 and

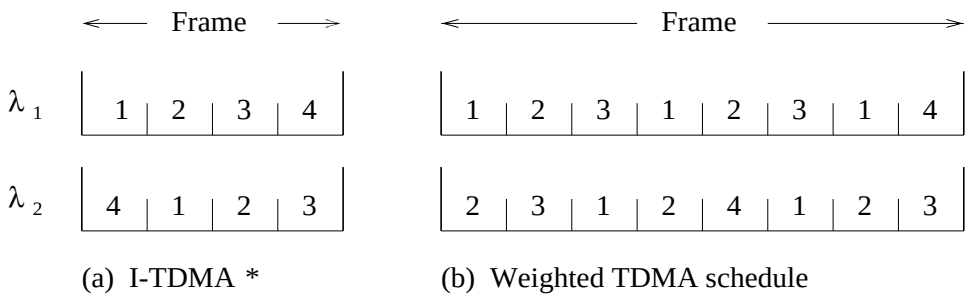


Figure 2: (a) I-TDMA*, and (b) weighted schedule providing full connectivity in the strong sense, for a network with $N = 4, C = 2, R_1 = \{1, 3\}, R_2 = \{2, 4\}$

4 ($R_2 = \{2, 4\}$). Figure 2(a) shows the I-TDMA* schedule, whereby each station is given exactly one permission within a frame to transmit on each channel. Figure 2(b) shows one possible weighted TDMA schedule which gives a different number of permissions per frame to each source-channel pair. Also note that no collisions are possible under either schedule.

For the rest of this paper we focus on determining schedules that provide full connectivity in the strong sense. In particular, we study the problem of obtaining the frame length M and quantities $a_{ic}(M), i = 1, \cdots, N,\ c = 1, \cdots, C$, to optimize certain performance measures (discussed in the next subsection). Quantity $a_{ic}(M)$ can be seen as the number of slots per frame assigned to node i to transmit its incoming traffic intended for wavelength λ_c. By fixing $a_{ic}(M)$, we indirectly allocate a certain amount of the bandwidth of wavelength λ_c to node i. As the traffic varies, the schedule length M and $a_{ic}(M)$ may vary as well. In this paper we assume that M and $a_{ic}(M)$ are fixed, since this variation will most likely take place over longer scales in time. If the traffic pattern is slowly and predictably changing over time (as was assumed in [18]), a schedule may be precomputed for the expected new traffic conditions. If changes in the traffic pattern are not predictable, the network nodes (or a special node dedicated to managing the network) may monitor packet transmissions and apply statistical techniques to determine whether the overall conditions have changed in a way that significantly affects the optimality of the current schedule. The problem of determining *when* the schedule needs to be updated is beyond the scope of this paper; however, once such a decision has been reached and a new schedule computed (using the techniques described later), say, by a special node, the new schedule can be used immediately

after all nodes have received a copy of it.

2.3 Performance Parameters

We will be concerned with evaluating the performance of schedules in terms of average packet delay, aggregate throughput, and packet loss probability. Packet delay is defined as the number of slots elapsed between the arrival of a packet at its source and the slot in which the packet is transmitted on the appropriate channel. (This definition ignores propagation delay; the latter, however, is independent of the particular schedule used, and ignoring it will not affect our conclusions regarding the relative performance of the various schedules.) Throughput is defined as the expected number of packets successfully transmitted per slot, while packet loss probability is the probability that a new packet will find the buffers at its source full and will be discarded.

The above definition of throughput assumes that the tuning latency (which is included as part of every slot) is negligible compared to the packet transmission time, and thus a *padding* can be included within each slot to allow the lasers to switch between wavelengths. This assumption is reasonable for Local and Metropolitan Area Networks with data rates in the order of several hundred Megabits per second and relatively large packet sizes. As an example, if the data rate is 622 Megabits per second and the packet length is $10,000$ bits, the packet transmission time is approximately $16\mu s$. Since tunable lasers with sub-microsecond tuning times do exist today [12] only a very small fraction of the slot time would be wasted for tuning. On the other hand, including the tuning latency in each and every slot would be highly inefficient for fast ATM switching environments, characterized by very high data rates (in the order of Gigabits per second or more) and very small packet sizes (e.g., ATM cells). In these situations, the packet transmission time may be only a fraction of the tuning latency of even the fastest currently available lasers, and techniques to construct schedules to hide the tuning latency [25, 4, 1].

As a final observation, the techniques in [25, 4, 1] are computationally expensive, and, although they have been shown to achieve near-optimal results on the average, in the *worst case* they will construct schedules of length equal to $1 + \Lambda$ times the optimal (Λ is the *nor-*

malized tuning latency, expressed in units of packet transmission time), which is equivalent to including a padding of length Λ within each slot. We believe, therefore, that these techniques should be used only in environments where the tuning latency is comparable to, or greater than the packet transmission time. The high throughput and low delay performance achieved by the relatively simple schemes presented here make them applicable even when the tuning latency takes up, say, 10-15% of the total slot time. Furthermore, the above techniques require that packets by a given source destined to a particular channel be transmitted in *contiguous* slots within the schedule; our schemes do not impose such restrictions, and result in better delay performance, as will be seen shortly.

3 Selecting the Frame Length to Insure Stability

Let us now suppose that the C buffers at each station have infinite capacity ($L_{ic} = \infty \ \forall \ i, c$), and that the sets, R_c, of stations sharing channel λ_c have been decided upon. Observe that the buffers for distinct source-destination pairs do not interact, and thus are independent. Consequently, the necessary and sufficient condition for stability is that the number of slots within a frame in which station i is permitted to transmit on channel λ_c, be greater than the number of packets destined to a receiver listening on λ_c that are expected to arrive at station i during a number of slots equal to the frame length [2] [17]:

$$Mq_{ic} \quad < \quad a_{ic}(M) \quad \forall \ i, c \tag{9}$$

As we shall see, the poor performance of I-TDMA* under non-uniform traffic loads arises from its failure to satisfy the above condition, even when the load offered to each channel is less than 1. Now, $\lfloor Mq_{ic} \rfloor + 1$ is a lower bound on $a_{ic}(M)$ if the stability condition (9) is to be satisfied:

$$\lfloor Mq_{ic} \rfloor + 1 \quad \leq \quad a_{ic}(M) \ \ \forall \ i, c \tag{10}$$

[2] Note that, because of (8), (9) implies that $\sum_{i=1}^{N} q_{ic} < 1 \ \forall \ c$, i.e., that the load offered to each channel is less than 1. The latter, in turn, implies that $\sum_{i=1}^{N} \sigma_i < C$, or that the total offered load does not exceed the network capacity.

