The Arrayed-Waveguide Grating-Based Single-Hop WDM Network: An Architecture for Efficient Multicasting

Martin Maier, Michael Scheutzow, and Martin Reisslein, *Member, IEEE*

***Abstract*—Research on multicasting in single-hop wavelength-division-multiplexing (WDM) networks has so far focused on networks based on the passive star coupler (PSC), a broadcast device. It has been shown that the multicasting performance is improved by partitioning multicast transmissions into multiple multicast copies. However, the channel bottleneck of the PSC, which does not allow for spatial wavelength reuse, restricts the multicast performance. In this paper, we investigate multicasting in a single-hop WDM network that is based on an arrayed-waveguide grating (AWG), a wavelength routing device that allows for spatial wavelength reuse. In our network, optical multicasting is enabled by wavelength-insensitive splitters that are attached to the AWG output ports. Multicasts are partitioned among the splitters and each multicast copy is routed to a different splitter by sending it on a different wavelength. We demonstrate that the spatial wavelength reuse in our network significantly improves the throughput-delay performance for multicast traffic. By means of analysis and simulations, we also demonstrate that for a typical mix of unicast and multicast traffic the throughput-delay performance is dramatically increased by transmitting multicast packets concurrently with control information in the reservation medium access control protocol of our AWG-based network.**

***Index Terms*—Arrayed-waveguide grating (AWG), multicasting, partitioning, reservation medium access control (MAC), single-hop wavelength-division-multiplexing (WDM) network, spatial wavelength reuse.**

I. Introduction

METROPOLITAN area (metro) wavelength-division-multiplexing (WDM) networks connect local access networks to the WDM backbone network. With the increasing speeds in access networks due to new technologies, such as Gigabit Ethernet, xDSL, and cable modems, and the deployment of very high-speed backbone networks, metro networks are emerging as a bottleneck in the Internet. This bottleneck—commonly referred to as metro gap—calls for the development of novel network architectures and protocols for metro WDM networks [1]. Indeed, several recent research efforts address the metro gap, see for instance [2]–[7] and [43]. These efforts are primarily directed at unicast (i.e., point-to-point) traffic. Multidestination (i.e., point-to-multipoint) traffic, however, is expected to account for a significant portion of the load on metro networks. This multidestination traffic load is due to emerging applications, such as teleconferences, multimedia stream distribution, telemedicine, and distributed games, and is further increased by the placement of content distribution proxies in metro networks. With multicasting, a source node reaches multiple destinations by sending a single multicast data packet, instead of sending multiple unicast packets. Thus, multicasting can significantly increase the efficient resource (transmitter, channel) utilization for multidestination traffic and can improve the cost effectiveness, which is critical for metro networks.

In this paper, we focus on single-hop WDM networks, where source and destination communicate directly with each other, without any traffic forwarding by intermediate nodes. Compared with multihop networks, single-hop networks have the minimum mean hop distance (unity) and do not waste any bandwidth for data forwarding. Thus, single-hop networks have the potential to provide a higher channel utilization and an improved throughput-delay performance compared with their multihop counterparts [8].

Since the mid-1990s, multicasting over single-hop WDM networks based on the passive star coupler (PSC) has received considerable interest [9]–[23]. With the emergence of the metro gap, multicasting over PSC-based single-hop WDM networks has received renewed interest [24]–[31]. A key problem with multicasting in PSC-based networks is that the larger the multicast size, the more difficult it is to find free receivers at all destination nodes, especially in heavy traffic. As a consequence, multicast transmissions have to be delayed or other transmissions have to be pre-empted, resulting in a decreased throughput-delay performance. To address this problem, the *partitioning* of a multicast transmission into several subgroups has been proposed [12]. Instead of sending a given multicast packet to all intended receivers at once, an improved throughput-delay performance is achieved by sending multiple copies of the multicast packet; each copy is received by a different destination subgroup. With partitioning each multicast copy requires a smaller number of receivers which are more likely to be free. Also, simultaneously, other transmitters can send multicast packets on different wavelengths to other free receivers. Thus, multiple wavelengths are used at any given time, resulting in

Manuscript received December 31, 2002; revised September 5, 2003. This work was supported in part by the Federal German Ministry of Education and Research within the TransiNet Project, in part by the DFG Research Center Mathematics for Key Technologies (FZT86), Berlin, Germany, and in part by the National Science Foundation under Grant Career ANI-0133252.

M. Maier is with the the Centre Tecnológic de Telecomunicacions de Catalunya (CTTC), Barcelona 08034, Spain.

M. Scheutzow is with the Department of Mathematics, Technical University Berlin, Berlin 10587, Germany (e-mail: ms@math.tu-berlin.de).

M. Reisslein is with the Department of Electrical Engineering, Arizona State University, Tempe, AZ 85287-5706 USA (e-mail: reisslein@asu.edu; web: http://www.fulton.asu.edu/~mre).

Digital Object Identifier 10.1109/JSAC.2003.819158

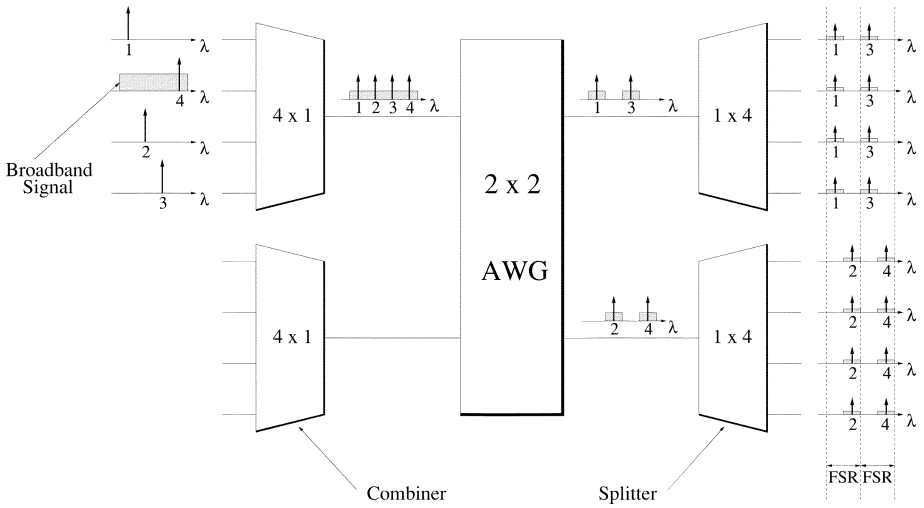


Fig. 1. Multicasting over an AWG with attached wavelength-insensitive combiners/splitters. A given transmission reaches all receivers at a given splitter. Multiple transmissions on different wavelengths reach different splitters.

an increased efficiency. The PSC, however, is a broadcast-and-select device. Thus, each multicast copy is distributed not only to the intended receiver subgroup but to all receivers, which wastes power and bandwidth. Recently, it was shown that partitioning suffers from a *channel bottleneck* in PSC-based single-hop WDM networks [29], [30]. This is due to the fact that partitioning requires more wavelengths. Since the PSC does not allow for spatial wavelength reuse, the number of available wavelength channels is limited and this prevents nodes from taking full advantage of the partitioning.

We investigate multicasting in a network that consists of an arrayed-waveguide grating (AWG) and wavelength-insensitive combiners and splitters attached to the AWG input and output ports. As opposed to the PSC, the AWG is a wavelength-routing device. Each multicast copy is routed to a different splitter by sending it on a different wavelength. Thus, a multicast is partitioned among the different splitters, and each multicast copy is received only by nodes which are attached to the respective splitter. The splitters are used to enable optical multicasting and are located at the network periphery. Each multicast packet is not duplicated until it reaches that splitter to which the corresponding receivers are attached. The same *wavelength* can be *spatially reused* in order to send other multicast packets to other splitters. Therefore, compared with the PSC, the AWG provides a higher degree of concurrency which in turn improves the throughput-delay performance of the system by means of partitioning and spatial wavelength reuse. To our knowledge this is the *first paper to investigate multicasting in a single-hop WDM network that is based on an AWG*.

This paper is organized as follows. In Section II, we discuss the physical properties of the AWG and the basic principles for multicasting over the AWG-based network. In Section III, we describe the AWG-based network architecture. In Section IV, we outline the employed medium access control (MAC) protocol. In Section V, we study the throughput-delay performance for multicasting in the proposed AWG-based network and compare it with the widely studied PSC-based networks. This section focuses exclusively on multicast traffic, i.e., all data packets are multicast packets. We reconfirm the benefits of *partitioning* the multicast transmissions and demonstrate that the AWG-based network with its *spatial wavelength reuse* has the potential to achieve significantly better multicast performance than the PSC-based networks. Next, in Section VI, we analyze the transmission of a typical mix of unicast and multicast traffic over the AWG-based network. For this typical traffic mix scenario, we examine the transmission of: 1) multicast packets *concurrently with spread control* information during the periodic reservation phase of our MAC protocol (thus increasing receiver utilization and multicast throughput) and 2) unicast packets with spatial wavelength reuse. We summarize our findings in Section VII.

