Multi-Destination Communication Over Single-Hop Lightwave WDM Networks

George N. Rouskas   Mostafa H. Ammar

GIT–CC–93/52

August 1993

Abstract

The importance of computer applications and telecommunication services relying on efficient multi-destination communication is growing rapidly. It is now likely that a significant portion of the overall traffic in future communication environments will be of the multi-destination type. At the same time, lightwave technology is emerging as a promising candidate for implementing the next generation of multiuser high-speed networks. In this paper we address the open issue of providing efficient mechanisms for multi-destination communication over one class of lightwave WDM architectures, namely, single-hop networks. We suggest, analyze, and optimize several alternative approaches for broadcast/multicast. One of our major contributions is the development of a suite of adaptive multicast protocols which have superior performance, are very simple to implement, and are insensitive to propagation delays.

College of Computing
Georgia Institute of Technology
Atlanta, Georgia 30332–0280

1 Introduction

The advent of computer applications and telecommunication services requiring multi-destination communication is changing the nature of data traffic carried by the communication networks considerably. It is now likely that a significant portion of the overall traffic in future communication environments will be of the multi-destination type. This reflects the growing importance of applications such as distributed data processing [1], broadcast information systems [2], and teleconferencing, among others, which represent the driving forces behind the development of high-speed networks. It is, therefore, important that next-generation networks employ efficient broadcast/multicast mechanisms [3].

Lightwave technology has emerged as a promising candidate for implementing this next generation of multiuser high-speed communication networks. A wide variety of novel design approaches and network architectures has been proposed in the literature to exploit the unique properties of the optical fiber, and especially its enormous information-carrying capacity [4, 5]. Wave Division Multiplexing (WDM) networks employ multiple, concurrent transmission channels and may deliver an aggregate throughput that can grow with the number of wavelengths deployed, and can be in the order of Terabits per second [6]. Our concern in this paper is with one class of the proposed architectures, namely, single-hop networks.

Single-hop networks [7, 8, 9, 10, 11, 12, 13] are all-optical in nature, i.e., a packet is sent from the source to the destination in one hop, without passing through intermediate stations. For a successful packet transmission, one of the transmitters of the source and one of the receivers of the destination must operate on the same wavelength. Since the number of transceivers per station is usually much smaller than the number of stations or wavelengths in the network, single-hop networks employ tunable transmitters and/or receivers to provide full connectivity among the stations. Because of the need to dynamically tune the transceivers, some form of coordination among the stations of a single-hop network is required. We focus on a wavelength-time assignment of the optical bandwidth, whereby time is slotted and each station may transmit only in slots specified by a predetermined schedule. This is an extension of TDMA over a multichannel environment.

For *single-destination* traffic, we have defined all possible types of schedules, and we have developed a general framework for analyzing and optimizing their throughput for any number of available wavelengths, any transceiver tunability characteristics, and non-uniform traffic [13]. The issue of *multi-destination* traffic has not, however, been adequately addressed in this environment. One important reason may have been the lack of understanding of complicating issues such as the variety of transceiver tunability characteristics and the domination of propagation delays. In this paper we attempt to fill this gap by suggesting, analyzing, and optimizing several alternative approaches to performing efficient broadcast/multicast over single-hop lightwave networks.

The rest of the paper is organized as follows. Section 2 describes the network model, and Section 3 presents schedules optimal for multi-destination traffic only. In Sections 4 and 5 we address the problem of obtaining schedules suitable for mixed (single- and multi-destination) traffic. In Section 6 we develop three adaptive multicast protocols, and present some numerical results in Section 7. Section 8 concludes the paper and points to future research directions.

2 System Model

We consider a network of N stations interconnected through a passive, broadcast optical medium that can support C wavelengths, $\lambda_1, \lambda_2, \ldots, \lambda_C$. In general, $C \leq N$. Each station is equipped with one receiver and one transmitter. The properties of the network depend on whether the transmitters only, or the receivers only are tunable. Following the terminology in [4] we refer to the resulting types of systems as TT-FR and FT-TR, respectively [1]. If the receivers (transmitters) are fixed, wavelength $\lambda(i) \in \{\lambda_1, \ldots, \lambda_C\}$ is assigned to the receiver (transmitter) of station i. The tunable transmitters

[1] A network in which both transmitters and receivers are tunable (TT-TR) can be operated so that either the transmitters or the receivers are always tuned to a fixed wavelength. Therefore, all results obtained for either TT-FR or FT-TR are applicable to TT-TR systems as well.

(receivers), on the other hand, consist of lasers (optical filters) tunable over a range of wavelengths which includes all $\lambda_c, c = 1, \ldots, C$.

We distinguish between single- and multi-destination packets; the latter need to be delivered to a number of stations, members of a *multicast group* which can be different for different packets. The multicast group of a packet originating at station i can be any subset of $\{1, \ldots, N\} - \{i\}$ with cardinality greater than one [2]. In the following, the term "broadcast traffic" will refer to the situation where the multicast group of all multi-destination packets of station $i, i = 1 \ldots, N$, is equal to $\{1, \ldots, N\} - \{i\}$; we will use the term "multicast traffic" otherwise.

The network operates in a slotted mode, with a slot time equal to the packet transmission time plus the tuning time. All stations are synchronized to the slot boundaries [14]. We define σ_i and ρ_i as the probability that a new single-destination and multi-destination packet, respectively, arrives at station i during a slot time. p_{ij} is the probability that an arriving single-destination packet is destined to station j, and $\sum_j p_{ij} = 1$ [3]. Each station has N buffers, one for storing packets destined to each of the $N - 1$ possible destinations, and one for storing multi-destination packets. Each buffer can hold one packet; packets arriving to a full buffer are lost. This is an extension of our model in [13], which in turn is an extension of the single-channel model in [15].

We assume single hop communication. Coordination between the transmitting and receiving stations is achieved by using a predefined, wavelength-time oriented schedule that works as follows. Time slots are grouped in frames of M slots. Within a frame, a_{ij} slots are assigned for packet transmissions between the source-destination pair (i, j). A schedule indicates, for all i and j, which slots during a frame can be used for transmissions from i to j; it can be described by variables $\delta_{ij}^{(t)}, t = 1, \ldots, M$, called *permissions*, and defined as

$$\delta_{ij}^{(t)} = \begin{cases} 1, & \text{if station } i \text{ has permission to transmit to station } j \text{ in slot } t \\ 0, & \text{otherwise} \end{cases} \qquad (1)$$

Obviously, $a_{ij} = \sum_{t=1}^{M} \delta_{ij}^{(t)}$. To ensure fairness, all the schedules we consider have the property that if the traffic originating at i and terminating at j is nonzero then at least one slot per frame is assigned for transmissions from i to j. Formally,

$$\forall\ i, j \quad \sigma_i p_{ij} > 0 \ \vee \ \rho_i > 0 \ \Rightarrow \ a_{ij} \geq 1 \quad \text{(Fairness Condition)} \qquad (2)$$

Whenever $C < N$, in a TT-FR (FT-TR) system a number of receivers (transmitters) have to be assigned the same wavelength, and share a single channel $\lambda_c, c = 1, \ldots, C$. We let R_c and X_c, subsets of $\{1, \ldots, N\}$, denote the set of receivers and transmitters, respectively, sharing channel λ_c,

$$R_c = \{j \mid \lambda(j) = \lambda_c\} \ \forall c, \quad \text{and} \quad X_c = \{i \mid \lambda(i) = \lambda_c\} \ \forall c \qquad (3)$$

2.1 Transmission Modes

The permissions, $\delta_{ij}^{(t)}$, given to the various source-destination pairs together with the transceiver tunability characteristics specify whether collisions, destination conflicts, or both are possible under a certain schedule. A collision occurs when two or more transmitters access the same channel during a slot. On the other hand, networks with tunable receivers may experience destination conflicts if multiple stations are permitted to transmit to the same destination on different channels within a given slot.

In [13] several *transmission modes* were introduced based on the transmissions allowed within a slot. Since only single-destination traffic was considered there, the following property is common to all these transmission modes regardless of the permissions given in a slot: in the absence of collisions and destination conflicts a packet transmitted by i to j in a slot, t, such that $\delta_{ij}^{(t)} = 1$, is received only by j. These transmission modes can be used for transmitting multi-destination packets; however, the packet

[2] Multicasts to groups of size 1 are treated as single-destination packets. Therefore, we assume that the number of recipients of a multi-destination packet is always ≥ 2.

[3] Thus, $\sigma_i p_{ij}$ is the probability that a packet for j arrives at i within a slot.

has to be transmitted by its source multiple times, until it is successfully received by all destinations in its multicast group. We refer to this as a *multi-packet* multicast (or broadcast); when only the transmitters are tunable (TT-FR networks) this is the only way to transmit multi-destination packets. On the other hand, *whenever the receivers are tunable* (FT-TR networks), a single transmission by a source may, if the permissions are appropriately set, reach multiple destinations simultaneously. This is not possible when only transmitters are tunable, and is referred to as a *single-packet* multicast. In this case, the packet is carrying a *multicast address* as its destination address. The multicast address is different than any of the addresses of individual stations and identifies the packet's multicast group. We assume that all stations, upon receiving the packet must at least be able to determine whether they are part of the multicast group or not; in the latter case, they discard the packet.

The following are the transmission modes we will consider in this paper, defined with respect to a slot t [4].