Because of (7) and (8), the following two conditions must hold:

$$\sum_{c=1}^{C}(\lfloor Mq_{ic}\rfloor + 1) \quad \leq \quad M \quad \forall\, i \tag{11}$$

$$\sum_{i=1}^{N}(\lfloor Mq_{ic}\rfloor + 1) \quad \leq \quad M \quad \forall\, c \tag{12}$$

However, since $Mq_{ic} < \lfloor Mq_{ic}\rfloor + 1$, it is easy to see that, unless M is sufficiently large, (11) and/or (12) may be violated, making it impossible to have $a_{ic}(M) \geq \lfloor Mq_{ic}\rfloor + 1$ as required by (9). We now show how to select M so that (11) and (12) are satisfied.

Consider channel λ_c, and select M'_c such that:

$$M'_c \sum_{i=1}^{N} q_{ic} \quad \leq \quad M'_c \Leftrightarrow N \quad \Leftrightarrow \quad M'_c \quad \geq \quad \frac{N}{1 \Leftrightarrow \sum_{i=1}^{N} q_{ic}} \tag{13}$$

For this frame length, M'_c, we get:

$$\sum_{i=1}^{N}\lfloor M'_c q_{ic}\rfloor \quad \leq \quad M'_c \sum_{i=1}^{N} q_{ic} \quad \leq \quad M'_c \Leftrightarrow N \quad \Leftrightarrow \quad \sum_{i=1}^{N}(\lfloor M'_c q_{ic}\rfloor + 1) \quad \leq \quad M'_c \tag{14}$$

Thus, by selecting $M' \geq \max_c\{M'_c\}$ we insure that (12) is satisfied.

By proceeding as above, we consider source i and select M''_i such that

$$M''_i \sum_{c=1}^{C} q_{ic} \quad \leq \quad M''_i \Leftrightarrow C \quad \Leftrightarrow \quad M''_i \quad \geq \quad \frac{C}{1 \Leftrightarrow \sum_{c=1}^{C} q_{ic}} \quad = \quad \frac{C}{1 \Leftrightarrow \sigma_i} \tag{15}$$

Then,

$$\sum_{i=1}^{C}\lfloor M''_i q_{ic}\rfloor \quad \leq \quad M''_i \sum_{c=1}^{C} q_{ic} \quad \leq \quad M''_i \Leftrightarrow C \quad \Leftrightarrow \quad \sum_{c=1}^{C}(\lfloor M''_i q_{ic}\rfloor + 1) \quad \leq \quad M''_i \tag{16}$$

Thus, frame length $M'' \geq \max_i\{M''_i\}$ insures that (11) is satisfied. In order to satisfy both (11) and (12), M has to be such that

$$M \quad \geq \quad \max\{M', M''\} \tag{17}$$

4 Minimization of the Average Packet Delay

We now turn our attention to the issue of constructing schedules such that the average packet delay over all source-channel pairs is minimized. Thus, we are seeking a solution to the following optimization problem.

Problem 1 *Given the number of stations, N, the number of available wavelengths, C, and the traffic parameters, $\sigma_i p_{ij}$, $i,j = 1,\cdots,N$, find a schedule such that the network-wide average packet delay is minimized, assuming that buffers of infinite capacity are available at each station.*

There are three dimensions to this problem [3]:

- the sets of receivers, R_c, sharing wavelength $\lambda_c, c = 1,\cdots,C$, must be constructed,
- the number of slots per frame, $a_{ic}(M)$, allocated to each source-channel pair (i,λ_c) must be obtained, and
- a way of placing the $a_{ic}(M)$ slots within the frame, for all i,λ_c, must be determined.

A similar study of a single-channel network [15] has shown that the optimization yields a very hard allocation problem. The corresponding multi-channel optimization problem is even harder, as the minimization is over all possible partitions of the set of receivers, $\{1,2,\cdots,N\}$, into sets $R_c, c = 1,\cdots,C$. Our approach, then, is to first construct sets R_c using a load-balancing heuristic [8]. In the following, we present a heuristic to obtain near-optimal schedules assuming that sets R_c are known.

Recall that the buffers for each source-channel pair are independent; therefore, if we consider each channel in isolation, all the results obtained in [15] will be applicable. We now review these results, which provide a lower bound on the multi-channel problem, *given a partition of* $\{1,\cdots,N\}$ *into sets* R_c.

Consider channel λ_c; the average packet delay is minimized when:

- The percentage of time station i is permitted to transmit on channel λ_c is [15, Eq. (3.3)]

$$x_{ic} \quad = \quad q_{ic} + (1 \Leftrightarrow \sum_{k=1}^{N} q_{kc}) \frac{\sqrt{1 \Leftrightarrow q_{ic}}}{\sum_{k=1}^{N} \sqrt{1 \Leftrightarrow q_{kc}}} \qquad \forall\ i \tag{18}$$

[3] This is just a logical decomposition of the optimization problem. The order in which the three subproblems are presented is irrelevant as the subproblems are interdependent, and an exact solution method would simultaneously resolve all of them.

- For each source, i, the $a_{ic}(M)$ permissions for i to transmit on channel λ_c are equally spaced within the frame, i.e.,

$$\forall\ i\ :\quad d_{ic}^{(k)}\ =\ d_{ic}\ =\ \frac{1}{x_{ic}},\quad k=1,\cdots,a_{ic}(M) \tag{19}$$

Note that x_{ic} and d_{ic} are independent of M. Given a frame length, M, satisfying (17), we assign a number of slots to the source-channel pair (i,λ_c) such that

$$\lfloor Mq_{ic}\rfloor+1\ \leq\ a_{ic}\ \leq\ \lceil Mx_{ic}\rceil \tag{20}$$

and constraints (7) and (8) hold.

4.1 Slot Allocation

Once $a_{ic}(M)$ have been determined for all i and λ_c, we need to construct the schedule so that the permissions assigned to each source-channel pair are placed within the frame according to (19). This is not feasible in general as d_{ic} may not be integers. Even if they are, scheduling the transmissions between all sources and channels in equally spaced slots may violate constraints (5) and (6). To overcome this problem in the single-channel case, a golden-ratio policy was developed in [15], requiring that the frame length be a Fibonacci number. It was also shown that this policy places the permissions within the frame in intervals close to the ones dictated by (19), and it achieves an average packet delay very close to the lower bound.