II. Principles of Multicasting Over AWG-Based Network

In this section, we review the key properties of the AWG which enable efficient multicasting in a single-hop WDM network. In this paper, we consider a cyclic AWG whose free spectral range is equal to the number of ports (times the channel spacing, to be consistent with the units). Without loss of generality, we consider an AWG with degree $D = 2$, i.e., a 2×2 AWG, see Fig. 1. In this example, four wavelengths are launched into the upper input port of the AWG. Every second wavelength is routed to the same output port. This period of the wavelength response is called free spectral range (FSR). We use $R = 2$ FSRs of the underlying AWG, each consisting of two wavelengths. A wavelength-insensitive 4×1 combiner is attached to each AWG input port. Similarly, a 1×4 wavelength-insensitive splitter is attached to each AWG output port. Each splitter equally distributes all incoming wavelengths to all attached receivers, resulting in splitting loss. Similarly, each combiner suffers from combining loss, as illustrated in Fig. 1. Several approaches to compensate for these losses and other network feasibility issues are discussed in [32] and [44]. In brief, one possible solution is to place erbium-doped fiber amplifiers (EDFAs) between each combiner (splitter) and AWG input (output) port, respectively. The advantage of splitters is that they allow for efficient optical multicasting. For example, in Fig. 1, the transmitter tuned to wavelength 1 has to send one

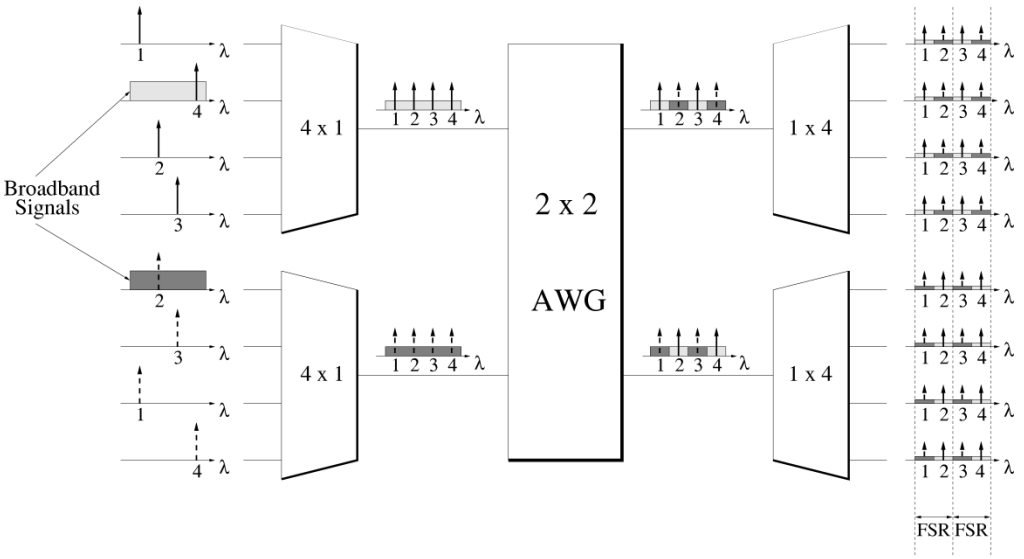


Fig. 2. Multicasting with spatial wavelength reuse. Multicast transmissions can simultaneously take place on all wavelengths at each AWG port without resulting in channel collisions.

single data packet in order to reach all receivers attached to the upper AWG output port. This conserves transmitter and channel resources. Note that if destination nodes of a given multicast group are attached to both AWG output ports via different splitters, the transmitter has to send the multicast data packet twice.

Note that each transmitter (receiver) has to be tunable over at least one FSR (consisting of D wavelengths) of the underlying AWG in order to send (receive) data to (from) all D AWG output (input) ports in one single hop. To fully exploit all R FSRs, the transmitters and receivers have to be tunable over $R \cdot D$ wavelengths. Throughout this paper, we consider transmitters and receivers with a tuning range of $R \cdot D$ wavelengths. As demonstrated in this paper, due to the extensive reuse of wavelengths and concurrent transmissions, our network achieves good performance for realistic tuning ranges, e.g., $R \cdot D = 8$.

As depicted in Fig. 1, a broadband light source signal, e.g., light emitting diode (LED), is fed into the upper AWG input port in addition to the four wavelengths. The broadband signal is spectrally sliced by the AWG such that a slice of the original signal (albeit attenuated) is routed to each receiver, irrespective at which splitter it is located. The broadband light source is used for the periodic broadcasting of control traffic (reservation requests for data packets) in our AWG-based network. Our MAC protocol (see Section IV) ensures that each receiver is periodically tuned to one of the control traffic slices to avoid receiver collisions for the control traffic. (In WDM networks, there are two types of collision: channel collision and receiver collision. A channel collision occurs when two or more nodes simultaneously access the same channel. Thus, in Fig. 1 a channel collision occurs when two or more transmitters that connect to the same combiner send control packets at the same time (control traffic channel collision) or when they simultaneously send data using the same wavelength (data channel collision). A receiver collision, also known as destination conflict, is said to occur when two or more signals arrive simultaneously on different wavelengths at a given destination node, but not all of them can be received. This may happen because either the destination node is not equipped with a sufficient number of receivers or the destination node's receivers are not tuned to the wavelengths of the arriving signals. Therefore, a packet may get lost at the receiver even though the transmission was absolutely channel collision free.) Using the control information each node acquires and maintains global knowledge. This knowledge is used for distributed scheduling without explicit acknowledgment, which results in improved channel utilization and decreased delay.

A. Multicasting With Spatial Wavelength Reuse

As opposed to the PSC, the AWG allows for spatial wavelength reuse at all ports. Fig. 2 illustrates that all four wavelengths and an additional broadband signal can be applied at both combiners simultaneously without resulting in collisions at the splitter output ports. (Note that data wavelengths and control slices overlap spectrally at the splitter output ports. In Section III, we discuss how destination nodes attached to the splitter output ports are able to separate the spectrally overlapping data and control signals at the receiver side.) Generally, with a $D \times D$ AWG each wavelength can be spatially reused D times. On one hand, for a given transceiver tuning range of $R \cdot D$ wavelengths, a small D implies that more FSRs R are used but also that a given multicast packet is received by more nodes attached to the corresponding splitter. On the other hand, a large D increases spatial wavelength reuse, reduces the number of used FSRs R, and partitions a multicast into smaller subgroups. This tradeoff between spatial wavelength reuse, using multiple FSRs and efficient multicasting with partitioning is further investigated in Section V of this paper.

B. Multicasting Concurrently With Control

For increased efficiency the control traffic is transmitted using spreading techniques [33], as discussed in more detail shortly. The spreading allows for the simultaneous transmission of control traffic and data traffic. The spread transmission of the control traffic in conjunction with the periodic tuning of the receivers to the control traffic slices provides opportunities for

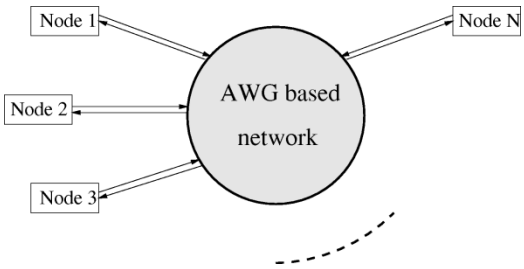


Fig. 3. Network architecture.

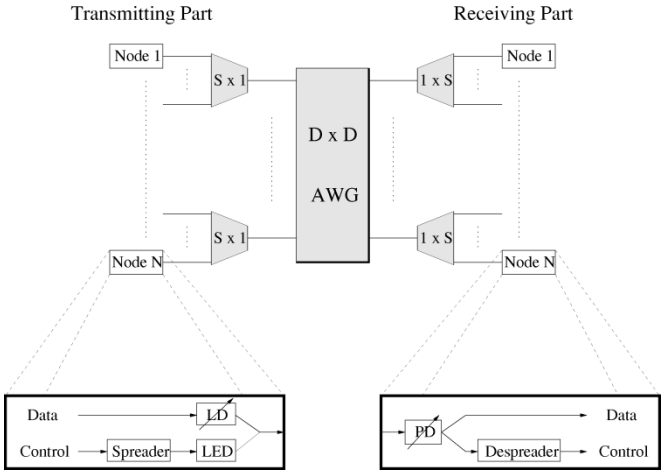


Fig. 4. Detailed network and node architecture.

efficient multicasting. As studied in detail in Section VI, transmitting multicast data traffic simultaneously with control traffic is one of the key techniques for efficiently accommodating a typical mix of unicast and multicast traffic in the AWG-based network.

III. AWG Based Network Architecture

Fig. 3 schematically shows the studied AWG-based single-hop network architecture. (We focus here primarily on the architectural features relevant for multicasting and refer the interested reader for more details to [32] and [44], where the network is studied for unicast transmissions.) There are N nodes, each attached to the network via two fibers. Every node uses one fiber for transmission and the other fiber for reception. The network and node architecture is depicted in more detail in Fig. 4. The network is based on a $D \times D$ AWG. A wavelength-insensitive $S \times 1$ combiner is attached to each AWG input port and a wavelength-insensitive $1 \times S$ splitter is attached to each AWG output port. The network, thus, connects $N = D \cdot S$ nodes. Each node contains a laser diode (LD) and a photodiode (PD) for data transmission and reception, respectively. Both LD and PD are tunable over $R \cdot D$ wavelengths. Fast tunable transmitters have been proven to be feasible in a cost-effective manner in [5]. Similarly, electrooptic tunable filters (EOTFs) [34] are promising candidates for realizing fast tunable receivers, which are expected not to be significantly more expensive than their fixed-tuned counterparts. We note that fast tunable filters are currently less mature than fast tunable lasers. However, we expect that the development of fast tunable filters will attract more attention in the future since using tunable receivers not only improves the network efficiency and performance by means of load balancing over all wavelengths [35], but also enables efficient multicasting. This is because all intended receivers can be tuned to the corresponding wavelength the multicast packet is transmitted on (which is the main topic of this paper).

In addition, each node uses a broadband light source, e.g., an off-the-shelf LED, for broadcasting control packets. The broadband LED signal (10–100 nm) is spectrally sliced such that all receivers obtain the control information. The signaling is done in-band, i.e., LED and LD signals overlap spectrally. In order to distinguish data and control information, we employ direct sequence spread spectrum techniques; at the transmitting part the control information is spread before externally modulating the LED (for a feasibility study of this concept the interested reader is referred to [33]). At the receiving part, the control information is retrieved by despreading a part of the incoming signal. By using multiple spreading codes, several nodes could transmit their control packets at the same time, leading to code division multiple access (CDMA). In this paper, we employ only one single code, just to enable the simultaneous transmission of data and control signals. This keeps the computational overhead at the nodes low, thus ensuring network scalability [36].