One-to-One mode. The one-to-one mode is such that in slot t (a) exactly C permissions are given to different source-destination pairs, one per channel, (b) no transmitter is given more than one permission, (c) no two stations may transmit a packet to the same destination, and (d) no two transmitters given permission to transmit may access the same channel. In terms of $\delta_{ij}^{(t)}$,

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\delta_{ij}^{(t)} = C;\ \sum_{j=1}^{N}\delta_{ij}^{(t)} \leq 1\ \forall\ i;\ \ \sum_{i=1}^{N}\delta_{ij}^{(t)} \leq 1\ \forall\ j;\ \ \begin{cases} \sum_{i\in X_c}\sum_{j=1}^{N}\delta_{ij}^{(t)} = 1, \text{FT-TR network} \\ \sum_{j\in R_c}\sum_{i=1}^{N}\delta_{ij}^{(t)} = 1, \text{TT-FR network} \end{cases} \tag{4}$$

Broadcast mode. (Only for FT-TR networks.) One station, i, the *owner* of slot t, is given permission to transmit to all stations (which have to tune their receivers to $\lambda(i)$, the transmit wavelength of i, in that slot). No other station has permission to transmit in slot t:

$$\forall\ j:\ \delta_{kj}^{(t)} = \begin{cases} 1, & k = i \\ 0, & k \neq i \end{cases} \tag{5}$$

Multicast mode. (Only for FT-TR networks.) One station, i, the owner of slot t, may transmit to a multicast group g. Other stations may transmit to destinations not in the multicast group, in one-to-one mode, and in channels other than $\lambda(i)$:

$$\delta_{ij}^{(t)} = \begin{cases} 1, & j \in g \\ 0, & j \notin g \end{cases}\ ;\quad \forall\ k \neq i:\ \delta_{ij}^{(t)} = \begin{cases} 0, & j \in g \\ \text{see (4) with } C = C-1, & j \notin g \end{cases} \tag{6}$$

The one-to-one mode is applicable to both TT-FR and FT-TR networks. The broadcast and multicast modes are defined in order to exploit the inherent broadcast capability of FT-TR systems and are not meaningful for TT-FR networks. The multicast mode is the most general mode; it reduces to the broadcast and one-to-one modes when the size of the multicast group is $N-1$ and 1, respectively. A schedule will be called one-to-one, broadcast, or multicast if it consists entirely of one-to-one, broadcast, or multicast slots, respectively.

Figure 1 demonstrates the different transmission modes defined in this section for a FT-TR network with $N = 4$ stations and $C = 2$ wavelengths. Channel λ_1 is shared by the fixed transmitters of stations 1 and 3 ($X_1 = \{1, 3\}$), while channel λ_2 is shared by the transmitters of stations 2 and 4 ($X_2 = \{2, 4\}$). The general one-to-one schedule in Figure 1(a) is such that exactly one of the transmitters in $X_i, i = 1, 2$, has permission to transmit in any given slot, and each station may receive from at most one source in a slot. The cyclic one-to-one schedule of Figure 1(b) is similar, except that each station may transmit to each possible destination exactly once per frame (whereas in Figure 1(a) stations 1 and 2 may transmit to 2 and 1, respectively, twice within a frame). The schedule in Figure 1(c) has broadcast slots, whereby a single transmission by the owner of a broadcast slot reaches all other stations. Figure 1(d) shows a cyclic broadcast schedule where each station is the owner of exactly one broadcast slot per frame. The multicast schedule of Figure 1(e) is such that in each slot

[4] Of the transmission modes introduced in [13] only the one-to-one mode will be considered here, mainly because it involves no packet loss due to collisions or destination conflicts. However, all our results can be easily generalized to include other modes. This may be desirable under certain single-destination traffic parameters for which other transmission modes have been shown to have better performance [13].

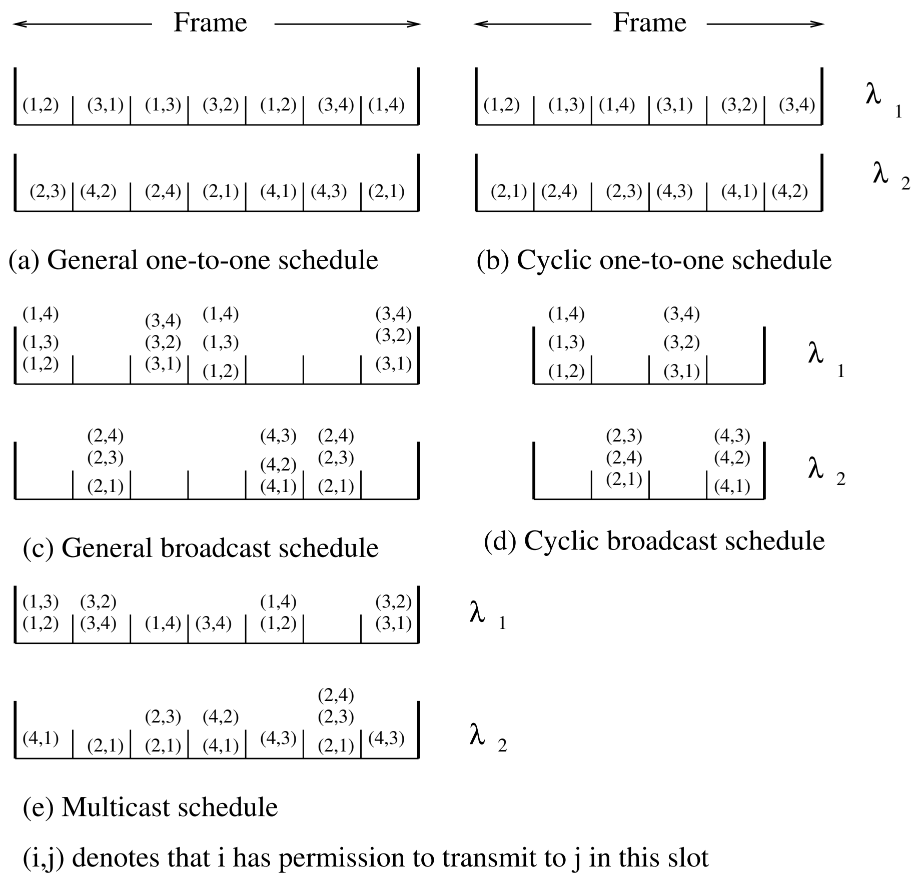

Figure 1: (a)-(b) One-to-one, (c)-(d) Broadcast, and (e) Multicast transmission modes for a FT-TR network with $N = 4$ stations and $C = 2$ wavelengths, $X_1 = \{1,3\}, X_2 = \{2,4\}$

one station (the owner) gets to transmit to a multicast group, while one of the transmitters not sharing the same wavelength with the owner may transmit to a station not in the multicast group.

2.2 Throughput Performance

We will be concerned with evaluating the performance of the various schedules in terms of throughput. This is defined as the expected number of successful packet *receptions* per slot. For one-to-one schedules, the number of packets transmitted is equal to the number of packets received. On the other hand, the number of packets received after a successful transmission by the owner of a multicast or broadcast slot will be equal to the size of the multicast group of the packet transmitted in that slot or $N-1$, respectively. Note that, the maximum number of packets that can be transmitted (and received) in one-to-one slots is C, equal to the number of available channels, while as many as $N-1$ packet receptions are possible in a broadcast or multicast slot. As will be seen later, schedules with broadcast slots tend to perform better than one-to-one schedules, especially when $C \ll N$.

3 Optimal Transmission Schedules for Multi-Destination Traffic Only[5]

Assuming that the traffic offered to the network is of the multi-destination type only ($\sigma_i = 0 \ \forall \ i$), and is described by ρ_i, the probability that a multi-destination packet arrives at station i in a slot, two types of schedules that can be used are one-to-one (for TT-FR networks) and broadcast schedules (for FT-TR networks). In the following two subsections we address the problem of obtaining optimal one-to-one and broadcast schedules for multi-destination traffic only; we defer the discussion on multicast schedules for a later section.

3.1 Optimal Broadcast Schedules

Let b_i be the number of broadcast slots within a frame (of length M) of which i is the owner (i.e., the only station which can use these slots to transmit). Let $d_{i,b}^{(k)}, k = 1, \ldots, b_i$, denote the distance, in slots, between the beginning of the $(k-1)$-th such slot (or the b_i-th slot of the previous frame, if $k = 1$) and the beginning of the k-th slot. i will have a multi-destination packet to transmit in the k-th slot, $k = 1, \ldots, b_i$, in a frame, if at least one packet arrived in the $d_{i,b}^{(k)}$ slots. The average number of multi-destination packets *transmitted* by i per slot is given by

$$T_{i,b} \quad = \quad \frac{1}{M} \sum_{k=1}^{b_i} 1 - (1-\rho_i)^{d_{i,b}^{(k)}} \tag{7}$$

A transmission of a multi-destination packet results in a number of packet receptions equal to the size of the multicast group; in terms of throughput, then, only the size of a group is important. Let $\bar{\eta}$ be the average size of a multicast group. For broadcast traffic only, $\bar{\eta} = N - 1$, while if all multicast groups of sizes $2, \ldots, m, m \leq N-1$, are equally probable, $\bar{\eta} = (m+2)/2$. The network throughput is

$$T \quad = \quad \sum_{i=1}^{N} \bar{\eta} \, T_{i,b} \tag{8}$$

Recall that only one station is allowed to transmit in a slot of a broadcast schedule. Given $\rho_i, i = 1, \ldots, N$, the problem of obtaining an optimal broadcast schedule is then equivalent to the single-channel problem in [15]. There, it was shown that the percentage of time, x_i, that station i should be given permission to transmit is:

$$x_i \quad = \quad \frac{\ln(1-\rho_i)}{\sum_{j=1}^{N} \ln(1-\rho_j)} \tag{9}$$

x_i is independent of the frame length M. Given M, a Fibonacci number greater than or equal to N [15], we assign b_i broadcast slots to station i such that

$$\lfloor M x_i \rfloor \leq b_i \leq \lceil M x_i \rceil \quad \forall \ i \quad \text{and} \quad \sum_{i=1}^{N} b_i \ = \ M \tag{10}$$

We then use the golden-ratio policy, also developed in [15], to place the b_i slots, $i = 1, \ldots, N$, within the frame. As an example, for a network with $N = 4, C = 2$, and $\rho_1 = \rho_2 = \rho_3 = 0.19, \rho_4 = 0.1$, we get $x_1 = x_2 = x_3 = 2/7, x_4 = 1/7$, and the optimal schedule is as in Figure 1(c). If $\rho_i = \rho \ \forall \ i$, the optimal broadcast schedule is a cyclic one, as in Figure 1(d).