Our approach is to use the golden ratio policy to place the permissions within each channel independently of the others. Considering channels in isolation may cause a source to be assigned to transmit on two or more channels in the same slot, violating (5). If this occurs, we must rearrange the schedule to remove these violations (recall that, from Corollary 1, this is always possible, since $a_{ic}(M)$ satisfy both (7) and (8), and thus, a schedule providing full connectivity in the strong sense always exists). To this end, we use algorithm REARRANGE, described in [24], with a worst case complexity of $O(N^2M^2)$.

We now propose the following Slot Allocation Heuristic.

Slot Allocation Heuristic (SAH)

1. If $C < N$, use a load-balancing heuristic to determine the set of receivers, $R_c, c = 1, \cdots, C$, that share each channel.

2. Select the smallest Fibonacci number, M, that satisfies (17), and obtain $a_{ic}(M)$ from (20) so that (7) and (8) hold.

3. Let $c = 1$, and use the golden ratio policy [15] to place the $a_{ic}(M), i = 1, \cdots, N$, slots for transmissions on channel λ_c. Repeat for $c = 2, \cdots, C$ to obtain initial schedule $S_0(M)$.

4. Run algorithm REARRANGE [24] to perturb $S_0(M)$, producing a schedule, $S(M)$, satisfying constraints (5) and (6).

5. Repeat Steps 2 through 4 for the next Fibonacci number, up to an upper limit, M_{max}. Select the frame length, M, and schedule, $S(M)$, that yields the lowest average delay.

5 Minimization of the Packet Loss Probability

In Section 4 we presented a heuristic to minimize the average packet delay, or, equivalently, the *expected queue size* across the CN buffers (recall that each station supports C queues, one per channel). We will now use these schedules in a finite-buffer environment. The problem that arises then, can be stated as:

Problem 2 *Given the number of stations, N, the number of available wavelengths, C, the traffic parameters, $\sigma_i p_{ij}$, $i, j = 1, \cdots, N$, a partition of receivers into sets R_c, and the maximum number of buffers at each station, $L_{i,max}$, $i = 1, \cdots, N$, determine, for all stations i, the buffer size, L_{ic}, for packets waiting for transmission on channel λ_c, so that $\sum_{c=1}^{C} L_{ic} = L_{i,max}$, and the network-wide probability of packet loss is minimized.*

In the following we first derive a lower bound on the packet loss probability, and then give a dynamic programming formulation for the problem of partitioning the $L_{i,max}$ buffers at each station i into queue sizes $L_{i1}, L_{i2}, \cdots, L_{iC}$.

5.1 Packet Loss Analysis

The lower bound on the packet loss probability is based on the observation that the mean queue length for source-channel pair (i, λ_c) is minimized when i is assigned to transmit on λ_c in slots which are exactly d_{ic} slots apart (see (19)). Since the buffers for each source-channel pair are independent, we may consider pair (i, λ_c) in isolation.

We observe the system at the instants just before the beginning of slots in which i may transmit on λ_c. Consider the l-th such slot. We define $r_{ic}^{(l)}(n, L_{ic})$ as the probability that i has n packets in its buffer (of size L_{ic}) for λ_c at the beginning of the l-th slot, $0 \leq n \leq L_{ic}$. We also define $P_{ic}(v)$ as the probability that v packets for λ_c arrive at i in the d_{ic} slots between the beginning of the l-th slot and the beginning of the $(l+1)$-th slot:

$$P_{ic}(v) = \begin{cases} \binom{d_{ic}}{v} q_{ic}^{v}(1 \Leftrightarrow q_{ic})^{d_{ic}-v}, & 0 \leq v \leq d_{ic} \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

$P_{ic}(> v)$, the probability that more than v packets arrive at i in the d_{ic} slots can be similarly defined.

Source i will have $n, n = 1, \cdots, L_{ic} \Leftrightarrow 1$, packets for λ_c at the beginning of the l-th slot if (a) i had $n+1$ packets at the beginning of slot $l \Leftrightarrow 1$, transmitted one on λ_c in the $(l \Leftrightarrow 1)$-th slot, and no packets arrived since, and (b) i had $n \Leftrightarrow v$ packets, transmitted one, and $v+1$ packets arrived [4]. Similar observations can be made for $r_{ic}^{(l)}(L_{ic}, L_{ic})$. We can then write the following set of recursive equations for $l = 2, 3, \cdots$. The initial conditions (25) for $l = 1$ are obtained by assuming that the frame starts at a slot in which i may transmit on λ_c.

$$\begin{aligned} r_{ic}^{(l)}(n, L_{ic}) &= r_{ic}^{(l-1)}(n+1, L_{ic})P_{ic}(0) \\ &\quad + \sum_{v=0}^{n} r_{ic}^{(l-1)}(n \Leftrightarrow v, L_{ic})P_{ic}(\min(n, v+1)) \quad n = 1, \cdots, L_{ic} \Leftrightarrow 1 \end{aligned} \tag{22}$$

$$r_{ic}^{(l)}(L_{ic}, L_{ic}) = \sum_{v=0}^{L_{ic}} r_{ic}^{(l-1)}(L_{ic} \Leftrightarrow v, L_{ic})P_{ic}(\geq \min(L_{ic}, v+1)) \tag{23}$$

[4] Except when $v = n$, in which case we require that n packets arrive. In (22) this case is covered by using $\min\{n, v+1\}$.

$$r_{ic}^{(l)}(0, L_{ic}) \quad = \quad 1 \Leftrightarrow \sum_{n=1}^{L_{ic}} r_{ic}^{(l)}(n, L_{ic}) \tag{24}$$

$$r_{ic}^{(1)}(n, L_{ic}) \quad = \quad 0 \quad 1 \leq n \leq L_{ic} \quad , \quad r_{ic}^{(1)}(0, L_{ic}) \quad = \quad 1 \quad \text{(Initial Conditions)} \tag{25}$$

We can regard this system as a discrete-time, discrete-space Markov process. The state of the process denotes the number of packets in the buffer of i for channel λ_c. State transitions occur at the end of each slot, and the transition probabilities are time independent. Then, (22) through (24) describe the transient behavior of the system. Since the buffer can become empty, and from the empty state the system can reach any other state, the Markov process consists of a single chain. Therefore, the system will eventually reach a steady state [17] such that:

$$r_{ic}(n, L_{ic}) \quad = \quad \lim_{l \to \infty} r_{ic}^{(l)}(n, L_{ic}), \qquad n = 1, \cdots, L_{ic} \tag{26}$$

