# Multi-destination communication in broadcast WDM networks: a survey

**Dhaval Thaker**
**George N. Rouskas**
North Carolina State University
Department of Computer Science
Raleigh, North Carolina USA

**ABSTRACT**

In this paper we survey protocols and scheduling algorithms designed to transport multi-destination traffic in broadcast WDM networks. The protocols are classified based on the underlying strategy used to transmit multicast packets, as well as on their assumptions regarding the network architecture. A number of approaches for scheduling both single- and multi-destination traffic are also described and discussed. We discuss the advantages and disadvantages of each scheme, and we also identify the regions of network operation for which each strategy is most appropriate.

## 1 Introduction

Optical networks employing wavelength division multiplexing (WDM) are now a viable technology for implementing a next-generation network infrastructure that will support a diverse set of existing, emerging, and future applications [10]. WDM bridges the gap between the lower electronic switching speeds and the ultra high transmission speeds achievable within the optical medium. WDM divides the enormous information carrying capacity of a single mode fiber into a number of channels, each on a different wavelength and operating at the peak electronic speed, making it possible to deliver an aggregate throughput in the order of Terabits per second. While WDM technology initially was deployed in point-to-point links, and has also been extensively studied, both theoretically and experimentally, in wide area or metropolitan area distances [9], a number of WDM local area testbeds have also been implemented [8] or are currently under development [7, 1]. To realize WDM local area networks, a passive star coupler is employed as a broadcast medium to connect all nodes in the network. Since the entire path between source and destination in such a network is entirely optical, and no electro-optic conversion of the signal is necessary, these networks are also known as *single-hop* WDM networks [10].

Multicasting, the ability to transmit a message from a single source node to multiple destination nodes, has emerged as one of the essential features of current and future networks [2]. With the development of computer and communication applications such as distributed computing, audio and video conferencing, software and video distribution, and database replication, support for multicasting must be an integral part of network design, rather than an afterthought, regardless of the network's underlying technology, data rates, or geographical reach. In this paper, we survey some of the approaches and techniques proposed in the literature to transport multi-destination traffic in broadcast WDM local area networks.

In a point-to-point network, a transmission by a node is received only by the node at the other end of the link. In a single-channel broadcast network, on the other hand, a transmission by a node is received by all the nodes attached to the channel. WDM broadcast networks occupy the center of the spectrum between these

**This work was supported by the Defense Advanced Research Projects Agency under grant F-30602-00-C-0034, and by the National Science Foundation under grant NCR-9701113 (CAREER).*

two extremes by providing a unique one-to-many transmission. Specifically, a transmission by a node in a broadcast WDM network on a given channel (wavelength) is received by *all* nodes listening on that channel at that point in time. This feature makes it possible to implement a number of different approaches for carrying multi-destination traffic in such a network, ranging from separately transmitting a copy of a message to each of its destination nodes, to transmitting multiple copies of the message with each copy received by a subset of the destination nodes, to transmitting a single copy of the message to all destinations at once. The main challenge in the design of efficient multicast scheduling algorithms (MSA) for broadcast WDM networks is in exploiting the one-to-many transmission feature to provide a balance between two conflicting goals, namely, minimizing the number of copies of a message that need to be transmitted (a measure of the bandwidth efficiency of the algorithm) and maximizing concurrency (a measure of the ability of the algorithm to efficiently utilize the available wavelengths). Providing such a balance is important in order to achieve the maximum possible utilization of the channel, receiver, and transmitter resources within a network with multi-destination traffic [20, 19].

This survey paper is organized as follows. In Section 2 we first present a model of the broadcast WDM network with multi-destination traffic, we define performance measures relevant in a multicast setting, and we also point out and discuss the various network parameters and characteristics that affect the design of multicast scheduling algorithms. In Section 3, we present and classify a number of strategies and algorithms for carrying multi-destination traffic in a broadcast WDM network. Since in any realistic environment multi-destination traffic will coexist with single-destination traffic, in Section 4 we discuss several strategies to support a combined load of single- and multi-destination traffic. We conclude the paper in Section 5.

## 2 Broadcast WDM Networks

As seen in Figure 1, an optical broadcast WDM network consists of a set $\mathcal{N} = \{1,2, \ldots ,N\}$ of nodes interconnected by a passive star coupler that supports a set $C = \{\lambda_1,\lambda_2, \ldots ,\lambda_C\}$ of wavelengths. In a typical network, the number of channels $C$ is at most equal to the number of nodes $N$, $C \leq N$. When the number of channels is strictly less than the number of nodes, we will say that the network is *wavelength (or bandwidth) limited*. Each node is equipped with a number of either fixed tuned or tunable transmitters and receivers that can be used for data communication. For simplicity, we assume that the tunable components (either transmitters or receivers) can tune to, and transmit/listen on any of the $C$ wavelengths. If the operation of the network relies on the presence of a control channel, then a separate pair of

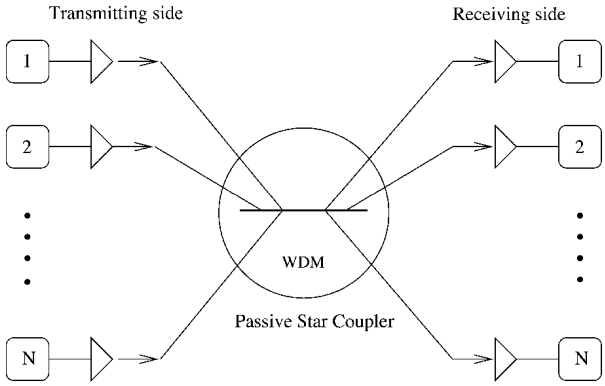


**Figure 1:** A broadcast WDM network.

transceivers is required for every node. In general, this pair of transmitter and receiver is fixed tuned to the predetermined wavelength of the control channel, and cannot be used for data communication.