IV. Medium Access Control (MAC) Protocol

In this section, we discuss the MAC protocol, which controls the access of the tunable transceivers to the shared wavelengths.

A. Timing Structure

The timing structure of our MAC protocol is schematically shown in Fig. 5. As illustrated, R adjacent FSRs are exploited at each AWG port. Each FSR consists of D contiguous channels, where D denotes the physical degree of the underlying AWG. Time is divided into *cycles* which repeat periodically. Each cycle is further subdivided into D *frames*. The frame format of one wavelength is depicted in Fig. 6. A frame contains $F \in \mathbb{N}$ slots. The slot length is equal to the transmission time of a control packet (which is discussed shortly). The transceiver tuning time is assumed to be negligible, which is a realistic assumption for electrooptic transceivers with a tuning time of a few nanoseconds (and a small tuning range, e.g., $D \cdot R = 8$ wavelengths). Each frame is partitioned into the first M, $1 \leq M < F$, slots (shaded region) and the remaining $(F - M)$ slots. In the first M slots, control packets are transmitted and all nodes tune their receivers to one of the corresponding LED slices (channels) in order to obtain the control information. In each frame within a cycle, the nodes attached to a different AWG input port send their control packets. Specifically, all S nodes attached to AWG input port o (via a common combiner) send their control packets in frame o, $1 \leq o \leq D$ (see Fig. 5). (In order to allow for sufficient control throughput, M should be chosen larger for increasing S and *vice versa*.) Hence, after D frames (one cycle) all nodes have equally had the opportunity to send their control packets, ensuring fairness. To make the entire system scalable, the M slots are not fixed assigned. Instead, control packets are sent on a contention basis using a modified version of slotted ALOHA (we deploy a version of reservation ALOHA (R-ALOHA), for details please refer to [32],

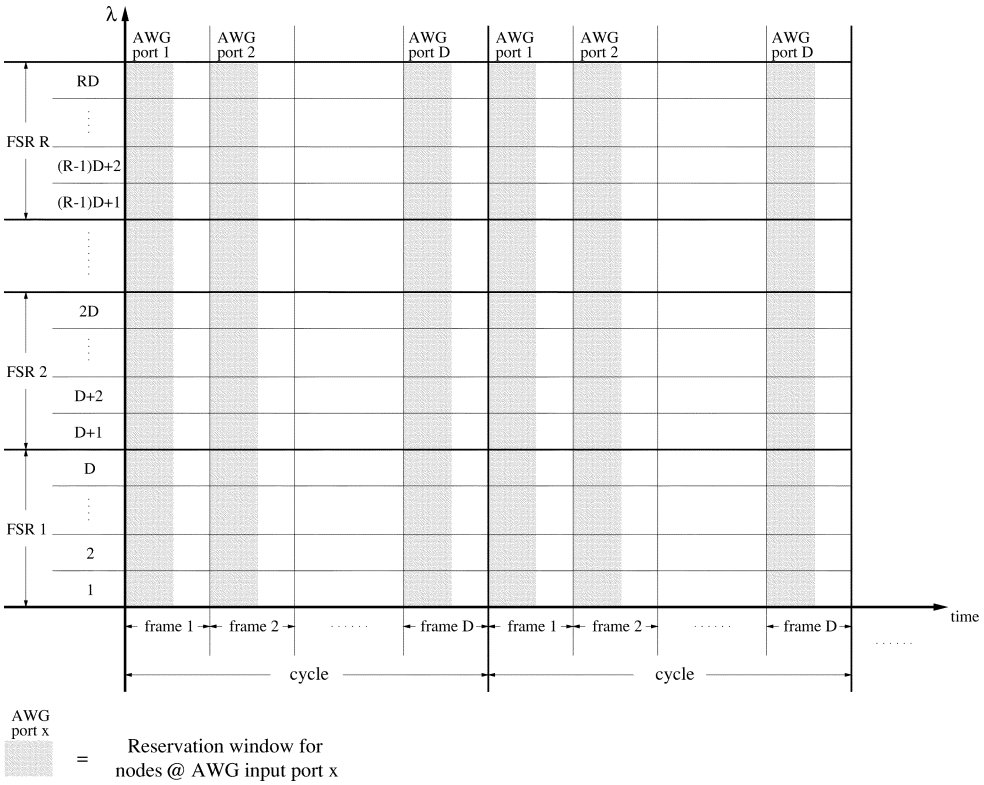


Fig. 5. Wavelength assignment at a given AWG port.

[44]). Control packets arrive at the receivers after the one-way end-to-end propagation delay (i.e., half the round-trip time).

In the last $(F-M)$ slots of each frame no control packets are sent, allowing receivers to be tuned to any arbitrary wavelength. This freedom enables transmissions between any pair of nodes. The parameter M trades off two types of concurrency. During the first M slots of a given frame o, control and data packets can be transmitted simultaneously, but only from nodes attached to AWG input port o. In this time interval, packets originating from other AWG input ports cannot be received. During the last $(F-M)$ slots of each frame, on the other hand, all receivers are unlocked and can be tuned to any arbitrary wavelength. During this time interval, data packets from any AWG input port can be received, thus allowing for data packet transmissions with spatial wavelength reuse (provided the data packet is no longer than $(F-M)$ slots).

B. Control Packet Transmission

If a node has no data packet in its buffer the LED and LD remain idle. When a data packet arrives at node i, $1 \leq i \leq N$, node i's LED broadcasts a control packet in one of the M slots of the frame allocated to the AWG input port that node i is attached to. The slot is chosen randomly according to a uniform distribution. A control packet consists of four fields, namely, destination address, length, and type of the corresponding data packet, and parity bits from error detection and correction coding. In the case of a unicast packet, the destination address is the address of the (single) destination node. In the case of a multicast packet, the destination address might consist of N bits where each bit represents a specific destination node (and a bit is set to one if the corresponding node belongs to the multicast group, otherwise, the bit is set to zero), or several bits denoting the corresponding multicast group. As illustrated in Fig. 6, the data packet can be of variable size L, $1 \leq L \leq F$, where L denotes the length in slots. The type field contains one bit and is used to enable packet and circuit switching. While we focus on packet-switched unicasting and multicasting in this paper, our protocol extends to circuit switching in a straightforward fashion; see [32] and [44] for a discussion on the circuit switching aspects of our protocol. The error detection and correction coding is used by the receiver to detect and correct sporadic bit errors in the control packet, which due to the generally extremely small bit error rates of optical systems and the passive nature of our single-hop network are very rare. A large number of bit errors that cannot be corrected is almost surely due to colliding control packets and is interpreted as such by the source nodes (which retransmit the control packets) and all the other nodes (which ignore the collided control packets). Techniques similar to [37] can be used for the error detection and correction coding in our distributed MAC protocol [38].

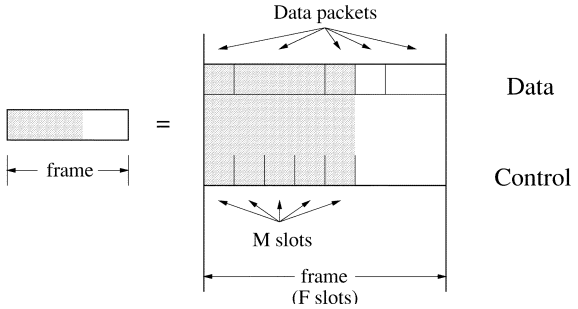


Fig. 6. Frame format.

C. Control Packet Reception and Data Packet Scheduling

Every node collects all control packets by tuning its receiver to one of the corresponding channels during the first M slots of each frame. Thus, it learns about all other nodes' activities and whether its own control packet was successful or not. If its control packet has collided, node i retransmits the control packet in the next cycle with probability p and with probability $(1-p)$ it defers the transmission by one cycle. The node retransmits the control packet in this next cycle with probability p, and so forth. Successful control packets are put in a distributed queue at each node.

All nodes process the control packets successfully received in the first M slots of a frame by executing the same scheduling algorithm. We note that this distributed scheduling requires that all nodes receive the (uncollided) control packets correctly, which is ensured by the error detection and control coding as described in Section IV-B. We also note that the distributed MAC protocol may be affected by malfunctioning and failing nodes, which has to be addressed by higher layer protocols and is beyond the scope of this paper.

For the scheduling we employ the first-come-first-served and first-fit scheduling algorithm, i.e., the data packets are scheduled in the first possible slots on the lowest available wavelength. We adopt this simple greedy scheduling algorithm since in high-speed networks arbitration algorithms need to be of low complexity [16]. A multicast packet with Δ, $1 \leq \Delta \leq D$, destination splitters is scheduled Δ times (each time for transmission on a different wavelength to a different subgroup of nodes). Note that these multiple transmissions require only one single control packet, resulting in a decreased signaling overhead [39].

As discussed in more detail in Sections V and VI, we consider two different variations of the first-come-first-served first-fit scheduling algorithm. For the multicast-only traffic scenario considered in Section V, multicast packets from nodes attached to AWG input port o, $1 \leq o \leq D$, are scheduled in all F slots of frame o of a cycle (i.e., during the frame in which the source node sends control packets), as well as in the last $F-M$ slots of the other $D-1$ frames of the cycle.

In the unicast and multicast traffic mix scenario considered in Section VI, multicast packets from nodes attached to AWG input port o are only scheduled in the F slots of frame o. In doing so, each transmitted multicast data packet benefits from the fact that all receivers are listening to the respective wavelength (at least for the first M slots of that frame), alleviating the receiver availability problem and resulting in a high receiver utilization. Unicast data packets are scheduled in the aforementioned F slots, as well as in the last $(F-M)$ slots of the remaining $D-1$ frames of a cycle, thus capitalizing on spatial wavelength reuse.

If there are not enough slots available within the scheduling window (to be defined shortly) the data packets are not transmitted and the corresponding source nodes have to retransmit the control packets in the next cycle. (Nodes which lose the scheduling are aware of this because all nodes have global knowledge and execute the same distributed scheduling algorithm.)