As an example, when $L_{ic} = 1$ we have that

$$r_{ic}(1, 1) \quad = \quad 1 \Leftrightarrow r_{ic}(0, 1) \quad = \quad 1 \Leftrightarrow (1 \Leftrightarrow q_{ic})^{d_{ic}} \tag{27}$$

while for $L_{ic} = 2$ we get

$$r_{ic}(2, 2) \quad = \quad r_{ic}(2, 2)\ P_{ic}(1) \quad + \quad P_{ic}(\geq 2) \tag{28}$$

$$r_{ic}(1, 2) \quad = \quad r_{ic}(2, 2)\,\{P_{ic}(0) \Leftrightarrow P_{ic}(1)\} \quad + \quad P_{ic}(1) \tag{29}$$

In general, $r_{ic}(n, L_{ic})$ may be determined by either solving the set of linear equations that result from (22), (23), and (24), when we replace $r_{ic}^{(l)}(n, L_{ic})$ with $r_{ic}(n, L_{ic})$, or by iteratively solving (22), (23), and (24) for $r_{ic}^{(l)}(n, L_{ic}), l = 2, 3, \cdots$, until they converge to $r_{ic}(n, L_{ic})$. Then the probability of a packet arriving at station i been lost, *given that the packet is destined to a receiver listening on λ_c and the buffer for that channel has a capacity of L_{ic} packets* is

$$Q_{ic}(L_{ic}) \quad = \quad \sum_{n=0}^{L_{ic}} r_{ic}(n, L_{ic}) P_{ic}(> L_{ic} \Leftrightarrow n) \tag{30}$$

The probability that a packet arriving at station i has to be transmitted on channel λ_c is just q_{ic}/σ_i. Therefore, the probability of packet loss for packets arriving at station i given a partition of the $L_{i,max}$ buffers into C queues of sizes $L_{i1}, \cdots, L_{iC}$ is

$$Q_i(L_{i1}, \cdots, L_{iC}) \quad = \quad \frac{1}{\sigma_i} \sum_{c=1}^{C} q_{ic} Q_{ic}(L_{ic}) \qquad L_{i1} + \cdots + L_{iC} = L_{i,max} \tag{31}$$

5.2 Optimal Buffer Partitioning

Problem 2 now reduces to obtaining, for all i, queue sizes $L_{i1}, \cdots, L_{iC}$, such that Q_i as given in (31) is minimized. As previously, we consider source i in isolation, and let us renumber the C channels (if necessary) so that $q_{i1} \geq q_{i2} \geq \cdots \geq q_{iC}$. Note that the optimal partition (the one which minimizes (31)) must be such that $L_{i1} \geq L_{i2} \geq \cdots \geq L_{iC}$, and $\sum_{c=1}^{N} L_{ic} = L_{i,max}$. Therefore, $L_{iC} \leq \lfloor \frac{L_{i,max}}{C} \rfloor$. Furthermore, if an optimal partition of the $L_{i,max}$ buffers into C queues is such that $L_{iC} = z$, then $L_{i1}, \cdots, L_{i,C-1}$ is an optimal partition of the $L_{i,max} \Leftrightarrow z$ buffers into $C \Leftrightarrow 1$ queues.

Let $Q_i^\star(L_{i1}, \cdots, L_{iC}; L_{i,max})$ denote the minimum packet loss probability when the total number of buffers at station i to be partitioned into C queues is $L_{i,max}$. We can then write the following recursive equation:

$$Q_i^\star(L_{i1}, \cdots, L_{iC}; L_{i,max}) \;=\; \min_{1 \leq L_{iC} \leq \lfloor \frac{L_{i,max}}{C} \rfloor} \left\{ \frac{1}{\sigma_i}\, q_{iC}\; Q_{iC}(L_{iC}) \;+\; Q_i^\star(L_{i1}, \cdots, L_{i,C-1}; L_{i,max} \Leftrightarrow L_{iC}) \right\} \quad C > 1 \tag{32}$$

$$Q_i^\star(L_{i1}; L) \;=\; \frac{1}{\sigma_i}\, q_{i1}\; Q_{i1}(L) \quad \forall\; L \tag{33}$$

The boundary condition (33) stems from the fact that, when there is only one channel, it is best to assign to it all the available buffers. Given $q_{i1}, \cdots, q_{iC}$, and $L_{i,max}$, we use (32) and (33) to determine the optimal partition of the available buffers into C queues.

6 Numerical Results

6.1 Average Packet Delay

We consider the 8-station mesh type, disconnected type, ring type and two-server traffic matrices with probabilities p_{ij} as shown in Figures 3, 4, 5, and 6, respectively. We let $\sigma_i = \sigma \; \forall \; i$; this does not compromise the generality of our results, as the traffic characteristics are determined by p_{ij}. Figure 4 also shows the weighted TDMA schedule of frame length $M = 21$ produced by the Slot Allocation Heuristic (SAH) for the disconnected type traffic

matrix, $C = 8$ available wavelengths, and $\sigma = 0.70$. We can see that, overall, SAH places the slots assigned to each source-destination pair so that their distances are very close to the ones dictated by (19). Similar behavior has been observed for the other traffic matrices, as well as a wide range of values for system parameters C and σ.

We used SAH (with $M_{max} = 2,584$) to construct optimized schedules for $C = 2, 4$, and 8 channels, and values of σ from 0.01 to 0.99. We then obtained through simulation the delay and throughput curves of these schedules, shown in Figures 3 to 6; the delay is given in slots, and the throughput in packets successfully received per slot. The asymptotes of the delay curves can be deduced from the corresponding throughput graph. The delay and throughput curves of the I-TDMA* schedule for the corresponding traffic matrices are also plotted; for a fair comparison, whenever $C < N$, we used the same sets of receivers sharing each channel for I-TDMA* as for the optimized schedules (i.e., those produced by a greedy load-balancing heuristic [8]).

It is immediately evident that the schedules constructed by SAH outperform I-TDMA* by a wide margin, in terms of both delay and throughput. In particular, there are situations, as in Figure 5, when an optimized schedule with $C = 2$ channels achieves better performance than the I-TDMA* schedule with the maximum number of channels, $C = 8$. The poor performance of I-TDMA*, even at very low offered loads, is due to the fact that it assigns exactly one slot per frame to each source-channel pair. As a result, the queue for source-channel pair (i, λ_c) will grow without bounds whenever $Mq_{ic} > 1$, i.e., when the average number of packets for λ_c arriving at source i within a frame is greater than 1. In Figure 5, and for the I-TDMA* schedule of frame length $M = 7$ ($C = 8$ channels), this condition is satisfied when $\sigma > .204$ for the source-channel pairs for which $q_{ic} = p_{ic} = 0.7$. The optimized schedules, on the other hand, assign a larger number of slots to source-channel pairs with high values of q_{ic}, and thus, are able to operate at significantly higher offered loads before their delay behavior is affected. In addition, the slots assigned to a given source-channel pair are placed in almost equal distances within the frame, which also guarantees a near-optimal performance in terms of delay (see (19)).