[5] Analysis and optimization of schedules for single-destination traffic only ($\rho_i = 0 \ \forall \ i$) is presented in [13].

3.2 Optimal One-to-One Schedules

Consider a multi-destination packet arriving at station i and finding the multi-destination packet buffer empty (and, thus, it is accepted). Let g be the multicast group to which the packet is addressed; g is a subset of $\{1, \ldots, N\} - \{i\}$ with $\mid g \mid \geq 2$. The packet is said to be *eligible* for transmission in a slot, t, in which i may transmit to j (i.e., $\delta_{ij}^{(t)} = 1$), if $j \in g$ and the packet has not yet been transmitted to j. The packet is *new* if it has not yet been transmitted to any destination in g, otherwise it is *old*.

The time between the arrival of a new multi-destination packet and the time it is transmitted to all the stations in its multicast group, g, is called the *multicast cycle* (or *broadcast cycle* in the special case of $\mid g \mid = N - 1$). In general, the length of the multicast cycle experienced by a packet depends on (a) the particular stations in its multicast group, (b) the one-to-one schedule under consideration, and the permissions given to different source-destination pairs, and (c) the slot within a frame in which the packet arrives (which determines in what order the different permissions will be encountered). The packet remains in the buffer for the duration of its multicast cycle; it is removed as soon as it is transmitted to all its destinations. Since multi-destination packets arriving while a packet is occupying the buffer are lost the throughput of multi-destination traffic is in inverse proportion relation with the average multicast cycle length.

The problem of determining one-to-one schedules optimal for multi-destination traffic only can be reduced to the problem of obtaining optimal broadcast schedules discussed in the previous subsection, provided that all stations are equally probable to belong to a multicast group. Under this assumption, a broadcast slot with i as its owner can be thought of as being "equivalent" to $N - 1$ one-to-one slots, $t_1, \ldots, t_{N-1}$, such that $\forall\ j \neq i,\ \exists\ k : \delta_{ij}^{(t_k)} = 1$. This is because i needs exactly $N - 1$ such slots to broadcast a packet in a one-to-one schedule [6]. Unlike broadcast schedules, however, we must also take into account the fact that up to C broadcasts may proceed concurrently, one on each different channel.

Without loss of generality, consider a FT-TR network. Since $C \leq N$, a channel $\lambda_c, c = 1, \ldots, C$, may be shared by a set of transmitters, X_c. In general, stations should share channels so that some load balancing be achieved, and, unless $\rho_i = \rho\ \forall\ i$, each channel may be shared by a different number of transmitters. Given ρ_i, the techniques developed in [13] can help us determine how transmitters should share the available channels, i.e., how X_c should be constructed. Given X_c, we may then solve the single-channel problem [15] for each channel $\lambda_c, c = 1 \ldots, C$, to obtain x_i (note the difference between (9) and (11)):

$$\forall\ i \in X_c :\ x_i \ = \ \frac{\ln(1 - \rho_i)}{\sum_{j \in X_c} \ln(1 - \rho_j)} \quad , c = 1, \ldots, C \tag{11}$$

Once x_i have been found and given M, we can obtain the number of "equivalent" broadcast slots from

$$\forall\ i \in X_c :\ \lfloor M x_i \rfloor \leq b_i \leq \lceil M x_i \rceil \quad \forall\ i \quad \text{and} \quad \sum_{i \in X_c} b_i \ = \ M \tag{12}$$

We can then construct an optimal one-to-one schedule of frame length $M(N - 1)$ by replacing each of i's b_i "equivalent" broadcast slots with $N - 1$ one-to-one broadcast slots as above. In the special case $C = N$ each station would be assigned a unique transmit wavelength, $x_i = 1\ \forall\ i$ from (11), and the optimal one-to-one schedule is a cyclic one with a frame length of $N - 1$. Also, whenever $\rho_i = \rho\ \forall\ i$, but $C < N$, the optimal schedule is a cyclic one as in Figure 1(b).

4 Transmission Schedules for Mixed Traffic

Schedules optimized for one class of traffic only will obviously have the best performance as long as only this class of traffic is offered to the network. A realistic network, however, will have to accommodate a mix of both single-destination and multi-destination traffic. We now describe two types of schedules,

[6] This "equivalency" holds even under the more general assumption of multicast (not broadcast) traffic, and is the analog of assigning broadcast slots regardless of the average multicast group size (see the previous subsection).

namely, one-to-one schedules and schedules with both one-to-one and broadcast slots, and explain how they can be used to transmit both single-destination and multi-destination packets.

4.1 Schedules with One-to-One and Broadcast Slots

Schedules with one-to-one and broadcast slots are useful only for networks with tunable receivers (FT-TR); they can be obtained by merging one-to-one and broadcast schedules similar to those in Figures 1(a)-(d) (more on schedule merging later). One-to-one slots are used exclusively for single-destination packets, while broadcast slots are used solely for broadcasting multi-destination packets to the network. A multi-destination packet is transmitted by the source only once, and is delivered to all stations in its multicast group within a time period equal to the maximum one-way propagation delay.

For a given schedule, the throughput of single-destination traffic is independent of the amount of multi-destination traffic, and vice versa. Let M denote the number of slots per frame, including one-to-one and broadcast slots. Let a_{ij} be the number of one-to-one slots within a frame in which i may transmit to j, and define $d_{ij}^{(k)}$ in a way similar to $d_{i,b}^{(k)}$ (Section 3.1). Using the same reasoning as that used in the derivation of (7) the throughput of traffic originating at i and destined to j is

$$T_{ij} \quad = \quad \frac{1}{M} \sum_{k=1}^{a_{ij}} 1 - (1 - \sigma_i p_{ij})^{d_{ij}^{(k)}} \tag{13}$$

If $\bar{\eta}$ is the average size of a multicast group, the aggregate network throughput can be computed by (13), (7), and

$$T_{total} \quad = \quad T_{multi} \; + \; T_{single} \quad = \quad \sum_{i=1}^{N} \bar{\eta} \; T_{i,b} \; + \; \sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij} \tag{14}$$

4.2 Schedules with Only One-to-One Slots

When one-to-one schedules are used to carry mixed traffic (as would be needed in a TT-FR system) both single- and multi-destination packets are transmitted over the same slots. Let t be a slot such that $\delta_{ij}^{(t)} = 1$. Whenever both an eligible multi-destination packet (i.e., a packet with j in its multicast group, but not yet transmitted to j) and a single-destination packet for j are in the buffers of i at the beginning of slot t we say that a *transmission contention* arises. The rules to resolve the contention, and thus decide which packet will be transmitted, constitute a *contention-resolution policy*. The contention-resolution policy used directly affects the throughput of both classes of traffic.

In this paper we will consider and compare the performance of the following three transmission policies, listed in decreasing order of expected multicast cycle length experienced by multi-destination packets [7].

Policy 1. In case of contention, either the multi-destination or the single-destination packet is transmitted with probability 1/2.

Policy 2. If the multi-destination packet involved in the contention is *new*, either that or the single-destination packet is transmitted with probability 1/2. *Old* multi-destination packets, however, have priority over single-destination packets whenever contention arises.

Policy 3. Multi-destination packets always have priority in case of contention.

Let t be the slot in which an arriving multi-destination packet is accepted in the buffer, and t' be the slot in which it is first transmitted. For Policy 3, t' is the first slot after t in which the source has permission to transmit. Under Policies 2 and 3, the packet will be transmitted to all subsequent slots in which it becomes eligible for transmission, regardless of contention with single-destination packets. Because of the Fairness Condition (2), the packet is guaranteed to be transmitted to all stations in the multicast group within M slots, M the frame length, from the beginning of t'. It will be received by

[7] A policy is needed only in the case of contention between a single-destination and an eligible multi-destination packet. If only one packet is in the buffer at the beginning of a slot, it is transmitted regardless of its type.

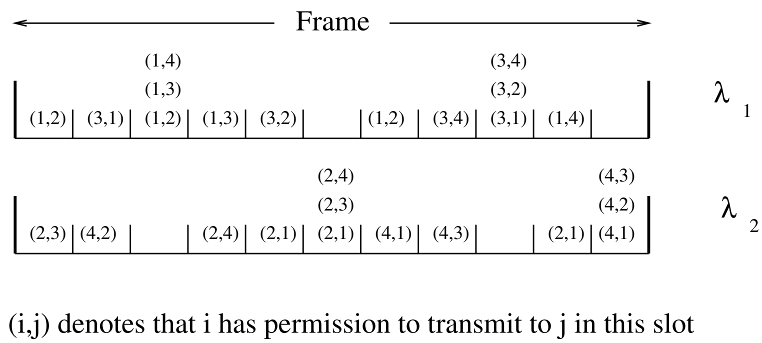

Figure 2: Schedule merging

all stations within a time period of length equal to M slots plus the maximum one-way propagation delay. This ensures the timely delivery of the packet to all of its destinations. No such guarantee is possible under Policy 1 [8].

The throughput of one-to-one schedules under different policies was evaluated using simulation.

5 Schedule Optimization for Mixed Traffic

The problem of obtaining schedules that optimize the throughput of mixed traffic can be stated concisely as

> Given traffic parameters ρ_i, σ_i, and p_{ij}, $i, j = 1, \ldots, N$, the number of available wavelengths, C, and the tunability characteristics of the network (TT-FR or FT-TR), find the optimum schedule.

As a first step, one has to determine the type of schedule and, if applicable, the type of contention-resolution policy to use. From previous experience [13] we expect this to be a very hard allocation problem. In this section we develop an optimization heuristic to construct schedules that not only perform very well, but also guarantee a (user-specified) level of throughput for each class of traffic.