The length of the scheduling window is set to D cycles for two reasons. First, it takes up to D cycles to transmit a multicast packet of length $L=F$. To see this, note that in the adopted timing structure a given node can send a packet of length $L=F$ (no matter whether unicast or multicast) only once in a cycle. This is because the receivers have to tune to the LED slice with the control information for the first M slots of every frame. A given node sends its control packets once per cycle, and only then is the transmitter able to reach the desired receiver(s) for F consecutive slots. Since the intended receivers of a multicast packet can be located at all D splitters the multicast packet may have to be transmitted D times. Therefore, the scheduling window has to be at least D cycles long. Second, setting the scheduling window to the minimum number of cycles ensures that a node needs to maintain a scheduling table of no more than D cycles into the future. This keeps the computational overhead at each node small, which is of paramount importance in very high-speed optical networks.

V. Multicast With Partitioning and Spatial Wavelength Reuse

In this section, we study the transmission of multicast packet traffic over the AWG-based network. We compare the throughput-delay performance of the AWG-based network with the extensively studied PSC-based network. For this study, we consider only multicast traffic, i.e., each packet is destined to a multicast group. The size of the multicast group, i.e., the number of destination nodes, and the members of a given multicast group are independently randomly drawn for each packet. The multicast group size is uniformly distributed over [1, $N-1$] nodes and the multicast group members are uniformly distributed over all network nodes [1, N] except the transmitting source node, as is typically considered in multicast studies. The destination nodes of a given multicast packet are persistent, i.e., are not renewed when the corresponding control packet fails and is retransmitted. A given packet is *long* (occupies F slots) with probability q, $0 \leq q \leq 1$, and is *short* (occupies $F-M$ slots) with the complementary probability $1-q$. Recall that on the AWG a long multicast packet from a node attached to AWG input port o, $1 \leq o \leq D$, can only be scheduled in frame o of a given cycle, i.e., in the frame in which the node sends control packets. Short multicast packets from a node at port o are scheduled in frame o, as well as the other $(D-1)$ frames of a given cycle according to the adopted first-come-first-served first-fit scheduling discipline.

We consider the following commonly studied performance metrics.

- *Mean transmitter throughput* defined as the mean number of transmitting nodes in steady state.
- *Mean multicast throughput* defined as the mean number of multicast completions per frame. (Multicast throughput is equal to the ratio of mean transmitter throughput and mean number of required transmissions in order to reach all receivers of a given multicast packet. Thus, multicast throughput measures the multicast efficiency of each packet transmission.)
- *Mean receiver throughput* defined as the mean number of receiving nodes in steady state.

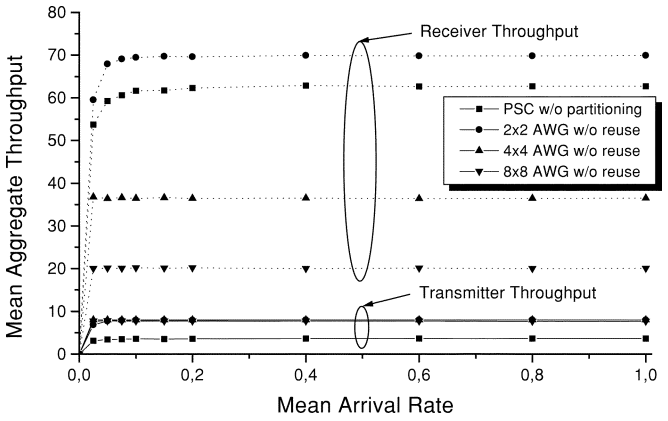


Fig. 7. Mean transmitter and receiver throughput versus mean arrival rate (packet/frame) for PSC (without partitioning) and AWG-based single-hop networks without spatial wavelength reuse.

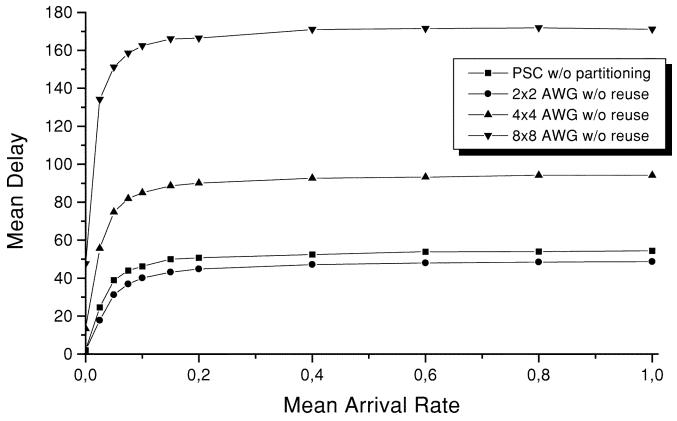


Fig. 8. Mean delay (frames) versus mean arrival rate (packet/frame) for PSC (without partitioning) and AWG-based single-hop networks without spatial wavelength reuse.

- *Mean delay* defined as the average time in frames from the generation of a packet until the completion of the multicast transmission.

For the simulations in this section, the network parameters are set to the following default values: Number of nodes $N = 200$, the transceiver tuning range $D \cdot R = 8$ remains constant for varying D and R, retransmission probability $p = 0.5$, number of slots per frame $F = 200$, number of reservation slots per frame $M = 30$, and scheduling window size of 64 frames. (The scheduling window is set to 64 frames to ensure a fair comparison of the considered network configurations with $D = 2, 4,$ and 8, of which the $D = 8$ configuration requires the largest scheduling window of eight cycles, which translates into 64 frames.) The propagation delay is assumed to be no larger than one frame. (The propagation delay is assumed to be no larger than one frame since in the PSC-based network there is no cyclic timing structure as opposed to the AWG based one. In the PSC-based network, each node is assumed to be able to (re)transmit a control packet in every frame.) We assume that all nodes are equidistant from the central AWG (PSC), which can be achieved in practice with standard low-loss fiber delay lines or by implementing electrical delays at the nodes. (Note that the appropriate delays must be introduced both for the transmitted signal leaving a node, as well as the incoming signal being received at the node to ensure that the signals from all nodes line up at the central AWG (PSC), as well as for the distributed scheduling.) The mean arrival rate denotes the probability that an idle node generates a multicast packet at the end of a frame. Each simulation was run for 10^6 slots including a warm-up phase of 10^5 slots. The width of the 98% confidence intervals obtained with the method of batch means was always smaller than 5% of the corresponding sample means.

A. Simulation Results

In Figs. 7 and 8, we set $q = 1.0$, i.e., we consider only long packets ($L = F = 200$ slots) which cannot benefit from spatial wavelength reuse in the AWG-based network. We compare the throughput-delay performance of a $D \times D$ AWG with $D \in \{2, 4, 8\}$ and a PSC-based single-hop WDM network. For a fair comparison in both networks each node is equipped with the same pair of one tunable transmitter and one tunable receiver for data transmission. In the PSC-based network, control is broadcast by using the inherent broadcast nature of the PSC. Each node is equipped with an additional transceiver fixed tuned to a separate wavelength. Thus, in the PSC-based network there are nine wavelengths, eight for data and one for control transmission. Nodes ready to (re)transmit control packets are allowed to randomly access $M = 30$ reservation slots in each frame of length $F = 200$ slots, using the same retransmission probability $p = 0.5$ as in the AWG-based counterpart. (Note that we could allow the transmission of control packets throughout the entire frame on the extra control wavelength in the PSC-based network. This would improve the throughput-delay performance if the slotted ALOHA control packet contention were a bottleneck. This bottleneck would result in a deteriorating throughput-delay performance for increasing arrival rates, as demonstrated in [40]. As we observe from Figs. 7 and 8, however, the throughput and delay are stable for increasing arrival rates, indicating that the control packet contention is not a significant bottleneck.) Figs. 7–11 are obtained by setting the mean arrival rate to {0.0001, 0.025, 0.05, 0.075, 0.1, 0.15, 0.2, 0.4, 0.6, 0.8, 1.0}.

Fig. 7 depicts the mean transmitter and mean receiver throughput versus mean arrival rate. We observe that the transmitter throughput in the AWG-based network is about twice as large as in the PSC-based network, where the latter one is assumed to operate without partitioning (the case where the PSC supports *logical* partitioning is discussed shortly). This is because due to its wavelength-routing nature the AWG provides (physical) partitioning such that nodes ready to send multicast packets are more likely to find free destination receivers for transmitting the corresponding multicast packets. Note that for all $D \in \{2, 4, 8\}$ the AWG provides the same transmitter throughput of eight. This is due to the fact that with a fixed transceiver tuning range of $D \cdot R = 8$ the number of available wavelength channels is limited such that additional transmissions cannot take place even though the corresponding destination receivers might be free. Hence, this figure confirms that partitioning can cause a channel bottleneck in the network. This channel bottleneck can be alleviated by spatial wavelength reuse, as discussed shortly. However, the physical AWG degree

D has an impact on the receiver throughput, as depicted in Fig. 7. While a 2×2 AWG yields a larger receiver throughput than the PSC, for $D \in \{4, 8\}$ we observe the opposite. This is due to the channel bottleneck caused by partitioning. To see this, recall that for $D \in \{2, 4, 8\}$ the number of transmitting nodes is equal to the maximum number of available wavelength channels. For increasing D fewer nodes are attached to the same splitter. Consequently, each transmitted multicast copy is received by a smaller number of destination nodes, resulting in a decreased receiver throughput. (We do not show multicast throughput here since in the PSC-based network without partitioning multicast and transmitter throughput are the same.)