The limitations of I-TDMA*, and the potential for improvement by using the optimized schedules are more pronounced when one considers larger size networks. In Figure 7 we plot

the delay and throughput curves for a 20-station ring-type traffic matrix (not shown here, but similar to the one in Figure 5). As we can see, the delay under an I-TDMA* schedule with the maximum number of channels ($C = 20$) grows without bound even for offered loads $\sigma < 0.1$. In contrast, an optimized schedule with as few as $C = 3$ channels experiences finite delays for $\sigma = 0.1$, while one with $C = 20$ channels may operate under loads as high as $\sigma = 0.9$.

6.2 Packet Loss

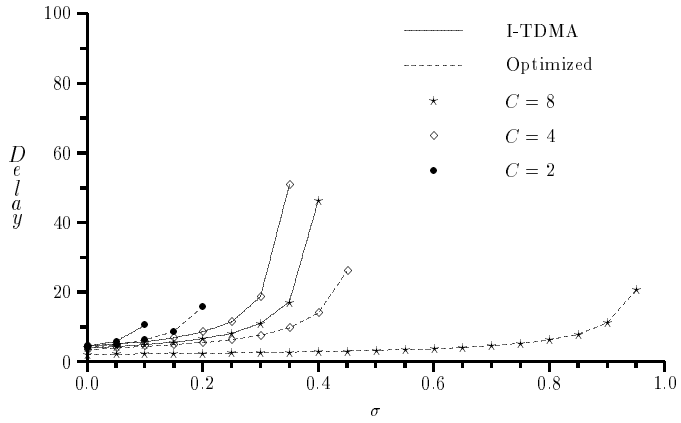
We again consider the 8- and 20-station traffic matrices studied in the previous section, but we now assume that each station employs a finite number of buffers, $L_{i,max}$. Without loss of generality, we let $L_{i,max} = L_{max}\ \forall\ i$. Our objective is to compare the packet loss probability under two scenarios: (a) when the $L_{i,max}$ buffers available at each station are allocated according to (32), and (b) when the $L_{i,max}$ buffers are equally partitioned among the various channels. Only schedules optimized for packet loss (finite buffers) are considered in this section.

Figures 8 to 10 plot the packet loss probability curves against the total number of buffers at each station, L_{max}, for various traffic matrices and various system parameters ($C = 8, \sigma = 0.7$ and $C = 4, \sigma = 0.3$ for the 8-station matrices, and $C = 10, 20, \sigma = 0.3$ for the 20-station matrix). The curves were obtained through simulation. Label "Equal" is used in the figures to denote scenario (b) above, i.e., the equal sharing of buffers among the channels. The plots indicate that, as the number of buffers increases, the buffer allocation determined by (32) results in a performance improvement between one and four orders of magnitude over an equal partitioning scheme, depending on the traffic matrix and system parameters and the number of available channels.

7 Concluding Remarks

We have considered single-hop WDM networks in which access to the various channels is controlled by weighted transmission schedules. We have addressed the problems of minimizing

0	0.33	0.33	0	0.34	0	0	0
0.33	0	0	0.33	0	0.34	0	0
0.33	0	0	0.33	0	0	0.34	0
0	0.33	0.33	0	0	0	0	0.34
0.34	0	0	0	0	0.33	0.33	0
0	0.34	0	0	0.33	0	0	0.33
0	0	0.34	0	0.33	0	0	0.33
0	0	0	0.34	0	0.33	0.33	0



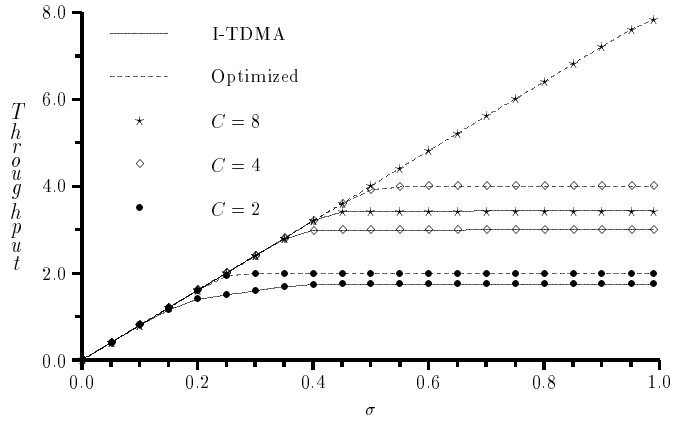
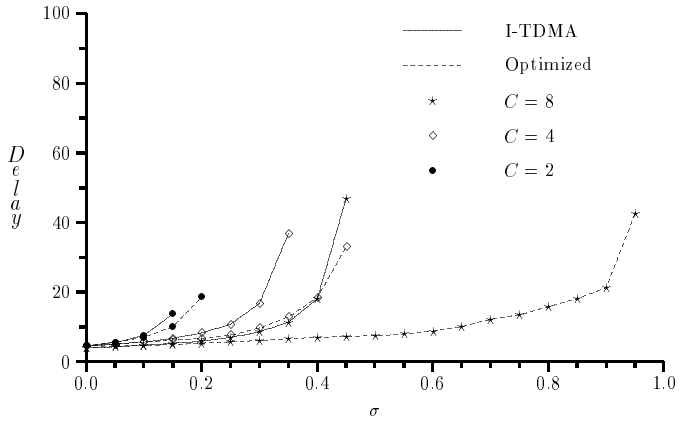


Figure 3: 8-station mesh-type traffic matrix and delay and throughput curves

0	0.30	0.30	0.30	0.025	0.025	0.025	0.025
0.30	0	0.30	0.30	0.025	0.025	0.025	0.025
0.30	0.30	0	0.30	0.025	0.025	0.025	0.025
0.30	0.30	0.30	0	0.025	0.025	0.025	0.025
0.025	0.025	0.025	0.025	0	0.30	0.30	0.30
0.025	0.025	0.025	0.025	0.30	0	0.30	0.30
0.025	0.025	0.025	0.025	0.30	0.30	0	0.30
0.025	0.025	0.025	0.025	0.30	0.30	0.30	0