The network is packet switched, with fixed size packets. Time is slotted with the slot time equal to the packet transmission time, plus, possibly the tuning latency (if it is assumed small compared to the packet transmission time). The tuning latency is defined as the time taken by transceivers to tune from one wavelength to another. All the network nodes are synchronized at the slot boundaries. Since we consider multicast traffic in this paper, we let $g \subseteq \mathcal{N} = \{1,2, \ldots ,N\}$ represent the destination set (multicast group) of a packet. We also let $G$ represent the number of currently active multicast groups. In general, at any given time, $G$ will be considerably smaller than the possible number of multicast groups which is equal to $2^N$.

We define the *wavelength throughput* $S$, $S \leq C$ of the network as the average number of packets transmitted on the $C$ channels per unit of time (slot). We note, however, that while high wavelength throughput is certainly desirable, this traditional definition of throughput does not accurately reflect the performance of a network with multicast traffic, as it fails to capture the *degree of efficiency* in the use of channel bandwidth. A measure of this efficiency is the average number $\bar{l}$ of times a packet is transmitted before all members of its multicast group receive it. Thus, both $S$ and $\bar{l}$ are important in characterizing the performance of the network. For example, a system that can achieve high wavelength throughput only by unnecessarily replicating each multicast packet (resulting in a high $\bar{l}$ value) may actually be inferior to one with a somewhat lower wavelength throughput but which is very efficient in how it transmits packets (i.e., it achieves a very low value for $\bar{l}$ ).

Let a *multicast completion* denote the completion of a multicast transmission of a packet to all receivers in its multicast group. We define the *multicast throughput* $D$ of the system as the average number of multicast completions per slot. This definition of throughput is independ-

ent of how multicast is actually performed (i.e., by performing a single or multiple transmissions), and thus is applicable to any network with multicast traffic. The multicast throughput is related to wavelength throughput and the degree of efficiency through the expression: $D = S/\bar{l}$. As we can see, the multicast throughput $D$ combines both parameters $S$ and $\bar{l}$ in a meaningful way, and it naturally arises as the performance measure of interest in a WDM network with multicast traffic.

In a broadcast WDM network, users contend for resources including the data and control channels and the transmitters and receivers at the various nodes. Successful and efficient transmission of multicast packets requires careful coordination and scheduling of these resources. Some form of coordination is necessary because a transmitter and a receiver must both be tuned to the same channel for the duration of a packet's transmission. Also, the network must avoid or minimize packet loss due to *collisions*, which take place when two or more nodes simultaneously transmit on the same channel, and *destination conflicts*, which arise when two or more packets, each on a different channel, are addressed to a single receiver in the same slot. These issues become more difficult to tackle in the presence of multi-destination traffic in the network. Thus, at the heart of every media access control protocol for broadcast WDM environments is a scheduling algorithm responsible for coordinating access to the available channels.

The design of strategies or scheduling algorithms for carrying multi-destination traffic is strongly dependent on the underlying assumptions regarding the architecture and parameters of the broadcast WDM network. Differences in issues ranging from the existence of a control channel to the tunability characteristics of each node to the tuning latency of the optical transceivers can result in radically different scheduling algorithms. For the rest of this section we take a closer look at the issues which can affect the design of algorithms for scheduling multi-destination packets in a broadcast WDM environment.

**Number of transceivers per node and tunability characteristics.** The number of transceivers per node used for data communication and their tunability characteristics can have a profound effect not only on the design but also the performance of multicast scheduling algorithms (MSAs). In the papers surveyed, the transmitters can be either fixed tuned or tunable, but the receivers are always tunable. This node structure is not surprising given the fact that tunability at the receiving end can support multicasting in a natural and flexible manner by allowing a single packet transmission on a certain wavelength to be received by multiple destinations which have tuned their receivers to that wavelength. It has also been observed that the presence of multiple tunable receivers per node may significantly increase the maximum achievable throughput [4, 16, 17]. Employing additional tunable components allows the designer greater flexibility in the design of an MSA by alleviating the problem of destination conflicts (discussed below), which can be a severe one when multicast traffic is considered. On the other hand, having multiple tunable transceivers per node can increase the complexity of the MSA, especially when the tuning latency cannot be neglected. In this case, minimizing the effect of the tuning latency in the schedule involves careful coordination not only among the various nodes, but also among the various tunable transceivers at each node.

**Tuning latency.** While optical device technology has made great advances in the past few years, electronic speeds are also increasing to 10 Gigabits per second and beyond. Consequently, depending on the packet size and the data rate in the network, the value of the transceiver tuning latency relative to the packet transmission time can have a significant impact on the complexity of the MSA. If the tuning time is negligible compared to packet transmission time [26, 12, 3, 16, 5], it can be accounted for by including appropriate guard bands around the data packet within each slot. In this case, simpler preemptive scheduling algorithms can be employed, since, at the end of each slot, a transceiver can tune to a different wavelength without incurring any cost (i.e., without increasing the length of a schedule). If tuning latency is large, including it within each slot is inefficient and can lead to very long delays and low throughput. Thus, sophisticated MSAs that explicitly address the tuning latency are needed. Typically, non-preemptive algorithms are employed [20, 19] which prevent frequent retunings by having a transceiver tune to a wavelength and complete a number of packet transmissions/receptions before tuning to a different channel. The design of non-preemptive algorithms is more complex compared to preemptive ones. Non-preemptive algorithms also have higher running-time requirements, but they can effectively mask the tuning latency and thus significantly reduce the amount of time required to clear a set of traffic demands.