Fig. 8 depicts the mean delay versus mean arrival rate for the PSC and $D \times D$ AWG-based single-hop WDM networks, where $D \in \{2, 4, 8\}$. Only the 2×2 AWG provides a smaller delay than the PSC. This is because in a 2×2 AWG-based network with partitioning more nodes can transmit simultaneously than in the PSC-based counterpart leading to a smaller delay. Whereas for $D \in \{4, 8\}$ the delay is significantly larger, since for increasing D multicast packets have to be sent to more splitters. Each of those multicast copies is transmitted in a separate cycle, each consisting of D frames. Therefore, with increasing D not only the average number of required multicast copies but also the cycle length increases, resulting in a larger delay.

Concluding the discussion of the results in Figs. 7 and 8, we note that the $D \times D$ AWG without spatial wavelength reuse is essentially equivalent to a PSC with a (fixed) partitioning of the receivers into D groups. Thus, the results in Figs. 7 and 8 provide also insights into the performance of multicasting over the PSC with partitioning. We observe that the PSC with two partitions outperforms the PSC without any partitions in terms of throughput and delay. We also observe that four or more partitions result in a significantly increased delay, reconfirming the results in [30]. In conclusion, the detrimental effect of partitioning kicks in for extensive partitioning. For moderate partitioning into two groups, on the other hand, the throughput-delay performance is improved, not worsened. Therefore, for all following comparisons with the AWG, we use the PSC with two partitions, i.e., the receivers in the PSC-based network are divided into two groups comprising nodes 1 through $\lfloor N/2 \rfloor$ and $\lfloor N/2 \rfloor + 1$ through N, respectively.

In Figs. 9–11, we investigate the impact of spatial wavelength reuse on the transmitter, receiver, and multicast throughput-delay performance of the AWG-based single-hop WDM network and compare with PSC-based networks. For this purpose, we set $q = 0$, i.e., all packets have a length of $F - M = 170$ slots which can be transmitted by spatially reusing all wavelengths. Fig. 9 illustrates the mean delay versus mean transmitter throughput of both PSC and AWG-based networks. Note that compared with Fig. 7 the maximum transmitter throughput of the AWG-based network without spatial wavelength reuse is smaller than eight, since frames are not fully utilized due to the smaller packet size of $F - M$ slots. We observe that by allowing for spatial wavelength reuse the transmitter throughput-delay performance of all $D \times D$ AWG-based networks is significantly improved with $D \in \{2, 4, 8\}$. Note that for $D = 2$ nodes cannot fully capitalize on the increased number of available wavelength channels. This is because with

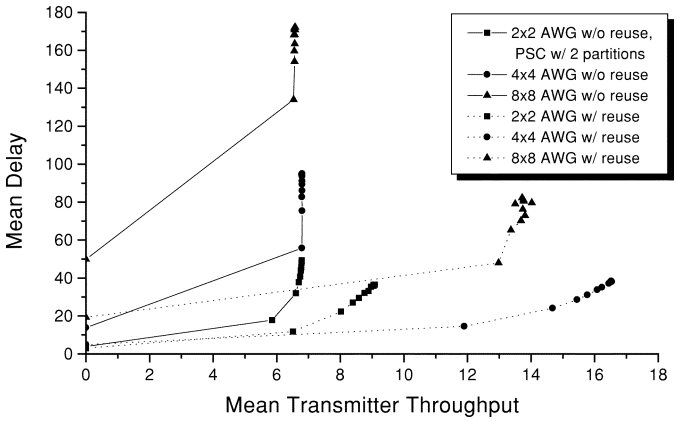


Fig. 9. Mean delay (frames) versus mean transmitter throughput for PSC (with two partitions) and AWG-based single-hop networks with and without spatial wavelength reuse.

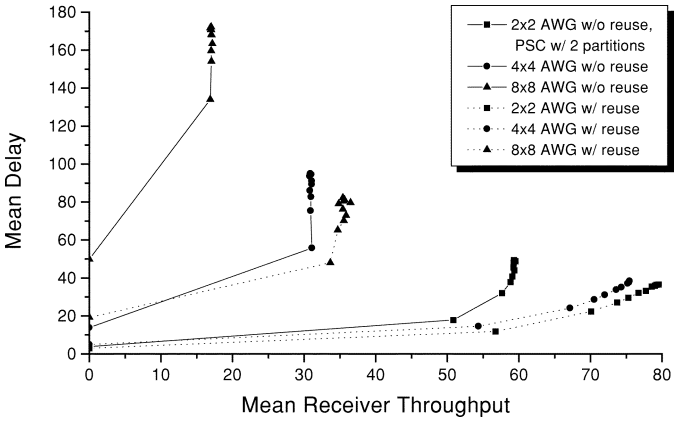


Fig. 10. Mean delay (frames) versus mean receiver throughput for PSC (with two partitions) and AWG-based single-hop networks with and without spatial wavelength reuse.

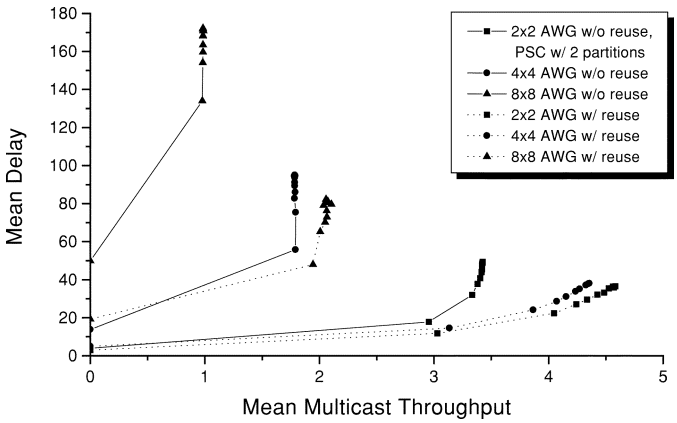


Fig. 11. Mean delay (frames) versus mean multicast throughput for PSC (with two partitions) and AWG-based single-hop networks with and without spatial wavelength reuse.

two partitions multicast copies destined to the same splitter are likely to experience receiver conflicts since on average each multicast copy is destined to more receivers for $D = 2$ than $D \in \{4, 8\}$. As a result, there are many destination conflicting multicast transmissions resulting in a modest transmitter throughput. The problem of destination conflicts is mitigated by dividing the receivers into more partitions. For $D = 4$ more

transmitters are likely to find the corresponding receivers free resulting in a transmitter throughput, which is more than twice as large as the one of a 4×4 AWG-based network without spatial wavelength reuse. Further increasing the number of partitions to $D = 8$ reduces the throughput, which appears to be due to the smaller number of wavelength channels $R = 1$ connecting each individual AWG input-output port pair for the fixed transceiver tuning range $R \cdot D = 8$. Overall, we find that with spatial wavelength reuse a 4×4 AWG-based network provides the smallest delay and the largest transmitter throughput which is more than twice that of a PSC-based network, which operates with two partitions but does not allow for spatial wavelength reuse.

Figs. 10 and 11 show that spatial wavelength reuse also significantly improves the receiver and multicast throughput-delay performance of AWG-based single-hop networks. Again, $D = 8$ is not a good choice to achieve an acceptable network performance, whereas, $D = 4$ and $D = 2$ exhibit about the same receiver and multicast throughput-delay performance improvement. In terms of multicast throughput, i.e., the mean rate of multicast completions, it is advisable to set $D = 2$. That is, with $D = 2$ the transmitter throughput is rather small (see Fig. 9) but each transmitted multicast copy is received by more intended destinations attached to the same splitter translating into an increased receiver throughput (see Fig. 10) and fewer required transmissions of a given multicast packet. Note that in terms of receiver and multicast throughput a 2×2 AWG-based single-hop network outperforms its PSC-based counterpart by approximately 30%, where the latter one deploys the same partitioning but is unable to provide spatial wavelength reuse.

VI. Multicasting Simultaneously With Control

Up to this point, we have considered only multicast packet traffic, i.e., each packet was destined to a random number of $1, \ldots, N-1$ nodes and we have examined the interplay between partitioning and spatial wavelength reuse. In contrast, in this section, we analyze the transmission of a typical unicast and multicast traffic mix over the AWG-based network. In this traffic mix a certain portion of the traffic is unicast while the remaining traffic is multicast. We focus on the interplay between unicast with spatial wavelength reuse and multicast concurrently with control traffic; we do not consider partitioning in this section. The motivation for this study is as follows. The results of the preceding section demonstrate that spatial wavelength reuse is beneficial for transmitting multicast traffic. Spatial wavelength reuse is not possible during the reservation phase, i.e., the first M slots of every frame when the control packets are transmitted. Thus, the reservation phase prevents the full exploitation of spatial wavelength reuse. Now consider the transmission of a typical mix of unicast and multicast traffic. For unicast traffic the wavelength channels are the primary bottleneck and receiver availability is typically not a problem; hence, spatial wavelength reuse (which alleviates the channel bottleneck) brings dramatic benefit for unicast traffic [2], [43]. Multicast traffic also benefits from spatial wavelength reuse, but typically receiver availability is its primary bottleneck. This suggests to schedule: 1) multicast packets in the frame with the reservation phase, during which all receivers are tuned to a slice carrying the spread control traffic, thus alleviating the receiver availability problem and 2) unicast packets in the remaining frames, where they can exploit spatial wavelength reuse. In this section, we develop an analytical model to study the interplay between unicast and multicast traffic. We examine how spatial wavelength reuse and multicasting concurrently with control improve the overall throughput-delay performance of the AWG-based network.