Frame

2	4	3	6	4	2	5	3	8	4	2	6	3	2	4	3	7	4	2	5	3		
3	6	4	1	5	3	8	4	1	6	3	1	4	3	7	4	1	5	3	1	4		
4	1	5	2	8	4	1	7	2	1	4	2	8	4	1	6	2	1	4	2	7		
5	2	8	3	1	7	2	1	3	2	8	3	1	6	2	1	3	2	7	3	1		
6	8	7	8	2	6	6	8	4	7	1	7	2	8	6	7	8	3	1	7	6		
7	5	2	5	7	1	4	2	5	8	7	8	7	1	3	8	5	7	8	8	5		
8	3	1	4	6	8	3	5	6	5	5	4	6	5	8	2	6	8	5	6	8		
1	7	6	7	3	5	7	6	7	3	6	5	5	7	5	5	4	6	6	4	2		



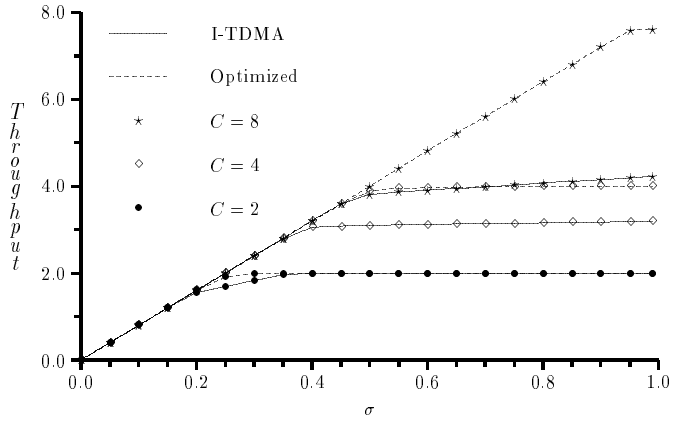
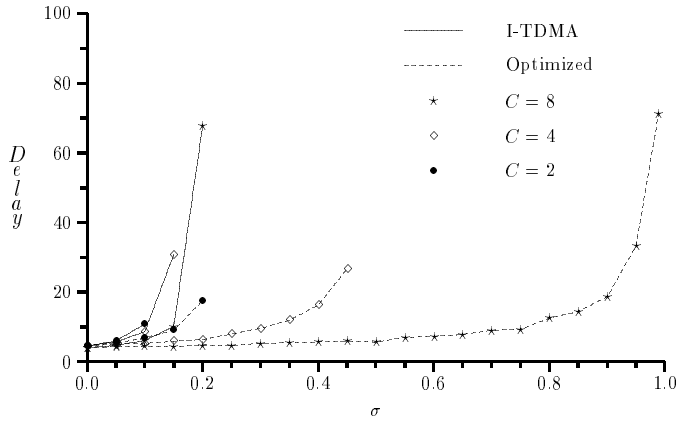


Figure 4: 8-station disconnected-type traffic matrix and delay and throughput curves

0	**0.70**	0.05	0.05	0.05	0.05	0.05	0.05
0.05	0	**0.70**	0.05	0.05	0.05	0.05	0.05
0.05	0.05	0	**0.70**	0.05	0.05	0.05	0.05
0.05	0.05	0.05	0	**0.70**	0.05	0.05	0.05
0.05	0.05	0.05	0.05	0	**0.70**	0.05	0.05
0.05	0.05	0.05	0.05	0.05	0	**0.70**	0.05
0.05	0.05	0.05	0.05	0.05	0.05	0	**0.70**
0.70	0.05	0.05	0.05	0.05	0.05	0.05	0



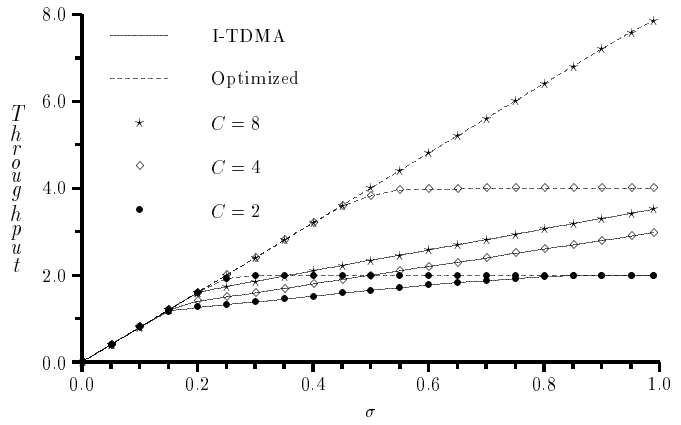
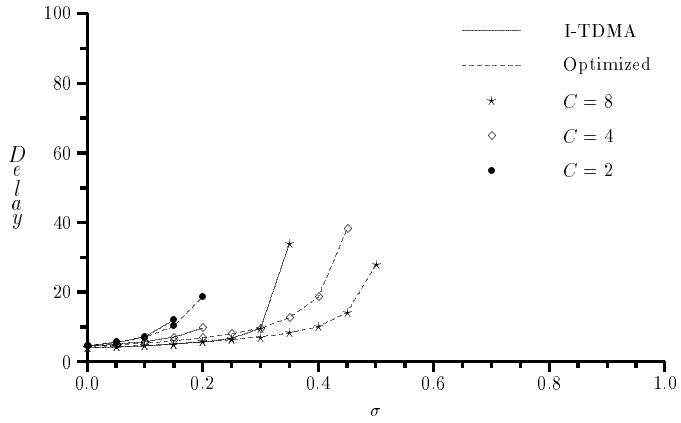


Figure 5: 8-station ring-type traffic matrix and delay and throughput curves

0	0.20	0.20	0.20	0.10	0.10	0.10	0.10
0.40	0	0.08	0.08	0.20	0.08	0.08	0.08
0.40	0.08	0	0.08	0.20	0.08	0.08	0.08
0.40	0.08	0.08	0	0.20	0.08	0.08	0.08
0.10	0.10	0.10	0.10	0	0.20	0.20	0.20
0.20	0.08	0.08	0.08	0.40	0	0.08	0.08
0.20	0.08	0.08	0.08	0.40	0.08	0	0.08
0.20	0.08	0.08	0.08	0.40	0.08	0.08	0