**In-band vs. out-of-band signaling.** Most broadcast WDM architectures that have appeared in the literature require the use of a control channel. The control channel is mainly used for the exchange of queue and traffic information among the nodes in the network, for slot reservation, as well as for other functions including network management and monitoring and global clock distribution. In general, one additional wavelength is required for the exchange of control information, and this wavelength cannot be used for data transmission. Systems with a centralized architecture, such as the ones in [12, 16], require two wavelengths for out-of-band signaling, one for sending and another for receiving control information from the scheduler. On the other hand, it is also possible to use in-band signaling that does not require a separate channel for control information. One example of an architecture which employs a distributed reservation protocol [24] to transmit the information needed by the MSA along with the transmission of data can be found [20, 19].

**Bandwidth allocation.** Three different approaches have been proposed in the literature for allocating the bandwidth among the network nodes. In the *pre-allocation* approach [26], the channel bandwidth is divided into slots and slots are pre-allocated to each node. In each pre-allocated slot, two nodes tune their transmitter and receiver, respectively, to the same channel for communication. The slot pre-allocation can be fixed (i.e., independent of incoming traffic) or it could be dynamic to handle traffic variations. For dynamic pre-allocation, all the participating nodes in the network compute a schedule. Schedule computation can be distributed or centralized. This approach can generate large allocation tables and can be computationally intensive when there are many multicast groups and/or groups of large size.

In the *reservation-based* approach [5], nodes reserve the slots for transmission. The reservation process is carried out on a separate control channel. Access to the control channel can be pre-allocated or random. The reservation-based approach may result in large packet delays for multicast packets with large destination sets. In the *random access* approach [16], nodes randomly access the channels to transmit packets. If there is a collision or destination conflict, nodes retransmit after a random or fixed interval, depending on the protocol. This approach is similar to conventional media access protocols such as Ethernet, and uses the broadcasting ability of the passive star coupler. This approach has the advantage of simpler scheduling compared to other approaches but it may lead to low throughput.

**Centralized vs. distributed architecture.** In a distributed architecture, schedule computation is performed by each node independently [3, 5, 22]. All the nodes share the necessary queue information and other traffic parameters using the control channel, and use the same algorithm to construct identical deterministic schedules. In a centralized architecture [12, 16], there is a single scheduler at the passive star coupler. This approach requires two control channels, one for sending control information to the scheduler and the other for receiving the schedule from it. In a centralized system, the scheduler knows the state of the network at any instant and it can schedule a retransmission immediately with out the overhead that is incurred in a distributed computation. The scheduler must continuously perform three tasks: receive a request, compute a schedule, and assign a slot for transmission to each node. Since these three tasks are performed by a single entity, as the number of channels and/or the data rate at which they operate increases, the load on the scheduler can be enormous. To reduce the processing requirement on the centralized scheduler, it is suggested in [12, 16] that very simple scheduling algorithm be employed. The centralized schedule computation approach is more suitable when the nodes are closely spaced (e.g., in a rack). For geographically distributed nodes, distributed schedule computation can reduce the control overhead and offers robustness against network failures.

## 3 Scheduling Algorithms for Multi-Destination Traffic

In this section we present and discuss the various MSAs for broadcast WDM networks that have appeared in the literature. We will use the framework introduced in [13] to classify the different MSAs. This framework was developed in the context of an $N \times N$ multicast packet switch. We note that a packet-switched WDM network with $N$ nodes and $C$ wavelengths can be modeled as a bandwidth-limited $N \times N$ input-queued space-division switch operating in a time-slotted mode. The bandwidth limitation is due to the fact that the number of wavelengths available with tunable optical devices is smaller than the potential number of nodes ($C < N$). As a result, in each time slot, a maximum of $C$ nodes may transmit their packets into the optical medium. On the other hand, a multicast switch with no bandwidth limitation ($C = N$) is potentially capable of switching packets from all $N$ nodes (input ports) to their destinations (output ports) in one time slot.

Using the terminology of [13], the strategies underlying the various MSAs can be classified in three broad categories. Note that the term *fanout* refers to the number of destinations of a multicast packet (i.e., the multicast group size).

1. **Unicast (sequential) service.** One copy of a multicast packet is separately transmitted to each of the destinations in the multicast group. Hence, the transmission of a packet takes at least as many slots as the number of destinations. This strategy results in high wavelength throughput but low multicast throughput (see Section 2), since essentially the same data packet is transmitted again and again.
2. **Multicast service with no fanout splitting.** Instead of transmitting a multicast packet to its destinations one at a time, another extreme is to insist that all destinations receive the packet in the same time slot. This strategy makes very efficient use of the bandwidth, since each multicast packet is transmitted exactly once. However, when the active multicast groups are not disjoint, this strategy can have poor performance in terms of both multicast throughput and delay.
3. **Multicast service with fanout splitting.** In between the above extreme strategies we have the multicast service discipline of fanout splitting, with better throughput and delay performance than either extreme. A multicast packet can be transmitted to more than one destination in a given time slot, depending on the availability of the destinations. The remaining destinations (if any) are served in later slots. In essence, the destination set of a multicast packet is partitioned into subgroups, and the packet is sequentially transmitted to each subgroup. A number of dif-

ferent fanout splitting strategies may be implemented based on the manner in which the destination set is partitioned, and will be discussed later. We also note that this is the most general service strategy, since unicast service and multicast service with no fanout splitting are special cases where the number of subgroups is equal to the fanout and one, respectively.

Finally, we note that the work in [6] also considered the problem of switching multicast traffic in a time-slotted switch with no fanout splitting. Specifically, it was shown that the problem of finding a conflict-free assignment of input queued packets to output slots so as to minimize the schedule length is $\mathcal{NP}$-hard. Consequently, it is not surprising that all the MSAs that have appeared in the literature are based on heuristics.