The heuristic is based on the concept of schedule *merging*. Let S_1 and S_2 be two schedules of frame lengths M_1 and M_2, respectively. Without loss of generality, assume that $M_1 \geq M_2$ and $M_1 = mM_2$. If m is an integer, merging of S_1 and S_2 is performed by inserting one slot of S_2 after every m slots of S_1, resulting in a new schedule S, of frame length $M = M_1 + M_2$. Schedule merging can be easily generalized to situations where M_1 is not an integer multiple of M_2, and can be performed in time $O(M_1 + M_2)$. In Figure 2 we show the result of merging the one-to-one schedule of Figure 1(a) ($M_1 = 7$), with the cyclic broadcast schedule of Figure 1(d) ($M_2 = 4$).

Our approach to constructing optimized one-to-one schedules is based on the following observations. Consider a one-to-one schedule, S, optimized for single-destination traffic only, under which multi-destination packets experience long multicast cycle lengths. One way to reduce the multicast cycle is to merge S with a one-to-one schedule, S', optimized for multi-destination traffic, effectively providing more slots in which multi-destination packets may be transmitted. If the amount of multi-destination traffic is high, it might be beneficial to merge S with $l, l > 1$, frames of S' instead of just one, reducing the average multicast cycle length even further. As l increases, the merged schedule will tend to favor multi-destination packets (note that as $l \to \infty$, the resulting schedule will be indistinguishable from an

[8] In an actual implementation, and in order to prevent starvation, the number of times a packet of any type is excluded by either Policy 1 or Policy 2 may be bounded, after which it will be transmitted with probability 1.

S' schedule, in which case the throughput of single-destination traffic may suffer). Therefore, we must choose an l such that the total throughput, T_{total}, is high and the throughput of single-destination traffic, T_{single} is at least α percent of T_{total}. α is specified by the designer, and reflects the relative importance of the two classes of traffic.

We now propose the following Schedule-Merging Heuristic (SMH).

Schedule-Merging Heuristic (SMH) (For constructing One-to-One schedules)

1. Given single-destination traffic parameters σ_i and p_{ij}, and the number of available wavelengths, C, obtain an optimized one-to-one schedule, S_0, as described in [13]. Let M be the number of slots per frame of S_0. Evaluate its throughput, $T_{total}(S_0)$, under mixed traffic.

2. Given ρ_i and C obtain a one-to-one schedule, S_0', optimized for multi-destination traffic only (as in Section 3.2). Let M' be the number of slots of this schedule. Set $l \leftarrow 1$.

3. Produce a new schedule, S_l, of $M + lM'$ slots per frame by merging the M slots of S_0 with l frames of S_0'.

4. If $T_{total}(S_l) > T_{total}(S_{l-1})$ *and* $T_{single}(S_l) \geq \alpha T_{total}(S_l)$ set $l \leftarrow l+1$ and repeat from Step 3. Otherwise, stop; the best schedule to use is S_{l-1}.

In order to construct schedules with one-to-one and broadcast slots (FT-TR networks only), we may use a similar procedure, i.e., merge a one-to-one schedule optimized for single-destination traffic with $l, l \geq 1$, frames of a broadcast schedule optimal for multi-destination traffic. As all $\rho_i \rightarrow 1$, letting $l \rightarrow \infty$ guarantees the highest overall throughput. Again, therefore, we need to select l so as to guarantee a minimum level of performance for single-destination traffic. To this end, we can use SMH modified so that in Step 2 instead of a one-to-one we obtain an optimal broadcast schedule (Section 3.1). Since we must also guarantee that the final schedule will have broadcast slots, the stoppping rule must also be changed to:

4. If $l = 1$ or $T_{total}(S_l) > T_{total}(S_{l-1})$ *and* $T_{single}(S_l) \geq \alpha T_{total}(S_l)$ set $l \leftarrow l+1$ and repeat from Step 3. Otherwise, stop; the best schedule to use is S_{l-1}.

6 Adaptive Multicast Protocols for FT-TR Networks

Allocating broadcast slots for the exclusive transmission of multi-destination packets may be wasteful in terms of throughput, especially when the average size of a multicast group is small compared to the number of stations in the network (a situation that often arises in distributed systems). As an example, consider a network with $N = 100$, $C = 50$, and a packet transmission in a broadcast slot. By definition, no other transmissions can take place in that slot. This is fine, provided that the packet's multicast group includes all 99 possible destinations. But assume that the multicast group consists of only, say, 5 stations. As a result of using broadcast slots, 49 channels are forced to being idle although other packet transmissions can take place in the same slot without interfering with the multi-destination packet's transmission. Using multicast slots instead might improve the situation, as in these slots transmission of single-destination packets to stations not in the multicast group are permitted. Since the multicast groups are usually not known in advance, one solution would be to allocate at least one slot per frame for transmissions to each possible multicast group. The number of these groups, however, explodes with the size of the network, making this approach impractical even for networks of moderate size.

We now present a suite of adaptive multicast protocols for FT-TR networks which assume that each station, i, is the owner of b_i multicast slots per frame. The protocols are adaptive in the sense that the transmissions allowed in these slots are not specified in advance; instead, they are dynamically updated to reflect current traffic loads. A station, i, may transmit to any multicast group in its multicast slots $t_1, \ldots, t_{b_i}$. However, as the multicast group changes, the permissions in each of the multicast slots (see (6)) also change so that the overall throughput is maximized.

6.1 The Basic Idea

The operation of the protocols is based on the assumption that a source, i, will typically transmit L consecutive packets, $L > 1$, to the same multicast group, g. This is true, for example, in the case of bulk arrivals, i.e., when a long message has to be fragmented in a number of fixed-size packets. L need not be constant; we assume that L is equally likely to be any integer in the range $L_{min} \leq L \leq L_{max}$. L_{min} and L_{max} may correspond to the minimum and maximum message size, respectively. We will now describe the basic idea behind the operation of the protocols by considering the transmissions in i's multicast slots; similar observations can be made for other stations' multicast slots.

In the first multicast slot with i as its owner all stations tune their receivers to $\lambda(i)$, the transmit wavelength of i. Let g be the multicast group to which the packet transmitted by i in that slot is addressed, and let L be the total number of packets i will transmit to the same group; $g = \phi$ if no packet is transmitted by i in that slot. Suppose that $\mid g \mid < N - 1$, and consider a station $j \neq i$. If $j \in g$ then j will continue listening to $\lambda(i)$ in subsequent multicast slots of i. However, if $j \notin g$, j is free to tune its receiver to the transmit wavelength of another station, k, in subsequent multicast slots of i. If k has a single-destination packet for j, and provided that $\lambda(k) \neq \lambda(i)$, it can transmit it in i's multicast slots, thus increasing channel utilization.

After i transmits all L packets to the same multicast group g, it will not be able to transmit to a group $g' \neq g$, unless all stations not in g are somehow notified. We therefore require that all stations tune their receivers to $\lambda(i)$ in specified multicast slots of i, called *synchronization* slots (as explained, the first multicast slot is a synchronization slot). The F multicast slots of i between synchronization slots are called *free* as receivers not in g are free to tune to any wavelength other than $\lambda(i)$. F is a network-wide constant and thus, all stations can synchronize by tuning to $\lambda(i)$ in synchronization slots.

Note that i may start transmitting packets to a new multicast group only in a synchronization slot. If F is large, i will, on the average, have to wait for a considerable number of slots to start transmitting to a new group. On the other hand, if F is very small relative to L there will be unnecessarily many synchronization slots in which no transmissions by stations other than i are allowed. F will, in general, be a function of L (i.e., L_{min}, L_{max}), as well as of the propagation delay (more on this later), and must be carefully selected in order to maximize the overall throughput.

We have not yet discussed how a receiver $j \notin g$ selects a transmitter $k \neq i$ to tune to in i's free multicast slots. There are two issues that need to be considered. First, $\lambda(k)$ must be different than $\lambda(i)$ to prevent packet loss due to collisions in free multicast slots. Second, k must also be informed of j's decision. Real-time negotiation between j and other stations to determine k is impractical because of the propagation delays involved.

To solve the first problem we start with a one-to-one schedule, S, of frame length M, and let a_i be the number of slots per frame in which i may transmit under S, $a_i = \sum_{t=1}^{M} \sum_{j=1}^{N} \delta_{ij}^{(t)}$. We then specify $b_i, b_i < a_i$, of these slots as multicast slots with i as their owner. Let t be one of these b_i slots and consider a station $j \neq i$ which, according to the one-to-one schedule S, has to tune its receiver to station k in slot t. If t is a synchronization slot, or if t is a free slot but $j \in g$, j will ignore the permissions specified by S and, in slot t, it will tune to $\lambda(i)$ instead. However, if t is a free multicast slot and $j \notin g$, j will tune to $\lambda(k)$ as S specifies. Note that, since S is a one-to-one schedule and both i and k are given permission to transmit in the same slot t, we have $\lambda(i) \neq \lambda(k)$.

6.2 Determining Group Membership

Since all stations execute the same protocol, the problem of informing k about j's decision is now partially solved: k knows that j will tune to $\lambda(k)$ in slot t if (a) t is a free slot, and (b) $j \notin g$. Deciding about (a) is done by k as part of the protocol for tuning its own receiver. Thus the problem reduces to how k may determine whether j is in the multicast group g or not. We now describe three protocols which differ in their assumptions about k's knowledge regarding membership in the multicast groups of packets originating at station $i \neq k$.

Global-knowledge Multicast Protocol (GMP). k maintains tables to map a multicast address in a packet originating at i into the stations-members of the multicast group. By listening to a synchronization slot of i it can tell whether j is in the multicast group or not. Since k must have similar tables for all i, this protocol may be very expensive in terms of memory requirements, as well as in terms of the communication cost for building and maintaining the tables.

Control-packet Multicast Protocol (CMP). k has no knowledge about the members of multicast groups of packets originating at i [9]. However, before transmitting a packet to a new multicast group, g, i will first transmit, in a synchronization slot, a control packet informing other stations about the members of g. Following the control packet transmission, i will transmit the L packets to g as discussed above. k uses the control packet to associate g with the group members. This protocol incurs the overhead of one extra packet, but this is not expected to be a problem, especially if $L_{min} \gg 1$. In addition, this protocol does not require building and maintaining potentially large global tables at each station.