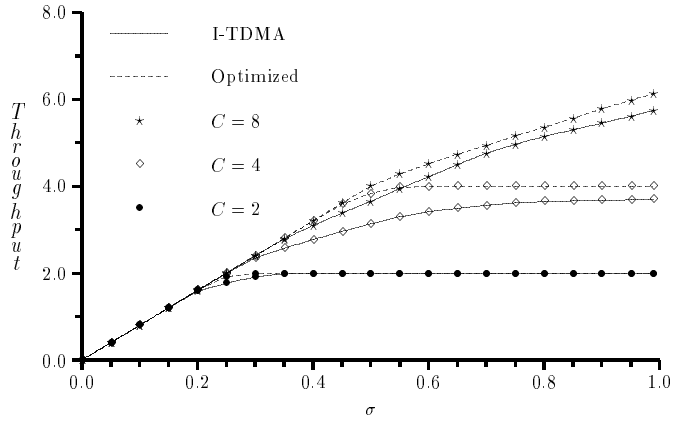


Figure 6: 8-station two-server-type traffic matrix and delay and throughput curves

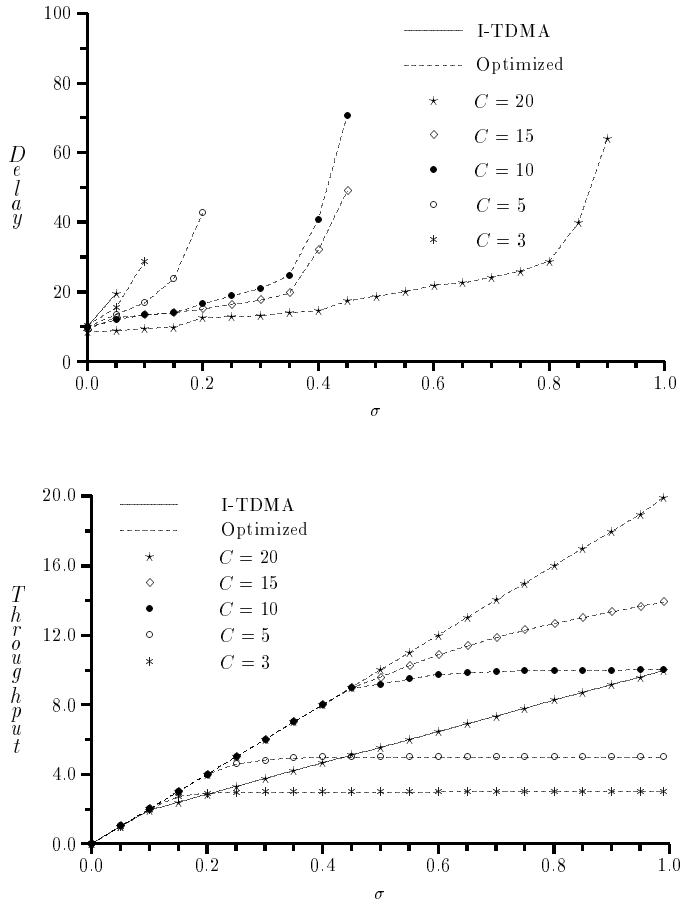


Figure 7: 20-station ring-type traffic matrix and delay and throughput curves

the delay and packet loss probability across the network. We have developed optimization methods that not only outperform previously proposed solutions, but also perform very well for communication patterns one expects to encounter in realistic environments. Techniques such as these are the first step towards lightwave WDM networks that dynamically adapt to changing traffic patterns.

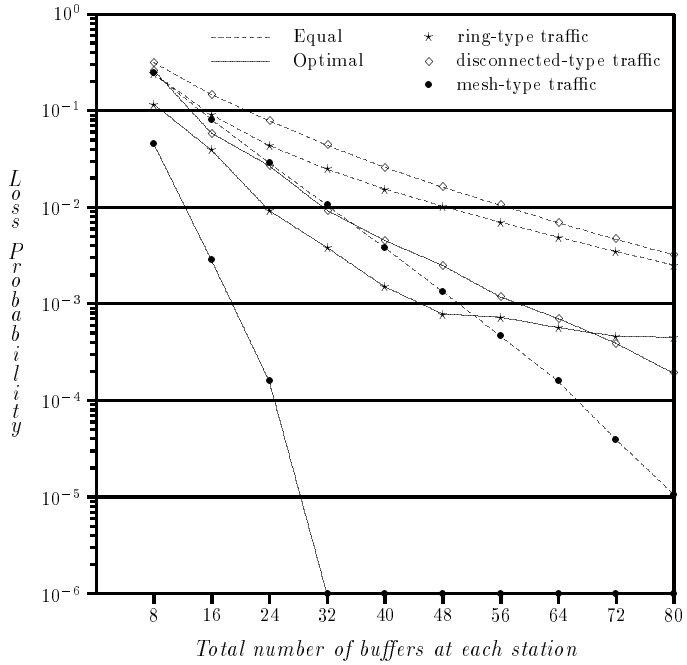


Figure 8: 8-station loss probability curves ($C = 8, \sigma = 0.7$)

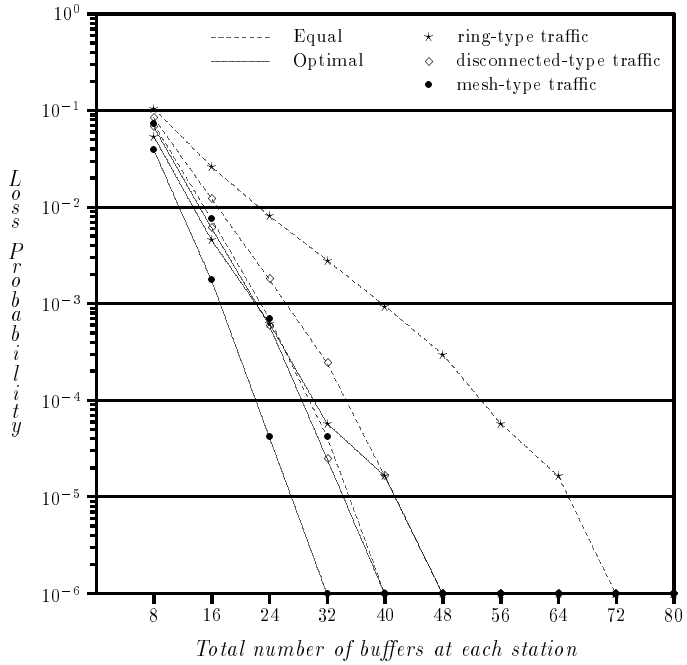


Figure 9: 8-station loss probability curves ($C = 4, \sigma = 0.3$)