In Table 1 we classify the MSAs that have appeared in the literature according to the strategy they implement, and according to their assumptions regarding the underlying network environment. In the following subsections, we consider each strategy separately, and we discuss in detail the algorithms, which appear in Table 1.

## 3.1 Unicast service

With this strategy, one copy of a multicast packet is sent to each member of the packet's destination set, with each copy transmitted in a different time slot. The main advantage of this approach is that it makes it possible to employ unicast scheduling algorithms which have been extensively studied, are well-understood, and are significantly simpler and computationally more efficient than corresponding multicast scheduling ones. As pointed out in [13], an approximate analysis of this strategy can be carried out by analyzing the corresponding WDM network with unicast traffic and ignoring the batch arrivals of multicast packets. The main drawback is that this strategy does not take advantage of the one-to-many transmission feature of broadcast WDM networks. Consequently, unicast service may result in the transmission of a large number of copies leading to inefficient use of the available bandwidth, i.e. it achieves a low degree of efficiency, and thus low multicast throughput. It was shown in [22,21] that unicast service is appropriate when the average multicast session is short and the average multicast group size is small relative to the number of nodes in the network. In such an environment, the total number of multicast packets in the network will be a small fraction of the unicast packets, and the overhead of implementing and running a specialized MSA may not be justified.

## 3.2 Multicast service with no fanout splitting

The multicast protocols presented in [5, 3, 25, 12] all use MSAs that implement multicast service with no fanout splitting. Under this strategy, the source of a multicast packet insists on transmitting a single copy of the packet in a time slot, which guarantees that all members of the packet's destination set will receive it. In other words, the algorithms require the simultaneous availability of all the three network resources involved in the transmission of a packet, namely, one transmitter of the source node, one receiver at each of the destination nodes, and one channel. While achieving the highest possible degree of efficiency, usually these algorithms achieve low wavelength throughput, and thus low multicast throughput.

The performance of multicast with no fanout splitting was studied in [4]. By making a number of *protocol-free* assumptions, namely, a distributed transmission protocol with no control overhead, collision-less transmission, and no propagation delay on the control channel, an analytical model was developed to determine the performance limits of the network. For the model, tuning latency is assumed to be zero, packet arrivals are taken to be Poisson, and packet lengths are exponentially distributed (note that this work is the only one to assume variable size packets). Each node has a buffer that can hold exactly one packet, and packets that cannot be immediately transmitted to all nodes in their multicast group are dropped. With these assumptions, the network was modeled as a birth-death queuing system, and expressions for throughput and packet drop probability were obtained. It was shown in [4] that while wavelength throughput is low in such a network, *receiver throughput*, defined as the average number of busy receivers, could be higher. The latter result is due to the fact that multiple nodes are involved in (i.e., receive) each packet transmission. The results are in agreement with [22, 21] were it was shown that multicast with no fanout splitting works well only when the average multi-

| Schedule Computation | Tuning Latency | Multicast Service with | | | |
|---|---|---|---|---|---|
| | | No Fanout Splitting | | Fanout Splitting | |
| | | Reference | Node Structure | Reference | Node Structure |
| Centralized | Zero | [12] | $CC^2$-$FT^1$-$TR^1$ | [16] | $CC^2$-$TT^1$-$TR^m$ |
| | Zero | [3] | $CC^1$-$TT^1$-$TR^1$ | [14] [15] | $CC^1$-$TT^1$-$TR^1$ |
| Distributed | Zero | [4] | $CC^1$-$TT^n$-$TR^m$ | [17] | $CC^1$-$TT^1$-$TR^m$ |
| | Arbitrary | [5] [25] | $CC^1$-$TT^1$-$TR^1$ | [19] [20] | $FT^1$-$TR^1$ |

**Table 1:** Classification of MSAs (CC: control channel, FT: fixed transmitter, TT: tunable transmitter, TR: tunable receiver).

cast session is long and the average multicast group size is comparable to the number of nodes in the network. (i.e., a broadcast or nearly broadcast scenario).

The four MSAs [5, 3, 25, 12] share a number of assumptions regarding the underlying network environment. Specifically, they all assume the presence of a control channel and the availability of tunable transmitters and receivers for data communication. The protocols differ in their assumptions regarding the tuning latency, the mechanism used to access the control channel and the details of operation of the MSA, as discussed in the following subsections.

### 3.2.1 TDMA access to the control channel

The multicast protocol in [5] uses TDMA in the control channel. Each node accesses the control channel in a round-robin fashion and transmits a control packet. The control packet contains a multicast address identifying the multicast group nodes. Upon receiving a control packet, all nodes in the network simultaneously run the MSA in a distributed and deterministic fashion to determine the time slot and the channel on which the source of the control packet will transmit to the multicast group. Since all nodes have access to the same information and run the same algorithm, they will compute the same schedule, and both the source and the intended receivers will know when and in what wavelength to tune for the multicast packet transmission to be successful.

The MSA employed is relatively simple, and it is based on the earliest availability of all necessary resources: channel, transmitter and receivers. First, the earliest time $T_r$ at which all the receiver nodes in the group become free is determined. Next, the earliest time $T_s$ at which both the source transmitter and the channel on which it is currently tuned are free is computed. If both are free then a new transmission can be scheduled on this channel, avoiding a tuning delay at the transmitter. If the channel is busy but the transmitter is free, then $T_s$ is computed as the earliest time that another channel becomes free. At time $t = \max\{T_s, T_r\}$ all the receivers in the multicast group tune to the channel to receive the multicast transmission. Note that both $T_r$ and $T_s$ are computed so as to account for the tuning time at the transmitter and receivers, thus, this algorithm can accommodate arbitrary transceiver tuning latencies.