Probabilistic Multicast Protocol (PMP). k has no way to find out whether j belongs to g or not. It will transmit a packet to j in a free multicast slot of i, if it has one, with probability q. No overhead in terms of memory or control packets is incurred, but the selection of an appropriate value for q is crucial in order to minimize packet loss due to destination conflicts (if $j \in g$, j will tune to $\lambda(i)$ in free multicast slots of i and k's transmissions in these slots will be wasted). In general, q should represent the probability that j will *not* belong to g. If $\bar{\eta}$ is the average number of stations in a multicast group, we set $q = 1 - \frac{\bar{\eta}}{N-1}$.

6.3 Effect of Propagation Delay

Under either GMP or CMP a transmitter $k \neq i$ must receive the packet transmitted by i in a synchronization slot before it can determine whether the stations to which it is scheduled to transmit in the next free multicast slots of i belong to g or not. Figure 3 illustrates how propagation delay may become a problem. In this Figure we show a synchronization slot of i followed by F free slots and an synchronization slot; the horizontal axis represents time increasing from left to right. The transmitters of both i and k are synchronized at the beginning of each slot [14]. But a packet transmitted by i will not be heard by the receiver of k until τ_{ik} slots later, where τ_{ik} is the propagation delay from i to k in slots. In the scenario of Figure 3, by the time k receives the packet transmitted by i in the first synchronization slot, free slot t_1 has already passed by its transmitter. Since at the beginning of t_1, k does not know whether $j \in g$ it may not transmit a packet to it.

As a result of the propagation delays some of the free multicast slots may not be used for single-destination transmissions; the longer the propagation delays the less free slots that may be utilized. In the extreme case when all F free slots are within a propagation delay, neither GMP nor CMP will be able to capitalize on the availability of free slots to improve the throughput. Thus, F is indeed a function of the propagation delay as mentioned earlier. Observe, though, that the propagation delay will have a negative effect only if it increases beyond the number of slots between *consecutive* multicast slots with the same owner. Going back to Figure 3, if τ_{ik}, in slots, is less than the distance between the first synchronization slot and t_1, k will be able to transmit in t_1, as well as in all other free slots (if $j \notin g$). Otherwise, k will still be able to use slots $t_2, \ldots, t_F$ as long as τ_{ik} is further increased by less than the distance between t_1 and t_2, and so on. By assigning multicast slots to i so that they are spaced out in the frame we can make the distance between two consecutive multicast slots much larger than one slot. We then expect GMP and CMP to be largely insensitive to propagation delays.

PMP was devised to overcome this problem. Under this protocol, k does not need to wait until it receives the packet transmitted in the first synchronization slot. Regardless of whether $j \in g$ or not, k will transmit in slot t_1 with probability q, provided that it has a packet for j. Thus, PMP is not affected by propagation delays at all.

The algorithm used by i for transmission in its multicast slot, and the algorithm used by j for

[9] Except, of course, for deciding whether k itself is in the multicast group or not.

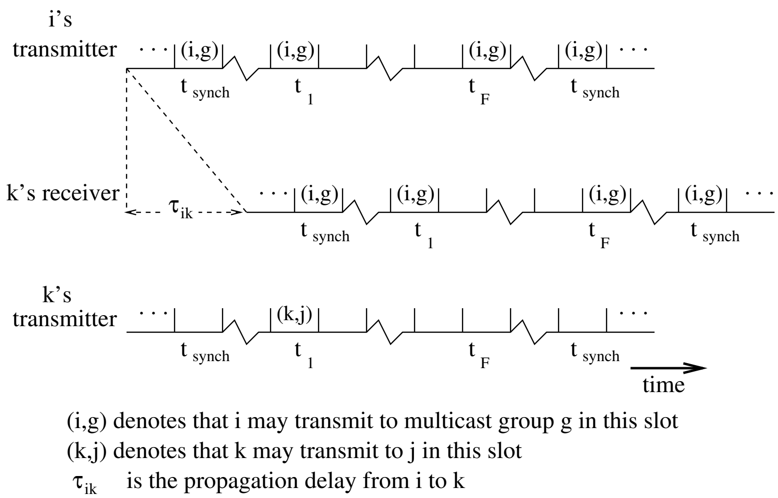

Figure 3: Effect of the propagation delay (not in scale)

tuning its receiver in i's multicast slots are common to all protocols and are shown in Figures 4 and 5, respectively. The algorithm used by k's transmitter, $k \neq i$, for transmissions in i's multicast slots differs depending on the protocol, as seen in Figure 6. The protocols are very simple to implement, and thus suitable for the high-speed environment we are considering.

7 Numerical Results

We consider three 8-station networks with single-destination traffic parameters described by the ring-type, disconnected-type, and quasi-uniform traffic matrices in Figures 7, 8, and 9, respectively. We also consider a 20-station network with a ring-type traffic matrix (not shown here, but similar to the matrix in Figure 7). For the multi-destination traffic parameters we assume, for ease of presentation, that $\rho_i = \rho \ \forall \ i$; since we are interested in the behavior of the various schedules as the amount of multi-destination traffic increases, our conclusions will be valid for the general case, i.e., different ρ_i. All simulation results were obtained with a confidence of 99% in less than 1% variation from the mean.

7.1 Schedules Optimized for One Class of Traffic Only

In this section we investigate how one-to-one schedules optimized for a single class of traffic perform under a mixed traffic scenario. Figure 10 plots the throughput of a one-to-one schedule optimized for the disconnected-type single-destination traffic matrix, as ρ increases from 0 to 1 [10]. In Figure 11 we plot the throughput of a cyclic one-to-one schedule, optimal for multi-destination traffic when $\rho_i = \rho \ \forall \ i$, when the single-destination traffic offered is of the quasi-uniform type. Although in both Figures we have $N = C = 8$ and broadcast multi-destination traffic, the behavior shown is typical of other values for C, as well as for multicast traffic.

As ρ increases, and for all policies, the throughput of the single-destination (broadcast) traffic decreases (increases). This is expected as more broadcast packets contend with single-destination packets for the same slots. The total throughput increases at first, as a result of the increased total offered load, until a saturation point is reached, after which any increase in ρ has a negligible effect on total throughput. Comparing the three contention-resolution policies we note that the best overall and best multi-destination traffic throughput for any value of $\rho > 0$ is achieved by Policies 3 and 2, followed

[10] Values of $\rho \rightarrow 1$ correspond to the situation where stations always have a multi-destination packet to transmit.

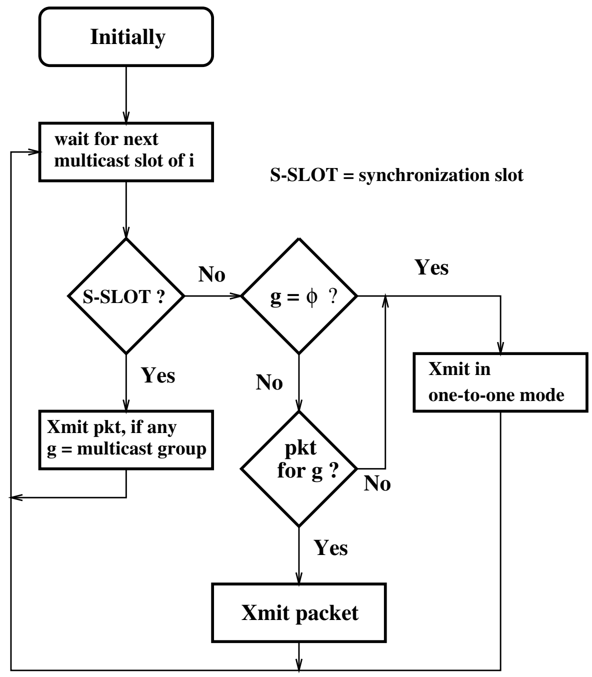

Figure 4: Algorithm executed by i's transmitter for transmission in i's multicast slots.

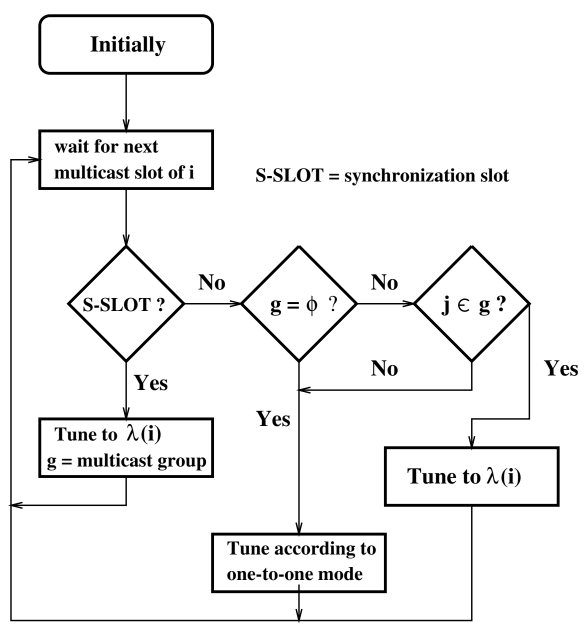

Figure 5: Algorithm executed at j's receiver for tuning in i's multicast slots.