A. System and Traffic Model

We conduct an asymptotic analysis, which is exact in the asymptotic limit $S \to \infty$ and gives good accuracy for finite S, as verified by simulations. Throughout our analysis, we assume that the propagation delay is no larger than one cycle, which is reasonable for metro networks. All nodes are equidistant from the AWG. We let the mean arrival rate σ denote the probability that an idle node at AWG input port o generates a new packet right before the beginning of frame o of a cycle. Similarly to Section V, a packet is either *long* (occupies F slots) or is *short* (occupies K slots, with $1 \leq K \leq F - M$). Additionally, a packet is either a *unicast* packet (destined to *one* node) or a *multicast* packet (destined to *all* nodes attached to one splitter). A unicast packet is destined to any of the N nodes (including the sending node, for simplicity) with equal probability $1/N$. A multicast packet is destined to any of the D splitters (including the splitter that the sending node is attached to) with equal probability $1/D$. (We note that assuming that a given multicast packet is destined to all nodes attached to a given splitter helps assess the maximum achievable receiver utilization that can be achieved by the passive optical splitters that locally broadcast each packet to all attached nodes. In the more general case, where a locally broadcast packet is not destined to all attached receivers, we expect a smaller receiver throughput. On the other hand, the transmitter throughput and multicast throughput are expected to increase and the mean delay is expected to decrease since receivers are more likely to be free, allowing for more simultaneous multicasts. A quantitative analysis of this more general case is an interesting avenue for future work.)

As a shorthand, we refer to the four packet types (long, multicast), (long, unicast), (short, multicast), and (short, unicast) with the tuples (l, a), $(l, 1)$, (s, a), and $(s, 1)$. Let $p_{l,a}$, $p_{l,1}$, $p_{s,a}$, and $p_{s,1}$ denote the probabilities that a newly generated packet is of type (l, a), $(l, 1)$, (s, a), or $(s, 1)$. Note that $p_{l,a} + p_{l,1} + p_{s,a} + p_{s,1} = 1$. If a control packet fails (either in the control packet contention or the data packet scheduling) the type of the corresponding data packet is not changed in our model, i.e., the packet type is persistent. However, we do assume nonpersistency [41] for the destination in our model, i.e., a new random destination (node or splitter) is drawn for each attempt to transmit a control packet.

Now, consider the nodes attached to a given (fixed) AWG input port o, $1 \leq o \leq D$. These nodes send their control packets in frame o of a given cycle. We refer to the nodes that at the beginning of frame o hold an old packet, that is, a control packet that has failed in control packet contention or data packet scheduling, as "*backlogged*." We refer to all the other nodes as "*idle*." Let η be a random variable denoting the number of idle nodes at AWG input port o. Let $\hat{p}_{l,a}$, $\hat{p}_{l,1}$, $\hat{p}_{s,a}$, and $\hat{p}_{s,1}$ denote the probabilities that a given node at port o is to send a control

packet corresponding to a data packet of type (l, a), (l, 1), (s, a), or (s, 1) next. Again, note that $\hat{p}_{l,a} + \hat{p}_{l,1} + \hat{p}_{s,a} + \hat{p}_{s,1} = 1$. We expect, for instance, that $\hat{p}_{l,a}$ is larger than $p_{l,a}$ since long multicast packets are more difficult to schedule than the other packet types and, thus, require more retransmissions (of control packets).

B. Analysis of Control Packet Contention

First, we calculate the probability κ that a given control slot out of the available M control slots in frame o contains a successful control packet. A given control slot contains a successfully transmitted control packet if either: 1) it contains exactly one control packet corresponding to a newly generated data packet (from one of the idle nodes) and no control packet from the backlogged nodes or 2) it contains exactly one control packet from a backlogged node and no control packet from an idle node. Hence

$$\kappa = \eta \frac{\sigma}{M}\left(1-\frac{\sigma}{M}\right)^{\eta-1}\left(1-\frac{p}{M}\right)^{S-\eta} + (S-\eta)\frac{p}{M}\left(1-\frac{p}{M}\right)^{S-\eta-1}\left(1-\frac{\sigma}{M}\right)^{\eta} \quad (1)$$

where we assume that the number of control packets from idle nodes is independent of the number of control packets from backlogged nodes, which as our simulations indicate is reasonable.

Recall from our traffic model in Section VI-A that each packet is destined to any one of the D AWG output ports with equal probability $1/D$. Thus, the number of control packets corresponding to (l, a), (l, 1), (s, a), (s, 1) data packets that: 1) originate from a given AWG input port o, $o = 1, \ldots, D$; 2) are successful in the control packet contention of frame o (of a given cycle); and 3) are destined to a given AWG output port d, $d = 1, \ldots, D$, are distributed according to the binomial distributions $BIN(M, \kappa\hat{p}_{l,a}/D)$, $BIN(M, \kappa\hat{p}_{l,1}/D)$, $BIN(M, \kappa\hat{p}_{s,a}/D)$, and $BIN(M, \kappa\hat{p}_{s,1}/D)$, respectively.

C. Analysis of Packet Scheduling

We now proceed to calculate the numbers of successfully scheduled packets. Recall that the numbers of packets to be considered for the schedule from a given AWG input port to a given AWG output port are distributed according to the binomial distributions given at the end of the preceding section. Let $X_{l,a}$, $X_{l,1}$, $X_{s,a}$, and $X_{s,1}$ be random variables denoting the number of packets of type (l, a), (l, 1), (s, a), and (s, 1) that: 1) originate from a given AWG input port o, $o = 1, \ldots, D$; 2) are successful in the control packet contention of frame o (of a given cycle); 3) are destined to a given AWG output port d, $d = 1, \ldots, D$; and 4) are successfully scheduled within the scheduling window of one cycle. We calculate $E[X_{l,a}]$, $E[X_{l,1}]$, $E[X_{s,a}]$, and $E[X_{s,1}]$ as functions of $\hat{p}_{l,a}$, $\hat{p}_{l,1}$, $\hat{p}_{s,a}$, $\hat{p}_{s,1}$, and κ [which in turn is a function of η as given in (1)].

The two critical resources (constraints) for the data packet scheduling are: 1) the wavelength channels on the AWG and 2) the tunable receiver at each of the nodes. Recall from Section IV that data packets are scheduled so as to avoid channel collisions (i.e., two or more packets being transmitted on the same wavelength channel at the same time) and receiver collisions (i.e., two or more packets being destined to the same receiver at the same time).

1) Channel Constraint: First, we examine the wavelength channel constraint. Consider the scheduling of packets from a given (fixed) AWG input port o to a given (fixed) AWG output port d over the scheduling window of D frames. Over this scheduling window the AWG provides R parallel wavelength channels during the long (F slot) transmission slot, i.e., during the frame in which the nodes at port o send their control packets. During each of the remaining $(D-1)$ frames, the AWG provides a short ($F-M$ slot) transmission slot; each again with R parallel wavelength channels.

Now, we consider the scheduling of the four different types of packets in these transmission slots. First, we consider each packet type in isolation. Clearly, we can schedule at most one (l, a)-packet during the scheduling window. To see this, note that a long packet can only be scheduled during the long transmission slot. Also, a multicast will occupy all receivers at the considered destination port d during the transmission slot. Formally, we let a, $a = 0$, 1, denote the number of scheduled (l, a)-packets.

Next, we consider (l, 1)-packets and let b denote the number of scheduled (l, 1)-packets. Long packets can again only be scheduled during the long transmission slot. For unicast packets, we ignore receiver collision, as their impact is typically small [2], [43]. Hence, $0 \le b \le R$.

Packets of type (s, a) could be scheduled in the long transmission slot, as well as in the short transmission slots. To examine the effect of multicasting concurrently with control, we schedule (s, a)-packets only in the long transmission slot and let c denote the number of scheduled (s, a)-packets. Note that an (s, a)-packet occupies all receivers at the considered destination splitter for a duration of K slots. Hence, $0 \le c \le \lfloor F/K \rfloor$.

Finally, note that (s, 1)-packets can be scheduled in both the long and the short transmission slots. We let d denote the number of (s, 1)-packets that are scheduled in the scheduling window of D frames. Clearly

$$0 \le d \le \left\{ \left\lfloor \frac{F}{K} \right\rfloor + (D-1) \cdot \left\lfloor \frac{F-M}{K} \right\rfloor \right\} \cdot R. \quad (2)$$

We have considered the scheduling of one packet type in isolation so far. To complete our model we need to consider the scheduling of combinations of the different packet types, as well as the receiver collisions. Note that receiver collisions due to multicast packets of a give type [(l, a) or (s, a)] from a given AWG input port are accounted for in the above limits for a and b. We examine the receiver collisions due to transmissions by the other ports in the next section and return to the scheduling of combinations of different packet types and receiver collisions due to transmissions from the same port in Sections VI-C3 and VI-C4.

2) Receiver Constraint: In our analytical model of the data packet scheduling, we account for receiver collisions due to multicast packets. We allow multicast packets to be scheduled from the nodes at a given AWG input port o to the receivers at a given AWG output port d at a given time only if there is not already a multicast or unicast packet from the same input port o or another input port $o' \neq o$ scheduled to output port d at the considered time. Receiver collisions due to the packets from the

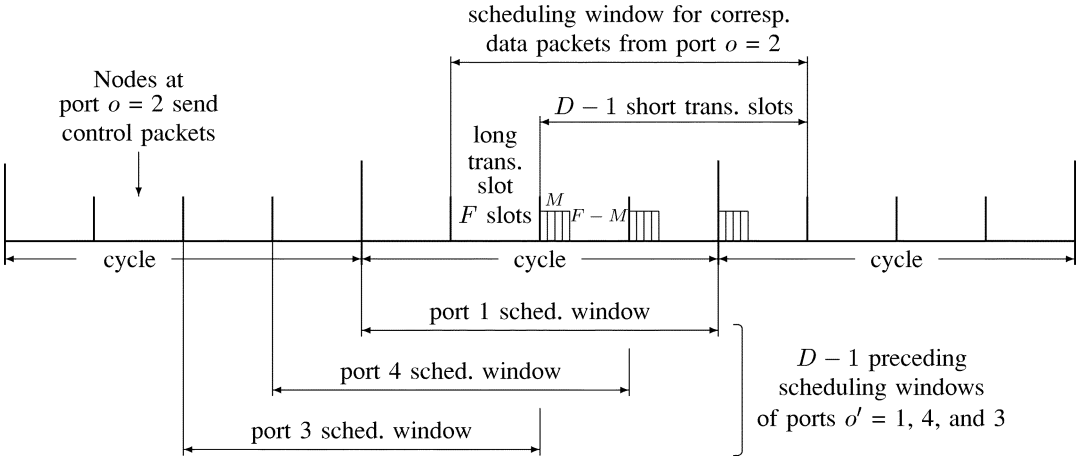


Fig. 12. Illustration of scheduling of data packets from port $o = 2$ for $D = 4$.

considered input port o are accounted for in the channel constraints discussed in the previous section and the schedulability conditions derived in Sections VI-C3 and VI-C4. In this section, the focus is on how the transmissions from the other input ports o' to the considered destination port d interfere with the transmissions from the considered input port o to port d. We note that throughout our analysis we ignore receiver collisions due to unicast packets, i.e., when scheduling a unicast packet, we do not verify whether there is already another unicast packet (from the same AWG input port or a different input port) destined to the same destination port at the same time. Our simulations in [2] and [43], as well as in Section VI-E of this paper account for receiver collisions due to unicast packets and demonstrate that this simplification gives very accurate results.