While computing the earliest times $T_r$ and $T_s$, the algorithm reserves the receivers of a multicast group as they become free until all receivers in the group become available. This feature can significantly limit the achievable throughput since reserved receivers cannot be used for other communication. To improve the performance, a modification was suggested in [25]. The modified MSA, known as *Backtrack* MSA improves the throughput by scheduling additional multicast transmissions to some of the free receivers, which are waiting for other busy receivers to become free.

The *Backtrack* MSA works as follows. First, the MSA in [5] is run to obtain a schedule as before. Now consider a new multicast request with source $s$ and multicast group $g$. Instead of running the MSA to find $T_r$ and $T_s$ for this request, the current schedule is first searched for slots in which a transmitter of $s$ and a receiver for each node in $g$ are free (possibly waiting for some busy receiver(s) to become free). If consecutive slots with this property are found that can accommodate the request, then the schedule is modified to include the multicast transmission from $s$ to $g$ in these slots. By satisfying this request without increasing the schedule length, the *Backtrack* MSA improves network performance in terms of both average packet delay and throughput. Overall, however, wavelength throughput can be very low for both protocols [5, 25].

A different protocol and scheduling algorithm for the same problem is presented in [12]. Each node sends its multicast (and unicast) transmission requests to a central controller via a control channel, the controller computes the schedule and broadcasts it to all nodes, again on the control channel. The controller uses a slot decomposition technique, similar to that used in satellite-switched TDMA (SS/TDMA) systems [11] to construct a slot matrix (schedule) which defines how transmissions should take place within the slots. For purposes of scheduling, unicast packets are assigned a weight of 1, while multicast packets to a group of size $k$ are assigned a weight of $1/k$. The slot matrix is constrained to have elements with values of at most 1. The slot decomposition algorithm constructs a slot matrix free of any conflicts, and such that a multicast packet is transmitted in a single slot (i.e., no fanout splitting).

### 3.2.2 Random access to the control channel

The multicast protocol presented in [3] also employs an MSA that insists on transmitting a packet to all destinations in its multicast group, but it uses a different access method for the control channel. Time on the control channel is divided into two phases, a "contention" phase and a "contention-less" phase. During the contention phase, nodes use the slotted Aloha protocol to transmit reservations for multicast transmissions. A reservation is considered successful if all three conditions hold true: (1) the reservation does not collide with other requests in the contention phase of the control channel (no control channel collision), (2) the multicast group specified in the reservation does not have any nodes in common with the groups specified by reservations transmitted in previous slots within this contention phase (no destination conflict), and (3) the total number of previously successful reservations is less than the number of available data channels (no data channel collision).

If a node fails to reserve a transmission slot due to any of the above conflicts, it wins a slot in the "con-

tention-less" phase of the *next* cycle of the control channel. Again, before allocating a mini-slot in the "contention-less" part in the next cycle, all of the above three conflicts should be resolved. Every node in the network monitors the control channel and is aware of the reservations that have been successful at any given time. Once a successful reservation is made, all the receivers in the multicast group tune to the transmitter node's wavelength; the algorithm assumes that tunable devices take a negligible time to tune to a different wavelength. Since it is assumed that the multicast transmission is completed in one slot, the control channel can become a bottleneck, as it is necessary to incur the reservation overhead for each and every multicast packet.

## 3.3 Multicast with fanout splitting

When fanout splitting is used, the multicast group of a packet is partitioned in subgroups, and the packet is sequentially transmitted to each subgroup. This strategy can result in a dramatic improvement in network performance, since packet transmissions can take place whenever a transmitter of the source node and a receiver at one or more destination nodes are available, without having to wait for all receivers to become free. Two issues arise in this case: (1) how to split (partition) groups with common receivers and (2) how to coordinate (schedule) the tuning of subgroups of receivers across the various channels. In the following subsections we consider three approaches that have appeared in the literature to address these issues.

### 3.3.1 Greedy scheduling algorithms

The work in [14] is based on the same architecture as in [5], with the exception that tuning latencies for the receivers are considered negligible. To improve the channel utilization of the network, [14] employs fanout splitting, and arranges the multicast transmissions in the schedule with the objective of minimizing the average receiver waiting time. The problem of scheduling multicast transmissions to subgroups of receivers is defined and referred to as the Multicast Partition Problem. Two greedy heuristics are then developed to solve the problem. The first heuristic, called the Earliest Available Receiver (EAR), schedules a transmission by the source to the first receiver, which becomes free. If additional receivers become available during this transmission, a transmission by the source to these receivers is scheduled immediately after the completion of the first one.

The second greedy approach, called the Latest Available Receiver (LAR), first schedules a transmission at the time the last receiver in a group becomes available. Next, LAR attempts to schedule earlier transmissions to other members of the group without creating any channel or receiver conflicts. A third variant called the Best Available Receiver (BAR), combines EAR and LAR to obtain schedules that minimize the receiver waiting time. Though BAR constructs better schedules than either EAR or LAR, its running time is higher than the other two heuristics. All three heuristic MSAs make the assumption that receivers take negligible time to tune across channels.

A different approach was presented in [15], where the problem of partitioning the destination set of each packet and scheduling the transmissions so as to minimize the packet delay is studied. The problem is shown to be $\mathcal{NP}$-hard, and a heuristic is presented and compared to an algorithm with no fanout splitting. The main idea behind the heuristic is to schedule as many destinations as possible to receive the packet in the same slot. Simulation results indicate that partitioning the multicast group performs well when the network is not bandwidth limited. Otherwise (i.e., when the number $C$ of channels is small compared to the number $N$ of nodes), the no fanout splitting strategy can perform better in terms of packet delay.