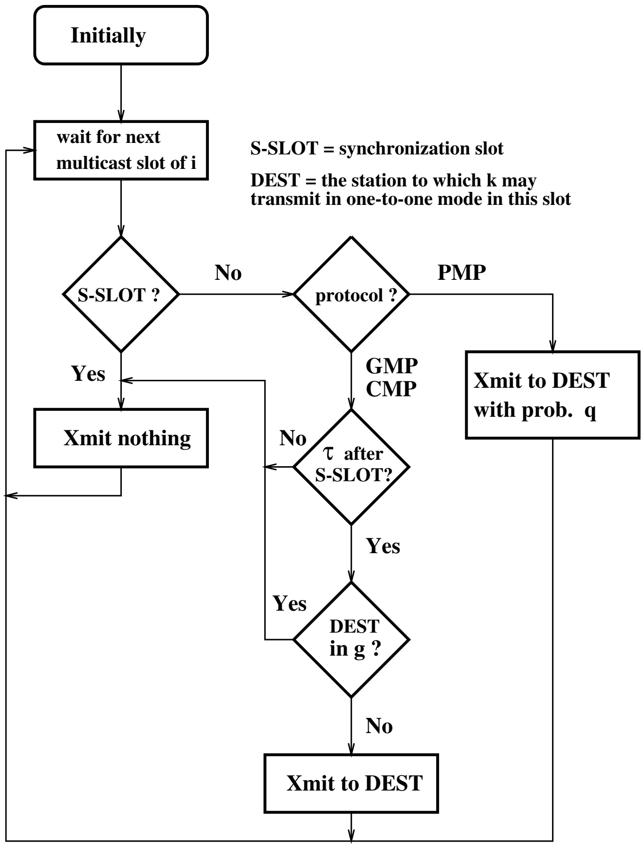

Figure 6: Algorithm executed by k's transmitter for transmission in i's multicast slots ($k \neq i$).

by Policy 1. By giving priority to multi-destination packets, Policies 2 and 3 achieve shorter average multicast cycles, and thus higher multi-destination throughput, at the expense of single-destination traffic.

In all our experiments we have found that schedules optimized for multi-destination traffic only achieve very high total throughput but are extremely unfair to single-destination packets, especially when Policies 2 or 3 are used. Policy 1 achieves a better balance of single- and multi-destination traffic throughput, but it does not guarantee timely delivery of a packet to its destinations. On the other hand, schedules optimized for single-destination traffic only may result in both low overall throughput and starvation of multi-destination packets regardless of the contention-resolution policy used.

7.2 Schedules Optimized for Mixed Traffic

Figure 12 plots the throughput of one-to-one schedules optimized with SMH for broadcast and disconnected-type single-destination traffic. Only Policies 1 and 2 are shown; the curves for Policy 3 are almost identical to those of Policy 2. The value of α, the minimum single-destination throughput as a percentage of total throughput, was set to 45%. Unlike previous Figures, schedules for different values of ρ may be different as they may correspond to different values of l in SMH (recall that l is the number of frames of a schedule optimized for multi-destination traffic merged with one frame of a one-to-one schedule optimized for single-destination traffic).

The curves of Policy 1 are monotonically increasing or decreasing, while those of Policy 2 have discontinuities. This can be explained by considering the operation of SMH, which guarantees that $T_{single}(S_l) \geq \alpha T_{total}(S_l)$. Let l_1, l_2 be the values of l selected by SMH for ρ_1 and ρ_2, respectively, with

0	0.60	0.10	0.05	0.05	0.01	0.03	0.05
0.05	0	0.50	0.02	0.10	0.05	0.04	0.10
0.10	0.05	0	0.50	0.10	0.02	0.05	0.05
0.05	0.10	0.03	0	0.60	0.05	0.10	0.02
0.04	0.02	0.05	0.03	0	0.70	0.05	0.10
0.10	0.15	0.03	0.05	0.04	0	0.45	0.10
0.04	0.05	0.05	0.01	0.10	0.05	0	0.65
0.55	0.10	0.01	0.10	0.05	0.10	0.04	0

Figure 7: Ring-type matrix

0	0.20	0.35	0.25	0.03	0.02	0.01	0.04
0.20	0	0.25	0.40	0.04	0.01	0.03	0.02
0.30	0.25	0	0.35	0.02	0.02	0.01	0.03
0.15	0.40	0.30	0	0.02	0.01	0.02	0.02
0.02	0.04	0.03	0.03	0	0.30	0.35	0.20
0.01	0.03	0.04	0.03	0.30	0	0.25	0.25
0.02	0.01	0.01	0.03	0.40	0.15	0	0.30
0.03	0.03	0.02	0.02	0.25	0.35	0.30	0

Figure 8: Disconnected-type matrix

0	0.25	0.10	0.05	0.25	0.10	0.05	0.15
0.20	0	0.05	0.20	0.05	0.15	0.15	0.20
0.05	0.20	0	0.10	0.15	0.30	0.10	0.05
0.10	0.05	0.25	0	0.10	0.05	0.30	0.10
0.05	0.05	0.25	0.20	0	0.20	0.10	0.15
0.10	0.30	0.05	0.05	0.15	0	0.20	0.15
0.15	0.20	0.15	0.20	0.05	0.15	0	0.05
0.30	0.10	0.10	0.15	0.25	0.05	0.05	0

Figure 9: Quasi-uniform matrix

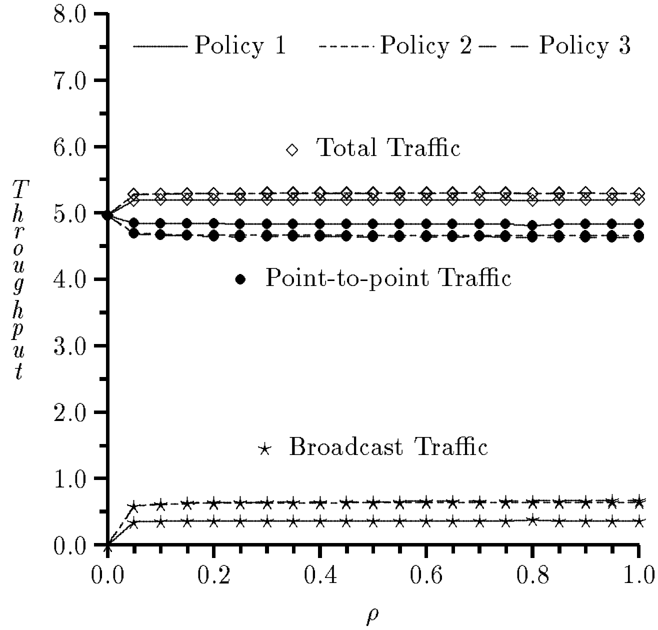

Figure 10: Throughput performance of a one-to-one schedule, $M = 144$, optimized for disconnected-type single-destination traffic ($N = C = 8$).

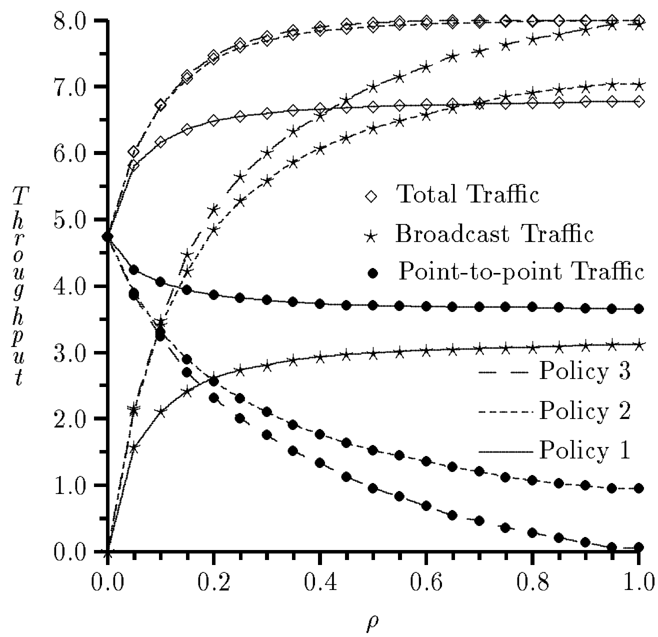

Figure 11: Throughput performance of a cyclic one-to-one schedule with quasi-uniform single-destination traffic ($N = C = 8$).

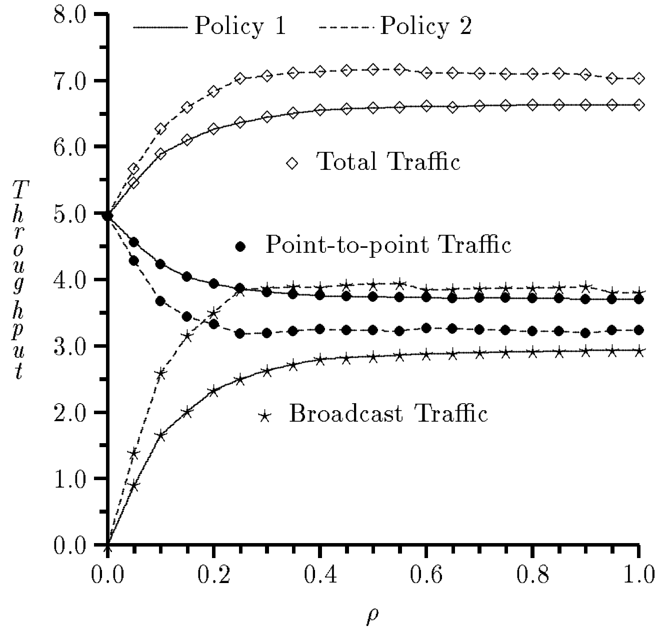

Figure 12: Throughput of one-to-one schedules optimized with SMH, $N = C = 8, \alpha = 0.45$ (disconnected-type single-destination matrix).

$\rho_1 < \rho_2$. In general, $l_1 \leq l_2$, unless the above condition is violated for $\rho = \rho_2, l = l_1$, in which case SMH is forced to select $l_2 < l_1$, effectively decreasing (increasing) the throughput of multi-destination (single-destination) traffic to satisfy the condition. In Figure 12 this situation never arises under Policy 1, but it does arise several times under Policy 2 (e.g., for $\rho_1 = 0.55, \rho_2 = 0.60$).

Comparing Figure 12 to Figure 10 we observe that the schedules produced by SMH have a much higher total throughput: up to 28% increase for Policy 1 and 35% increase for Policy 2. Total throughput is not as high as that achieved by cyclic one-to-one schedules (Figure 11), however, by varying the value of α one can produce a wide range of schedules between (and including) the two extremes shown in Figure 10 (large α) and Figure 11 (small α). We conclude that SMH is very general, giving the designer the flexibility to obtain schedules tailored to the specific needs of applications.