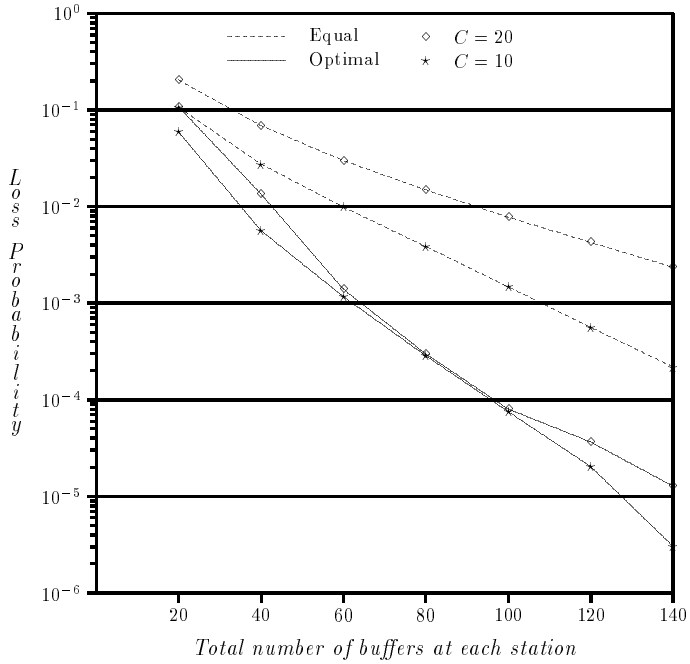


Figure 10: 20-station loss probability curves ($C = 10, 20, \sigma = 0.3$)

References

[1] M. Azizoglu, R. A. Barry, and A. Mokhtar. The effects of tuning time in banwidth-limited optical broadcast networks. In *Proceedings of INFOCOM '95*, pages 138–145. IEEE, April 1995.

[2] I. Baldine and G. N. Rouskas. Reconfiguration in rapidly tunable transmitter, slowly tunable receiver single-hop wdm networks. Technical Report TR-96-10, North Carolina State University, Raleigh, NC, 1996. (Submitted to Infocom '97.).

[3] K. Bogineni, K. M. Sivalingam, and P. W. Dowd. Low-complexity multiple access protocols for wavelength-division multiplexed photonic networks. *IEEE Journal on Selected Areas in Communications*, 11(4):590–604, May 1993.

[4] M. S. Borella and B. Mukherjee. Efficient scheduling of nonuniform packet traffic in a WDM/TDM local lightwave network with arbitrary transceiver tuning latencies. In *Proceedings of INFOCOM '95*, pages 129–136. IEEE, April 1995.

[5] Mon-Song Chen, N. R. Dono, and R. Ramaswami. A media-access protocol for packet-switched wavelength division multiaccess metropolitan area networks. *IEEE Journal on Selected Areas in Communications*, 8(6):1048–1057, August 1990.

[6] R. Chipalkatti, Z. Zhang, and A. S. Acampora. Protocols for optical star-coupler network using WDM: Performance and complexity study. *IEEE Journal on Selected Areas in Communications*, 11(4):579–589, May 1993.

[7] I. Chlamtac and A. Ganz. Channel allocation protocols in frequency-time controlled high speed networks. *IEEE Transactions on Communications*, 36(4):430–440, April 1988.

[8] E. Coffman, M. R. Garey, and D. S. Johnson. An application of bin-packing to multi-processor scheduling. *SIAM Journal of Computing*, 7:1–17, Feb 1978.

[9] P. W. Dowd. Random access protocols for high speed interprocessor communication based on an optical passive star topology. *Journal of Lightwave Technology*, LT-9:799–808, June 1991.

[10] A. Ganz and Y. Gao. Time-wavelength assignment algorithms for high performance WDM star based networks. *IEEE Transactions on Communications*, 42(4):1827–1836, April 1994.

[11] M. R. Garey, R. L. Graham, and D. S. Johnson. Performance guarantees for scheduling algorithms. *Operations Research*, 26:3–21, Jan 1978.

[12] P. E. Green. *Fiber Optic Networks*. Prentice-Hall, Englewood Cliffs, New Jersey, 1993.

[13] I. M. I. Habbab, M. Kavehrad, and C.-E. W. Sundberg. Protocols for very high-speed optical fiber local area networks using a passive star topology. *Journal of Lightwave Technology*, LT-5(12):1782–1793, December 1987.

[14] P. S. Henry. High-capacity lightwave local area networks. *IEEE Communication Magazine*, pages 20–26, October 1989.

[15] M. Hofri and Z. Rosberg. Packet delay under the golden ratio weighted TDM policy in a multiple-access channel. *IEEE Transactions on Information Theory*, IT-33(3):341–349, May 1987.

[16] P. A. Humblet, R. Ramaswami, and K. N. Sivarajan. An efficient communication protocol for high-speed packet-switched multichannel networks. *IEEE Journal on Selected Areas in Communications*, 11(4):568–578, May 1993.

[17] L. Kleinrock. *Queueing Systems, Volume 1: Theory*. John Wiley & Sons, New York, 1975.

[18] J-F. P. Labourdette, F. W. Hart, and A. S. Acampora. Branch-exchange sequences for reconfiguration of lightwave networks. *IEEE Transactions on Communications*, 42(10):2822–2832, October 1994.

[19] N. Mehravari. Performance and protocol improvements for very high-speed optical fiber local area networks using a passive star topology. *Journal of Lightwave Technology*, 8(4):520–530, April 1990.

[20] B. Mukherjee. WDM-Based local lightwave networks Part I: Single-hop systems. *IEEE Network Magazine*, pages 12–27, May 1992.

[21] G. R. Pieris and G. H. Sasaki. Scheduling transmissions in WDM broadcast-and-select networks. *IEEE/ACM Transactions on Networking*, 2(2):105–110, April 1994.

[22] G. N. Rouskas. *Single-Hop Lightwave WDM Networks and Applications to Distributed Computing*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, May 1994.

[23] G. N. Rouskas and M. H. Ammar. Multi-destination communication over single-hop lightwave WDM networks. In *Proceedings of INFOCOM '94*, pages 1520–1527. IEEE, June 1994.

[24] G. N. Rouskas and M. H. Ammar. Analysis and optimization of transmission schedules for single-hop WDM networks. *IEEE/ACM Transactions on Networking*, 3(2):211–221, April 1995.

[25] G. N. Rouskas and V. Sivaraman. On the design of optimal TDM schedules for broadcast WDM networks with arbitrary transceiver tuning latencies. In *Proceedings of INFOCOM '96*, pages 1217–1224. IEEE, March 1996.

[26] G. Semaan and P. Humblet. Timing and dispersion in WDM optical star networks. In *Proceedings of INFOCOM '93*, pages 573–577. IEEE, 1993.