### 3.3.2 Random scheduling algorithms

The work in [17] models a broadcast WDM network with $N$ nodes and $C$ wavelengths as a bandwidth limited time-slotted $N \times N$ switch, and extends the analysis first presented in [13] to obtain the saturation throughput when the nodes have one or more tunable receivers. It is assumed that tuning latency is negligible, and that a separate channel is used to carry relevant control information. The analysis considers a random selection policy at both the transmitting and the receiving ends. Specifically, at each time slot, a random set of $C$ nodes is selected from among the $N$ nodes and are allowed to transmit their multicast packets (note that since the performance parameter studied is saturation throughput, all the nodes are assumed to be constantly backlogged). Destination conflicts are also resolved in a random manner. In particular, if two or more nodes in a slot have packets for the same receiver, then the receiver selects one of the multicast packets destined for it with equal probability. By making the assumption that, when a node is selected to transmit, each of the nodes in its multicast group receives the packet with a constant probability independently of other receivers, a queuing model is developed from which the saturation throughput and average packet delay are obtained. The analysis is also extended to the case when each node has multiple receivers. It is shown that, if the number $C$ of channels is small, then network performance is limited by insufficient bandwidth. However, if the number of channels is relatively large, performance is limited by the occurrence of destination conflicts, and thus, employing multiple receivers per node can significantly increase the throughput and decrease the average delay.

The work in [16] also employs fanout splitting, and, as in [17], tuning latencies are taken to be negligible and

nodes are constantly backlogged and may have multiple receivers. Unlike in [17], however, the algorithms are designed for a centralized architecture in which a master scheduler maintains complete information about the state of the network, and instructs transmitters and receivers to tune to the appropriate channels.

At the transmitting end, the algorithm uses a random selection policy such that, when a node completes the transmission of a multicast packet (i.e., as soon as the packet is received by all members of its multicast group), another node, not currently in the middle of a multicast transmission, is randomly selected to transmit on this channel in the next slot. (Note that, since $C < N$, $C$ nodes are involved in a multicast transmission during any given slot, while $N - C$ nodes are backlogged and waiting to start a multicast transmission.) Two transmission policies are considered and analyzed. In the first, a node repeatedly transmits a packet until all nodes in the multicast group receive it. This policy has an important drawback. When several nodes have packets for the same receiver, a likely situation at high loads and for large multicast group sizes, the destination conflicts will persist over long periods of time, aggravating the head-of-line blocking effect and resulting in poor performance. To improve the situation, another transmission policy is proposed in [16]. Instead of continuously transmitting a packet, a node waits for a random delay between retransmissions. Since other nodes may access the channel between retransmissions, this policy alleviates the head-of-line blocking problem and achieves higher throughput.

The resolution of destination conflicts, an important issue in a multicast setting, is also considered in [16]. If two or more transmitters have packets for the same node, that node must decide which packet to receive. The conflict resolution algorithm may base its decision on traffic priorities, the time of arrival or the fanout size of the multicast packets, the amount of delay accumulated by the contending packets, etc. In [16] three conflict resolution policies are compared: one that randomly selects a packet, one that selects the packet with the earliest arrival time (FCFS), and one that selects the packet with the smallest number of (remaining) destinations. The intuition behind the last policy is that it maximizes the probability that a message will be released (received by all its destinations), thereby making way for a new message. Analytical and simulation results indicate that this policy performs better than either the FCFS or the random policies. As in earlier works, it is also shown that an improvement in performance is achieved when nodes have multiple receivers.

### 3.3.3 The virtual receiver concept

The MSAs discussed in the previous two sections attempt to simultaneously solve the two issues that arise in fanout splitting, namely, the partitioning of the multicast groups and the scheduling of transmissions. Both issues are difficult to deal with, especially in the presence of non-negligible tuning latencies (note that all algorithms discussed so far ignore tuning latencies) and when receivers may belong to multiple multicast groups. Furthermore, all algorithms attempt to partition the destination set of each packet into subgroups, an approach that has two drawbacks. First, since each packet is considered independently of others, the algorithms may not achieve good performance for the network overall. Second, significant overhead is incurred when a partitioning and scheduling decision has to be made for each packet.

The virtual receiver concept was developed in [19, 20] as a novel way to perform fanout splitting that overcomes these problems. A *virtual* receiver $V \subset \mathcal{N}$ is a set of *physical* receivers that behave identically in terms of tuning. Thus, from the point of view of coordinating the tuning of receivers to the various channels, all physical receivers in $V$ can be logically thought of as a single receiver. A virtual receiver set $\mathcal{V}$ is defined as a partition of the set $\mathcal{N}$ of physical receivers into a number of virtual receivers. Given a virtual receiver set $\mathcal{V}$, a multicast packet with destination set $g$ must be transmitted to all virtual receivers $V \in \mathcal{V}$ such that $V$ contains a destination of the packet (i.e., $g \cap V \neq \emptyset$). All receivers in $V$ have to filter out packets addressed to multicast groups for which they are not a member, but they are guaranteed to receive the packets to all groups of which they are members.

In effect, a virtual receiver set transforms the original network with $N$ transmitters, $N$ receivers, and multicast traffic, to an equivalent network with $N$ transmitters, $|\mathcal{V}|$ receivers, and unicast traffic. Thus, the virtual receiver concept decouples the problem of determining how many times a multicast packet should be transmitted (i.e., of partitioning the multicast groups) from the problem of scheduling the packet transmissions. As a result, one can take advantage of a wide range of algorithms that have been designed for unicast traffic, have well-understood properties, and which can handle arbitrary tuning latencies. For instance, [20] uses the algorithms developed in [23].