Figures 13 and 14 plot the throughput of schedules with one-to-one and broadcast slots optimized with SMH ($\alpha = 45\%$) for the 20-station ring-type network. Figure 13 is for broadcast traffic while Figure 14 is for multicast traffic, whereby the maximum size of a multicast group is equal to 5 (i.e., a multicast group consists of at least 2, and at most 5 stations; however, all stations are equally likely to be members of a group). In both cases SMH produces (static) schedules with the best possible total throughput. Note that in the case of multicast traffic with small multicast groups (Figure 14), the total throughput for the mixed traffic is *lower* than the throughput of single-destination traffic only ($\rho = 0$), although the offered load is higher (see the stopping rule for this type of schedules in Section 5). This is because, in this case, broadcast slots allocated for transmitting multicast packets are mostly wasted. We show how the adaptive protocols improve the situation shortly.

7.3 Tunable Transmitters vs. Tunable Receivers

Figures 15 and 16 plot the throughput of one-to-one schedules and schedules with one-to-one and broadcast slots, respectively, optimized with SMH ($\alpha = 45\%$), when the number of wavelengths is less than the number of stations $N = 8, C = 4$, and ring-type matrix). The curves for Policy 3 in Figure 15

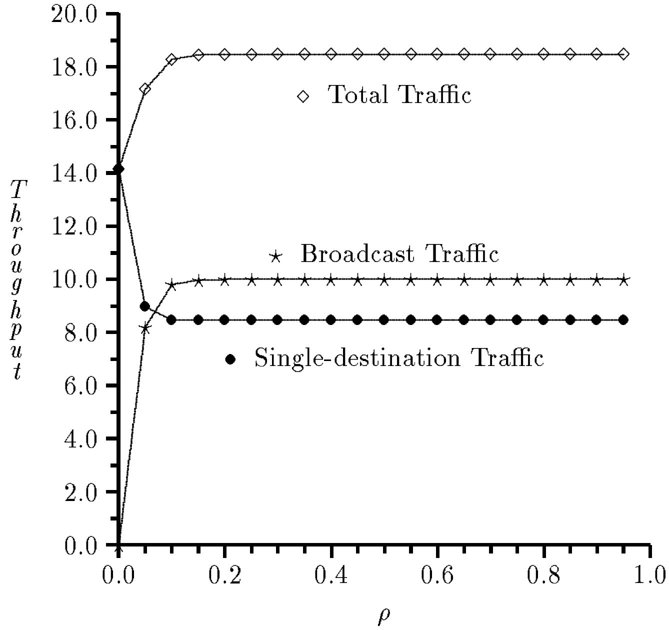

Figure 13: Throughput performance of one-to-one schedules with broadcast slots optimized with SMH under broadcast traffic (ring-type matrix, $N = C = 20, \alpha = 0.45$).

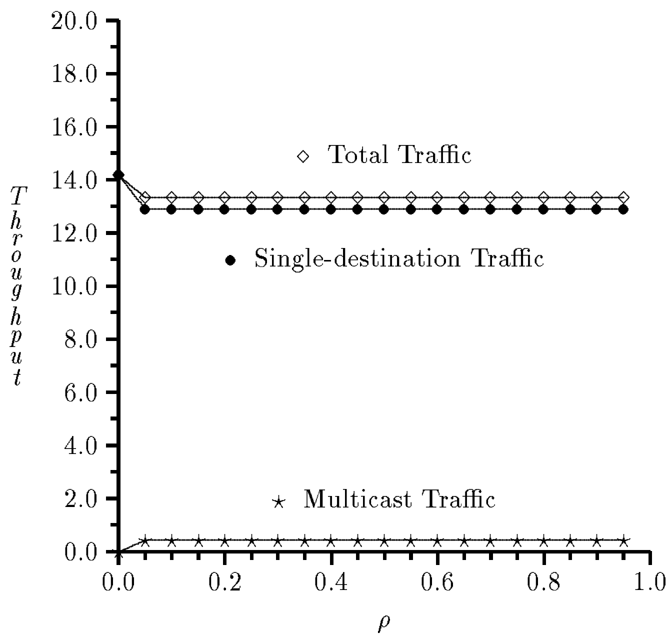

Figure 14: Throughput performance of one-to-one schedules with broadcast slots optimized with SMH under multicast traffic (ring-type matrix, $N = C = 20, \alpha = 0.45$, maximum multicast group size $= 5$).

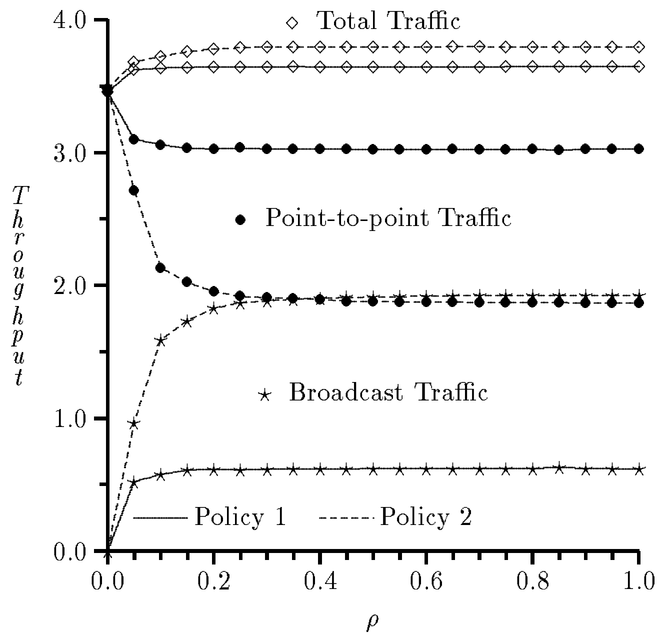

Figure 15: Throughput of one-to-one schedules optimized with SMH, $N = 8, C = 4, \alpha = 0.45$ (ring-type single-destination matrix).

are very close to those of Policy 2. As we can clearly see, schedules with one-to-one and broadcast slots achieve a total throughput higher than the number of channels, 4, which is the maximum a one-to-one schedule can achieve. This behavior is more pronounced when $C = 2$ (not shown here) indicating that, for networks with $C < N$ and mixed traffic, tunable receivers are preferable over tunable transmitters. The non-monotonic behavior of the curves in Figure 16 can be explained as in the previous section.

7.4 Adaptive Protocols

In this section we consider the 20-station network with ring-type single-destination traffic used in the results plotted in Figures 13 and 14, and we assume that the multi-destination traffic offered to the network is such that the maximum size of a multicast group is equal to 5 (thus, the average size of a multicast group, $\bar{\eta}$, is equal to 3.5). We set the values of L_{min} and L_{max}, the minimum and maximum number of consecutive multicast packets to be transmitted to the same multicast group, to 30 and 50, respectively; the number of packets actually transmitted to a certain group is a randomly chosen integer in the interval $L_{min} \leq L \leq L_{max}$. The value of ρ was set to 0.4. We also assume that the propagation delay from i's transmitter to j's receiver, $\tau_{ij} = \tau \; \forall \; i, j$. This can be taken as the worst case scenario, with $\tau = \max_{ij}\{\tau_{ij}\}$.

Figure 17 plots the throughput of GMP as F, the number of free multicast slots between two synchronization slots increases from 5 to 100, for two values for the propagation delay, $\tau = 100$ and $\tau = 200$. As expected, only the throughput of single-destination traffic is affected by the propagation delay; a longer propagation delay means that less free slots can be used to transmit single-destination packets. Also, the greater the value of F the less the frequency of synchronization slots which cannot be used for single-destination transmissions, and the higher the single-destination and total throughput. F also affects the multi-destination traffic throughput. When $F = 5$, a station will, on the average, have to wait for a small number of slots for a synchronization slot in order to transmit to a new multicast group. As F increases beyond $L_{max} = 50$, a station has to wait longer, and the throughput

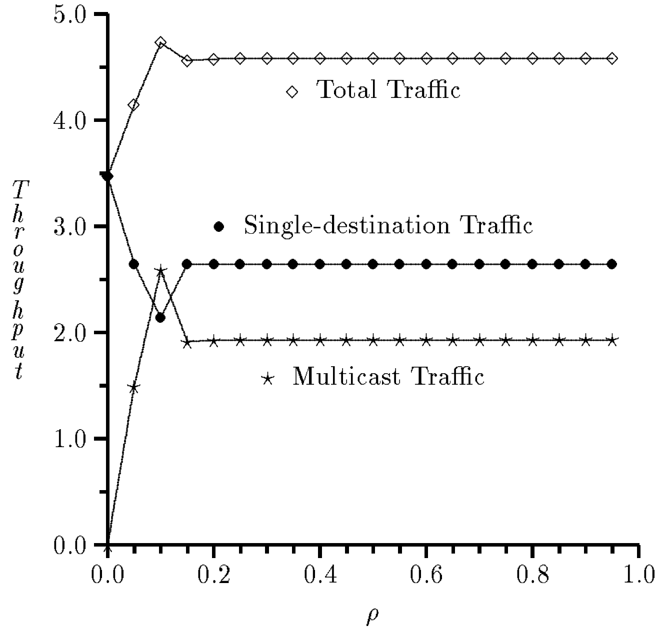

Figure 16: Throughput of schedules with one-to-one and broadcast slots optimized with SMH, $N = 8, C = 4, \alpha = 0.45$ (ring-type single-destination matrix).

decreases. The highest overall throughput is obtained when $F = 50$ for both values of the propagation delay shown.

The throughput of GMP ($F = 50$) as a function of the propagation delay, τ, is plotted in Figure 18, where τ is given in slots. With 500-bit packets, 1Gbps data rates, and speed of light in the fiber $v = 2 \times 10^8$ m/s, the packet transmission time corresponds to the propagation delay along 100m of fiber. The range of propagation delays plotted represent delays over distances up to 20Km, reasonable for LANs/MANs. Note that because of the extra padding needed for synchronization within each slot [14], the slot duration is actually larger than just the time needed to transmit a packet.