Recall from Section IV that the nodes at AWG input port o, $o = 1, \ldots, D$, send their control packets in frame o of a given cycle. Suppose that a sent control packet is successful in the control packet contention. Then, we attempt to schedule the corresponding data packet in the scheduling window that extends from frame o of the next cycle up to and including frame $(o-1)$ of the cycle thereafter, as illustrated in Fig. 12. Note that we assume here that the propagation delay is less than one cycle. Also, note that in case $o = 1$, the scheduling window coincides with the cycle boundaries. Now, consider the scheduling window from frame o to frame $(o-1)$ more closely. It consists of the long (F slot) transmission slot and $(D-1)$ short ($F-M$ slot) transmission slots. Packets of types (l, a), $(l, 1)$, and (s, a) are only scheduled in the long transmission slot. Packets of type $(s, 1)$, on the other hand, are scheduled in the long transmission slot, as well as the subsequent $(D-1)$ short transmission slots. Recall that for scalability reasons the packet scheduling is done on a first-come-first-served and first-fit basis.

Up to this point we have considered the scheduling of data packets from the nodes at a given AWG input port o, $o = 1, \ldots, D$, to the nodes at a given AWG output port d, $d = 1, \ldots, D$. Now, consider the scheduling of data packets from the nodes at the other AWG input ports o', $o' = 1, \ldots, D$, $o' \neq o$, to the nodes at AWG output port d. The scheduling windows of the other ports o' are staggered with respect to the scheduling window of port o, as illustrated in Fig. 12, for the $(D-1)$ scheduling windows that precede the considered scheduling window of port o. In each of these preceding scheduling windows, the (l, a), $(l, 1)$, and (s, a) packets are again scheduled in the first frame—the long transmission slot—and the $(s, 1)$ packets are scheduled in the long transmission slot, as well as the subsequent $D-1$ short transmission slots. As a consequence, the multicasts from the other ports o' do not interfere with the multicasts from the considered port o. However, $(s, 1)$-packets from the other ports o' may have been scheduled during the long transmission slot of the scheduling window of port o. More precisely, $(s, 1)$-packets from nodes at the other ports o' may have been scheduled for receivers at the considered destination port d during the last $(F-M)$ slots of the long transmission slot of port o. These already scheduled $(s, 1)$-packets interfere with the scheduling of multicast packets from port o.

We model this interference as follows. We divide the last $(F-M)$ slots in the long transmission slot of the scheduling window of port o into *columns* of width K slots each. Similarly, we divide the $(D-1)$ short transmission slots into columns of width K slots. Thus, there are $\xi := \lfloor (F-M)/K \rfloor$ columns in the long transmission slot and each of the short transmission slots. We refer to a column as *occupied* if in the $(D-1)$ preceding scheduling windows of the other ports o' at least one $(s, 1)$-packet has been scheduled in the column. Note that only the columns in the long transmission slot and the first $(D-2)$ short transmission slots of the scheduling window of port o can be occupied. Port o is the first to schedule data packets in the last short transmission slot of its scheduling window. Formally, we let C be a random variable denoting the number of occupied columns in a given scheduling window of port o. We let

$$\pi_l = P(C = l), \quad 0 \leq l \leq (D-1)\xi \tag{3}$$

denote the steady-state probability that l columns are occupied. We will evaluate the steady-state distribution π_l, $l = 0, \ldots, (D-1)\xi$, from a Markov chain model developed in Section VI-C5.

For the scheduling of the multicast packets from port o we need to take the number of occupied columns in the long transmission slot (i.e., the first frame) of the scheduling window of port o into consideration. Multicast packets cannot be scheduled in any occupied columns. Formally, let Γ denote the number of occupied columns in the long transmission slot of the scheduling window of port o. With the considered first-come-first-served

and first-fit scheduling policy the packets from each port are scheduled as early in the respective scheduling windows as possible. Hence

$$\Gamma = \min(C, \xi). \tag{4}$$

We now return to the analysis of the scheduling of combinations of different types of packets. We consider the cases of $\Gamma = 0$ occupied columns and $\Gamma \geq 1$ occupied columns separately.

3) Scheduling With $\Gamma = 0$ Occupied Columns: We denote the scheduling of combinations of packet types by the 5-tuple (i, a, b, c, d), which we refer to as *scheduling pattern*. The first element i in the scheduling pattern denotes the index up to which the control slots in the considered frame o have been inspected. Recall that the considered scheduling policy scans the control slots in increasing order of the index, that is, from $i = 1$ to $i = M$. If a control slot is empty or contains two (or more) collided control packets, then no data packet is scheduled. If a control slot contains exactly one control packet, that control packet is considered successful in the control packet contention and we attempt to schedule the corresponding data packet. If the data packet can be scheduled, then the corresponding counter a, b, c, or d is incremented by one. If the data packet cannot be scheduled (because there are not sufficient free channel and/or receiver resources), then the data packet fails in the scheduling and the counters a, b, c, and d remain unchanged. In summary, the scheduling pattern (i, a, b, c, d) indicates that the control slots up to index i, $i = 1, \ldots, M$, have been scanned and a packets of type (l, a), b packets of type (l, 1), c packets of type (s, a), and d packets of type (s, 1) have been successfully scheduled.

We now establish *schedulability conditions* to verify whether a given scheduling pattern is feasible. The first schedulability condition is

$$a + b + c + d \leq i. \tag{5}$$

Clearly, when we have scanned i control slots, we cannot have scheduled more than i packets.

The second schedulability condition is

$$a = 1,\ b = 0,\ c = 0,\ 0 \leq d \leq (D-1) \cdot \left\lfloor \frac{F-M}{K} \right\rfloor \cdot R. \tag{6}$$

The third schedulability condition is

$$a = 0,\ 1 \leq b \leq R,\ c = 0,\ 0 \leq d \leq (R-b) \cdot \left\lfloor \frac{F}{K} \right\rfloor + (D-1) \cdot \left\lfloor \frac{F-M}{K} \right\rfloor \cdot R. \tag{7}$$

The fourth schedulability condition is

$$a = 0,\ b = 0,\ 0 \leq c \leq \left\lfloor \frac{F}{K} \right\rfloor,\ 0 \leq d \leq \left\{ \left\lfloor \frac{F}{K} \right\rfloor - c + (D-1) \cdot \left\lfloor \frac{F-M}{K} \right\rfloor \right\} \cdot R. \tag{8}$$

We refer to a scheduling pattern (i, a, b, c, d) that satisfies the first schedulability condition (5) *and one out of* the schedulability conditions (6), (7), (8) as *feasible*. Let $P^i_{a,b,c,d}$ denote the probability that the scheduling pattern (i, a, b, c, d) arises. For all feasible scheduling patterns we calculate $P^i_{a,b,c,d}$ with the recursion

$$\begin{aligned} P^i_{a,b,c,d} = {} & P^{i-1}_{a,b,c,d} \cdot \left\{ \left(1 - \frac{\kappa}{D}\right) + \frac{\kappa}{D}(\alpha + \beta + \gamma + \delta) \right\} \\ & + P^{i-1}_{a-1,b,c,d} \cdot \frac{\kappa \hat{p}_{l,a}}{D} + P^{i-1}_{a,b-1,c,d} \cdot \frac{\kappa \hat{p}_{l,1}}{D} \\ & + P^{i-1}_{a,b,c-1,d} \cdot \frac{\kappa \hat{p}_{s,a}}{D} + P^{i-1}_{a,b,c,d-1} \cdot \frac{\kappa \hat{p}_{s,1}}{D} \end{aligned} \tag{9}$$

where

$$\alpha = \begin{cases} 0, & \text{if } (i,\ a+1,\ b,\ c,\ d) \text{ is feasible} \\ & \text{i.e., satisfies (5) and (6)} \\ \hat{p}_{l,a}, & \text{otherwise.} \end{cases} \tag{10}$$

$$\beta = \begin{cases} 0, & \text{if } (i,\ a,\ b+1,\ c,\ d) \text{ is feasible} \\ & \text{i.e., satisfies (5) and (7)} \\ \hat{p}_{l,1}, & \text{otherwise.} \end{cases} \tag{11}$$

$$\gamma = \begin{cases} 0, & \text{if } (i,\ a,\ b,\ c+1,\ d) \text{ is feasible} \\ & \text{i.e., satisfies (5) and (8)} \\ \hat{p}_{s,a}, & \text{otherwise.} \end{cases} \tag{12}$$

$$\delta = \begin{cases} 0, & \text{if } (i,\ a,\ b,\ c,\ d+1) \text{ is feasible} \\ & \text{i.e., satisfies (5) and either (6), (7), or (8)} \\ \hat{p}_{s,1}, & \text{otherwise.} \end{cases} \tag{13}$$

We initialize this recursion with $P^0_{a,b,c,d} = 1$ if $a = b = c = d = 0$, and $P^0_{a,b,c,d} = 0$ otherwise, and note that all undefined $P^i_{a,b,c,d}$ (e.g., those with negative a, b, c, or d) are set to zero.