The work in [20] concentrates on the problem of optimally obtaining a virtual receiver set that maximizes multicast throughput, which is shown to be $\mathcal{NP}$-hard. Four heuristics of varying degree of complexity are then presented for selecting the virtual receivers so as to provide near-optimal performance. Since the virtual receiver set is selected by considering the total traffic demand to the network, this approach achieves better performance than is possible when each packet is considered independently of others.

## 4 Combined Scheduling of Single- and Multi-Destination Traffic

In any realistic environment, the network load will consist of a mix of single- and multi-destination traffic. This problem was specifically studied in [21, 26], and it

has also been addressed by several other authors. This section discusses strategies for scheduling such a combined traffic load.

In [22], the observation was made that the scheduling algorithm to be used will depend on the relative amount of each type of traffic offered to the network. Three types of multicast traffic were identified, and it was suggested that different algorithms be used to schedule this traffic along with unicast traffic.

- **Type 1 multicast traffic** is such that the typical multicast session lasts for a short time, but the average multicast group size is large (a broadcast or nearly broadcast scenario). For this type of traffic it was suggested in [22] that all nodes in the network periodically tune their receivers to the same channel in the same slots (called *broadcast* slots) to receive multicast transmissions (nodes must then filter out transmissions not intended for them).
- **Type 2 multicast traffic** is such that both the typical multicast session and the average group size is small. In this case, it is suggested that multicast packets be replicated and transmitted to each destination separately.
- **Type 3 multicast traffic** is such that the duration of the average multicast session is long (the average group can be of any size). Since multicast traffic can be a significant component of the overall traffic, it was suggested that multicast packets be transmitted in special slots, called *multicast* slots. Multicast slots are defined for each group, and all nodes in a group tune their receivers to the same channel in the group's multicast slots to receive transmissions from a certain source. Adaptive multicast protocols were designed in [22] to dynamically allocate multicast slots so as to keep channel utilization at high levels.

The work in [26] defines a two-dimensional multicast threshold which is a function of the session duration and the group size, and which quantifies some of the ideas developed in [22]. The main conclusions regarding scheduling of a combined load of single and multi-destination traffic are very similar to those in [22].

The following strategies have appeared in the literature for scheduling both single- and multi-destination traffic.

1. **Unicast Traffic as Special Case of Multicast Traffic.** Many protocols that have appeared in the literature, including [16, 17 ,5], account for unicast traffic by allowing multicast groups of size one. By appropriately selecting a distribution of multicast group sizes, it is possible to study (analytically or via simulation) the performance of the network under a wide range of traffic scenarios. One of the advantages of this approach is that a single scheduling algorithm is used in the network. The strategy was extensively studied using simulation in [21], and it was found that it produces good schedules under a wide range of traffic scenarios and network parameters.
2. **Multicast Traffic Treated as Unicast Traffic.** As we mentioned above, replicating each multicast packet and separately transmitting it to each destination works well only for Type 2 multicast traffic [22]. This result was confirmed by the study in [21] where it was shown through comprehensive simulation results that this strategy is appropriate only under limited circumstances, namely, when there are few and short multicast sessions and the group sizes are small.
3. **Separate Scheduling of Unicast and Multicast Traffic.** With this strategy, two schedules are obtained one for unicast traffic and one for multicast traffic. Each schedule is constructed by employing an appropriate unicast or multicast scheduling algorithm, respectively. A schedule for the overall traffic is then obtained by concatenating the two schedules. The main disadvantage of this approach is that two different algorithms must be run in order to compute the overall schedule. However, it was shown in [21] that this strategy produces short schedules, and thus, has good performance in terms of multicast throughput in most network environments regardless of the specific mix of single- and multi-destination traffic.
4. **Schedule Merging Heuristics.** An alternative to separate scheduling, this strategy first constructs two schedules, one for unicast and one for multicast traffic, and then merges the two to obtain a schedule for the overall traffic. As shown in [22], careful merging of the two schedules can result in a schedule that can have good performance in terms of average packet delay and channel utilization. A somewhat similar approach [26] starts with a single schedule for unicast traffic, and appropriately modifies it to include multicast transmissions, by having several receivers tune to the same channel in a slot that was previously designated for unicast transmission.

## 5 Conclusion

We have surveyed algorithms and strategies for scheduling multi-destination traffic in broadcast WDM networks. Due to extensive work in the last few years, this research area is now well understood, and efforts are currently under way to put our knowledge into practice by implementing some of these techniques in testbed environments [7, 1]. One open area that will benefit from future research is the provision of quality-of-service (QoS) for multi-destination traffic in a broadcast WDM environment.

## 6 References


[1] The NGI Helios project. http://www.anr.mcnc.org/projects/Helios/Helios.html.

[2] M. Ammar, G. Polyzos, and S. Tripathi (Eds.). Special issue on network support for multipoint communication. *IEEE Journal of Selected Areas in Communications,* 15(3), April 1997.

[3] M. Bandai, S. Shiokawa, and I. Sasane. Performance analysis of a multicasting protocol in a WDM based single-hop lightwave network. In *Proceedings of Globecom '97,* pages 561-565, 1997.

[4] M. Borella and B. Mukherjee. Limits of multicasting in a packet switched WDM network. *Journal of High Speed Networks,* 4(2):155-167, 1995.

[5] M. Borella and B. Mukherjee. A reservation-based multicasting protocol for WDM local lightwave networks. In *Proceedings of ICC' 95,* pages 1277-1281, IEEE, 1995.

[6] W-T. Chen, P-R. Sheu, and J-H. Yu. Time slot assignment in TDM multicast switching system. *IEEE Transactions on Communications,* 42(1):149-165, 1994.

[7] M. Kuznetsov *et al.* A next-generation optical regional access network. *IEEE Communications Magazine,* 38(1):66-72, January 2000.