Figure 18 confirms our assertion that GMP is, to a large degree, insensitive to propagation delays; single-destination throughput only slightly decreases as propagation delays increase. Comparing to the values of the throughput for $\rho = 0.4$ in Figure 14 (showing the optimized schedules with broadcast slots for identical traffic), we note that the total throughput of GMP, across all propagation delays, is higher. In addition, GMP maintains a better level of throughput for multi-destination traffic. The behavior of CMP (not shown) is very similar, although it incurs slightly lower multi-destination throughput due to the extra control packet.

Figure 19 plots the throughput of PMP as a function of F (recall that PMP is totally insensitive to propagation delays). The highest attainable throughput is somewhat lower than that of GMP and CMP in the range of propagation delays shown in Figure 18. However, the advantage of PMP lies in the fact that it does not incur any overhead (as CMP and GMP do), and that it can be used even for extremely long propagation delays, when the throughput of both CMP and GMP would degrade.

The real advantage of the three protocols, however, lies on their adaptability. Consider what would happen to the throughput of the schedules in Figure 13 if the characteristics of multi-destination traffic changed so that, instead of broadcast packets, packets with a maximum multicast group size of 5 arrived. T_{multi} for $\rho = 0.4$ would drop from 10.0 to 1.84 as a result of $\bar{\eta}$ in expression (14) been changed from 19 (broadcast traffic) to 3.5. T_{single} would not be affected, but T_{total} would drop to

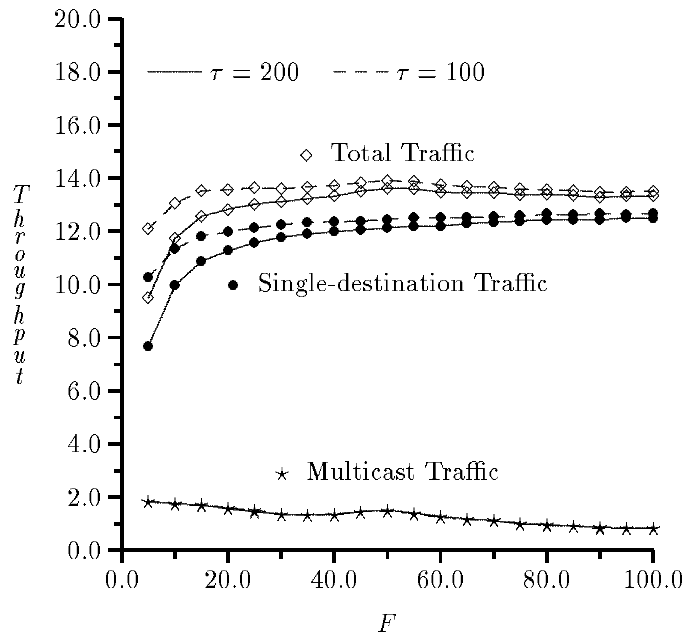

Figure 17: Effect of F on throughput of GMP ($\rho = 0.4, L_{min} = 30, L_{max} = 50, \tau = 100, 200$, maximum multicast group size = 5).

10.31, as compared to almost 14.0 for the three adaptive protocols (Figures 18 and 19). The protocols increase the throughput by permitting single-destination packet transmissions in multicast slots. But these transmissions are adjusted to reflect the current multicast groups. Consequently, the three protocols would permit no such transmissions under broadcast traffic, achieving a total throughput identical to that in Figure 13. We conclude that the adaptive protocols have the best performance under changing traffic parameters.

8 Concluding Remarks

In this paper we have addressed the problem of carrying both multi- and single-destination traffic over single-hop WDM networks. We presented an optimization heuristic, suitable under a mixed traffic scenario, to overcome the inefficiencies and unfair behavior of schedules optimized for one class of traffic only. We also developed a suit of adaptive multicast protocols that can be useful under changing traffic conditions. Our results indicate that schedules for tunable receivers (FT-TR or TT-TR networks) achieve a higher channel utilization compared to schedules for fixed-receiver (TT-FR) networks. We also conclude that slot assignment adaptability is both desirable and feasible when multi-destination traffic is being carried.

Our current research focuses on: (a) how to apply the techniques presented in this paper to provide higher-level services to applications (for example, response collection [16]), and (b) how to hide the transceiver tuning time, now assumed to be included in the slot duration, by overlapping it with transmissions from other stations, thus making single-hop networks practical for current state of the art optical tunable devices.

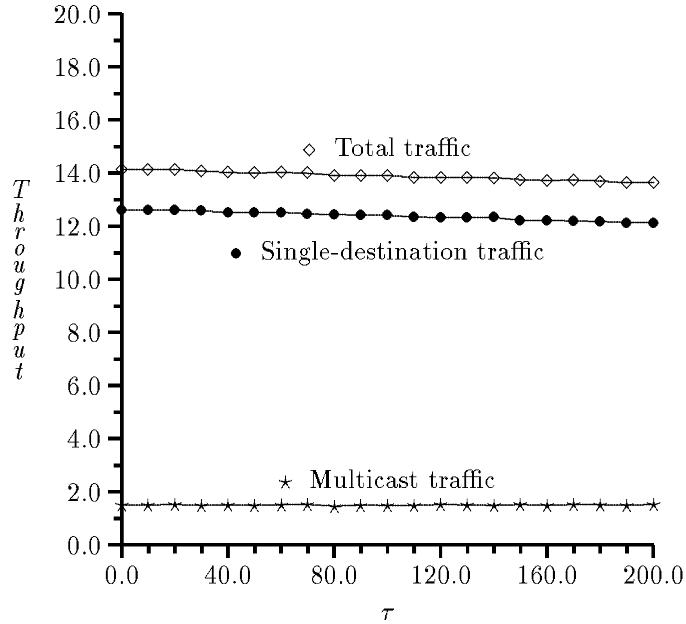

Figure 18: Effect of τ on throughput of GMP ($\rho = 0.4, L_{min} = 30, L_{max} = 50, F = 50$, maximum multicast group size = 5).

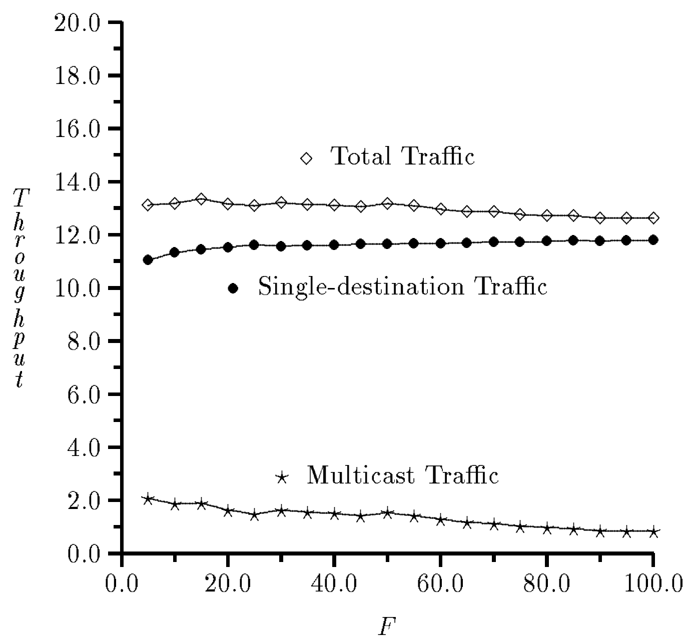

Figure 19: Effect of F on throughput of PMP ($\rho = 0.4, L_{min} = 30, L_{max} = 50$, maximum multicast group size = 5).

References

[1] M. Ahamad and M. H. Ammar. Performance characterization of quorum-consensus algorithms for replicated data. *IEEE Transactions on Software Engineering*, 15(4):492–496, April 1989.

[2] D. Gifford. Polychannel systems for mass digital communication. *Communications of the ACM*, 33:141–151, Feb 1990.

[3] B. Rajagopalan. Reliability and scaling issues in multicast communication. In *Proceedings of SIGCOMM '92*, pages 188–198. ACM, 1992.

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

[5] B. Mukherjee. WDM-Based local lightwave networks Part II: Multihop systems. *IEEE Network Magazine*, pages 20–32, July 1992.

[6] C. A. Brackett. Dense wavelength division multiplexing networks: Principles and applications. *IEEE Journal on Selected Areas in Communications*, SAC-8(6):948–964, August 1990.

[7] 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.

[8] 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.

[9] R. Chipalkatti, Z. Zhang, and A. S. Acampora. High speed communication protocols for optical star coupler using WDM. In *Proceedings of INFOCOM '92*, pages 2124–2133. IEEE, May 1992.

[10] P. A. Humblet, R. Ramaswami, and K. N. Sivarajan. An efficient communication protocol for high-speed packet-switched multichannel networks. In *Proceedings of SIGCOMM'92*, pages 2–13. ACM, 1992.

[11] 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.

[12] K. M. Sivalingam, K. Bogineni, and P. W. Dowd. Pre-allocation media access control protocols for multiple access WDM photonic networks. In *Proceedings of SIGCOMM '92*, pages 235–246. ACM, 1992.

[13] G. N. Rouskas and M. H. Ammar. Analysis and optimization of transmission schedules for single-hop WDM networks. In *Proceedings of INFOCOM '93*, pages 1342–1349. IEEE, March 1993. An expanded version has been submitted to the IEEE/ACM Transactions on Networking.

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

[15] A. Itai and Z. Rosberg. A golden ratio control policy for a multiple-access channel. *IEEE Transactions on Automatic Control*, AC-29(8):712–718, Aug 1984.

[16] M. H. Ammar and G. N. Rouskas. On the performance of protocols for collecting responses over a multiple-access channel. In *Proceedings of INFOCOM '91*, pages 1490–1499. IEEE, April 1991. To appear in IEEE Transactions on Communications.