[8] E. Hall *et al.* The Rainbow-II gigabit optical network. *IEEE Journal Selected Areas in Communications,* 14(5):814-823, June 1996.

[9] R. E. Wagner *et al.* MONET: Multiwavelength optical networking. *Journal of Lightwave Technology,* 14(6):1349-1355, June 1996.

[10] O. Gerstel, B. Li, A. McGuire, G. N. Rouskas, K. Sivalingam, and Z. Zhang (Eds.). Special issue on protocols and architectures for next generation optical WDM networks. *IEEE Journal Selected Areas in Communications,* 18(10), October 2000.

[11] I. Gopal and C. Wong. Minimizing the number of switchings in an SS/TDMA system. *IEEE Transactions on Communications,* 33(6):497-501, June 1985.

[12] N-F. Huang, Y-J. Wu, C-S. Wu, and C-C. Chiou. A multicast model for WDM-based local lightwave networks with a passive star topology. In *Proceedings of TENCON '93,* pages 470-473, 1993.

[13] J. H. Hui and T. Renner. Queuing analysis for multicast packet switching. *IEEE Transactions on Communications,* 42(2/3/4):723-731, February 1994.

[14] J. Jue and B. Mukherjee. The advantage of partioning multicast transmissions in a single-hop optical WDM network. In *Proceedings of ICC '97,* pages 427-431. IEEE, 1997.

[15] H-C. Lin and C-H. Wang. Minimizing the number of multicast transmissions in single-hop WDM networks. In *Proceedings of ICC 2000,* pages 1645-1649. IEEE, 2000.

[16] E. Modiano. Random algorithms for scheduling multicast traffic in WDM broadcast-and-select networks. *IEEE/ACM Transactions on Networking,* 7(3):425-434, June 1999.

[17] A. Mokhtar and M. Azizgolu. Packet switching performance of WDM broadcast networks with multicast traffic. In *Proceedings of SPIE '97,* pages 220-231, 1997.

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

[19] Z. Ortiz, G. N. Rouskas, and H. G. Perros. Scheduling of multicast traffic in tuneable-receiver WDM networks with non-negligible tuning latencies. In *Proceedings of SIGCOMM '97,* pages 301-310. ACM, September 1997.

[20] Z. Ortiz, G. N. Rouskas, and H. G. Perros. Maximizing multicast throughput in WDM networks with tuning latencies using the virtual receiver concept. *European Transactions on Telecommunications,* 11(1):63-72, January/February 2000.

[21] Z. Ortiz, G. N. Rouskas, and H. G. Perros. Scheduling of combined unicast and multicast traffic in broadcast WDM networks. *Photonic Network Communications,* 2(2):135-154, May 2000.

[22] G. N. Rouskas and M. H. Ammar. Multi-destination communication over tunable-receiver single-hop WDM networks. *IEEE Journal on Selected Areas in Communications,* 15(3):501-511, April 1997.

[23] G. N. Rouskas and V. Sivaraman. Packet scheduling broadcast WDM networks with arbitrary transceivers tuning latencies. *IEEE/ACM Transactions on Networking,* 5(3):359-370, June 1997.

[24] G. N. Rouskas and V. Sivaraman. A reservation protocol for broadcast WDM networks and stability analysis. *Computer Networks,* 32(2):211-227, February 2000.

[25] S-T. Sheu and C-P. Huang. An efficient multicast protocol for WDM star-coupler networks. In *Proceedings of the IEEE International Symposium on Computers and Communications,* pages 579-583, 1999.

[26] Wen-Yu Tseng and Sy-Yen Kuo. A combinational media access protocol for multicast traffic in single-hop WDM LANs. In *Proceedings of Globecom '98,* pages 294-299, IEEE 1998.

**Dhaval Thakar**
**Room #454F, 4th Floor, EGRC,**
**Centennial Campus, NC State University, Raleigh.**
**dthaker@unity.ncsu.edu**

*Dhaval V. Thaker received the B.S. degree in Electronics Engineering from Regional Engineering College, Surat, India in 1996. He is currently persuing the M.S. degree in Computer Networking, North Carolina State University. His Master's thesis is on the design of multicast scheduling algorithms in a partially tunable broadcast WDM network.*

*His current research interests include optical networks architecture, protocols, and Network Security.*

**George N. Rouskas**
**Associate Professor**
**Department of Computer Science**
**North Carolina State University**
**Box 7534**
**446 EGRC—1010 Main Campus Drive**
**Raleigh, NC 27695-7534.**
**rouskas@csc.ncsu.edu**

*George N. Rouskas (S '92, M '95) received the Diploma in Electrical Engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 1989, and the M.S. and Ph.D. degrees in Computer Science from the College of Computing, Georgia Institute of Technology, Atlanta, GA, in 1991 and 1994, respectively. He joined the Department of Computer Science, North Carolina State University in August 1994, and he has been an Associate Professor since July 1999. His research interests include network architectures and protocols, optical networks, multicast communication, and performance evaluation. He is a recipient of a 1997 NSF Faculty Early Career Development (CAREER) Award, and a co-author of a paper that received the Best Paper Award at the 1998 SPIE conference on All-Optical Networking. He also received the 1995 Outstanding New Teacher Award from the Department of Computer Science, North Carolina State University, and the 1994 Graduate Research Assistant Award from the College of Computing, Georgia Tech. He was a co-guest editor for the IEEE Journal on Selected Areas in Communications, Special Issue on Protocols and Architectures for Next Generation Optical WDM Networks which appeared in 2000, and is on the editorial boards of the IEEE/ACM Transactions on Networking and the Optical Networks Magazine. He is a member of the IEEE, the ACM and of the Technical Chamber of Greece.*