4) Scheduling With $\Gamma \geq 1$ Occupied Columns: We assume throughout this section that

$$\left\lfloor \frac{M}{K} \right\rfloor + \left\lfloor \frac{F-M}{K} \right\rfloor = \left\lfloor \frac{F}{K} \right\rfloor. \tag{14}$$

If this condition is not satisfied, the analysis of the scheduling with occupied columns becomes more complicated since the specific order of the scheduling of the packets from the considered port o plays a role in the schedulability conditions; see the Appendix for details. If (14) is satisfied, the schedulability conditions with $\Gamma \geq 1$ occupied columns are similar to the conditions discussed in the preceding section, with the differences that (*i*) $\Gamma \geq 1$ columns in the long transmission slot are not available to (s, a)-packets and that (*ii*) (l, a) packets cannot be scheduled. Thus, the first schedulability condition is as given by (5). The second schedulability condition from the preceding section (6) is removed from consideration. The third schedulability condition is as given by (7) since we ignore the receiver collisions due to (s, 1) packets from the other ports and (l, 1) packets from port o. The fourth schedulability condition is modified to

$$a = 0,\ b = 0,\ 0 \leq c \leq \left\lfloor \frac{F}{K} \right\rfloor - \Gamma,\ 0 \leq d \leq \left\{ \left\lfloor \frac{F}{K} \right\rfloor - c + (D-1) \cdot \left\lfloor \frac{F-M}{K} \right\rfloor \right\} \cdot R. \tag{15}$$

This condition accounts for the receiver collisions due to (s, 1) packets from the other ports and (s, a) packets from port o. The receiver collisions with $(s, 1)$ packets from port o are again ignored.

We modify the definition of the scheduling pattern to the 5-tuple (Γ, i, b, c, d) which indicates that given Γ occupied columns, the control slots up to index i, $i = 1, \ldots, M$, have

been scanned and b packets of type $(l, 1)$, c packets of type (s, a), and d packets of type $(s, 1)$ have been successfully scheduled.

We let ${}^{\Gamma}Q^{i}_{b,c,d}$ denote the probability that the scheduling pattern (Γ, i, b, c, d) arises. For all feasible scheduling patterns, we calculate ${}^{\Gamma}Q^{i}_{b,c,d}$ with the recursion

$$\begin{aligned}{}^{\Gamma}Q^{i}_{b,c,d} = {}&{}^{\Gamma}Q^{i-1}_{b,c,d}\cdot\left\{\left(1-\frac{\kappa}{D}\right)+\frac{\kappa}{D}(\hat{p}_{l,a}+\beta+\gamma+\delta)\right\}\\ &+{}^{\Gamma}Q^{i-1}_{b-1,c,d}\cdot\frac{\kappa\hat{p}_{l,1}}{D}+{}^{\Gamma}Q^{i-1}_{b,c-1,d}\cdot\frac{\kappa\hat{p}_{s,a}}{D}\\ &+{}^{\Gamma}Q^{i-1}_{b,c,d-1}\cdot\frac{\kappa\hat{p}_{s,1}}{D}\end{aligned}\tag{16}$$

where

$$\beta=\begin{cases}0, & \text{if } (\Gamma,\ i,\ b+1,\ c,\ d) \text{ is feasible}\\ & \text{i.e., satisfies (5) and (7)}\\ \hat{p}_{l,1}, & \text{otherwise}\end{cases}\tag{17}$$

$$\gamma=\begin{cases}0, & \text{if } (\Gamma,\ i,\ b,\ c+1,\ d) \text{ is feasible}\\ & \text{i.e., satisfies (5) and (15)}\\ \hat{p}_{s,a}, & \text{otherwise}\end{cases}\tag{18}$$

$$\delta=\begin{cases}0, & \text{if } (\Gamma,\ i,\ b,\ c,\ d+1) \text{ is feasible}\\ & \text{i.e., satisfies (5) and either (7) or (15)}\\ \hat{p}_{s,1}, & \text{otherwise.}\end{cases}\tag{19}$$

We initialize this recursion with ${}^{\Gamma}Q^{0}_{b,c,d}=1$ if $b=c=d=0$, ${}^{\Gamma}Q^{0}_{b,c,d}=0$, otherwise, and note that all undefined ${}^{\Gamma}Q^{i}_{b,c,d}$ (e.g., those with negative b, c, or d) are set to zero.

5) Markov Chain Model for Number of Occupied Columns C: In this section, we derive the steady-state probabilities $\pi_l = P(C=l), l=0,\ldots,(D-1)\xi$, that l columns in the scheduling window of the considered port o, $o=1,\ldots,D$, are already occupied by the other ports o', $o'=1,\ldots,D$, $o'\neq o$, when port o begins its data packet scheduling. Toward this end, we construct an irreducible, positive recurrent Markov chain with the states $C=0, C=1,\ldots,C=(D-1)\xi$. The Markov chain makes state transitions in every frame. Specifically, we interpret C_n as the number of columns in the scheduling window of port o that are already occupied when the scheduling of the data packets from port o commences. After the data packets from port o have been scheduled, the Markov chain makes a state transition. We interpret C_n+1 as the number of occupied columns in the scheduling window of port $o+1$, that is, upon the state transition the considered scheduling window moves one frame into the future. (If port $o=D$ was originally considered, then C_{n+1} is the number of occupied columns in the scheduling window of port $o=1$.)

Let Z be a random variable denoting the number of columns in the short transmission slots of the scheduling window of port o that are occupied by $(s, 1)$-packets from port o when the scheduling of the data packets from port o is completed. When counting the number of columns occupied by the data packets from port o, we ignore whether these columns have already been occupied by some other port or not. The number of columns in the long transmission slot of port o that are occupied by the packets from port o are not included in Z since the scheduling window advances by one frame when port o is done with the scheduling. Thus, the first frame of port o's scheduling window is no longer included in port $(o+1)$'s scheduling window. With Z the state transition probabilities r_{ij} of the Markov chain C_n are given by

$$\begin{aligned}r_{ij}&=P(C_{n+1}=j|C_n=i)\\ &=P\left(\max\{Z,\ i-\xi\}=j|C_n=i\right)\\ &=\begin{cases}P(Z=j|C_n=i), & \text{if } j>i-\xi\\ P(Z\leq j|C_n=i), & \text{if } j=i-\xi\\ 0, & \text{if } j<i-\xi.\end{cases}\end{aligned}\tag{20}$$

To see this, note that as we make the state transition from the scheduling window of port o to the scheduling window of port $(o+1)$, the considered scheduling window advances one frame into the future and the first ξ columns of the scheduling window of port o are no longer considered. Also, note that the data packets are scheduled in a first-come-first-served and first-fit manner. Hence, the Z first columns in the advanced scheduling window are occupied by packets from port o and the $\max\{i-\xi,0\}$ first columns are occupied by packets scheduled prior to the data packet scheduling from port o. Thus, for $j>i-\xi$ columns to be occupied in the advanced scheduling window, the data packets from port o must occupy $Z=j$ columns. For $j=i-\xi$ occupied columns in the advanced scheduling window, the packets from port o may occupy $Z=0,\ldots,j$ columns.

Next, we calculate the probabilities $P(Z=j|C_n=i)$ for i, $j=0,\ldots,(D-1)\xi$. First, we consider these probabilities for $i=0$. We have for $j\geq 1$

$$\begin{aligned}&P(Z=j|C_n=0)\\ &\quad=\sum_{d=R(j-1)+1}^{Rj}P^{M}_{1,0,0,d}\\ &\qquad+\sum_{b=1}^{R}\sum_{d=(R-b)\left\lfloor\frac{F}{K}\right\rfloor+R(j-1)+1}^{(R-b)\left\lfloor\frac{F}{K}\right\rfloor+Rj}P^{M}_{0,b,0,d}\\ &\qquad+\sum_{c=0}^{\left\lfloor\frac{F}{K}\right\rfloor}\sum_{d=\left(\left\lfloor\frac{F}{K}\right\rfloor-c\right)R+R(j-1)+1}^{\left(\left\lfloor\frac{F}{K}\right\rfloor-c\right)R+Rj}P^{M}_{0,0,c,d}.\end{aligned}\tag{21}$$

For $j=0$, we have

$$P(Z=0|C_n=0)=1-\sum_{j=1}^{(D-1)\xi}P(Z=j|C_n=0).\tag{22}$$

Next, we consider the probabilities with $1\leq i\leq\xi$. We have for $j\geq 1$

$$\begin{aligned}P(Z=j|C_n=i)=&\sum_{b=1}^{R}\sum_{d=(R-b)\left\lfloor\frac{F}{K}\right\rfloor+R(j-1)+1}^{(R-b)\left\lfloor\frac{F}{K}\right\rfloor+Rj}{}^{i}Q^{M}_{b,0,d}\\ &+\sum_{c=0}^{\left\lfloor\frac{F}{K}\right\rfloor}\sum_{d=\left(\left\lfloor\frac{F}{K}\right\rfloor-c\right)R+R(j-1)+1}^{\left(\left\lfloor\frac{F}{K}\right\rfloor-c\right)R+Rj}{}^{i}Q^{M}_{0,c,d}.\end{aligned}\tag{23}$$

Furthermore, we have for $i>\xi$ and $j\geq 1$

$$P(Z=j|C_n=i)=P(Z=j|C_n=\xi)\tag{24}$$

and finally for $1 \le i \le (D-1)\xi$ and $j = 0$

$$P(Z = 0|C_n = i) = 1 - \sum_{j=1}^{(D-1)\xi} P(Z = j|C_n = i). \quad (25)$$

With the calculated state transition probabilities r_{ij}, $i, j = 0, \ldots, (D-1)\xi$, we find the steady-state probabilities π_l, $l = 0, \ldots, (D-1)\xi$, as the solution to

$$\pi_j = \sum_i \pi_i \cdot r_{ij} \quad (26)$$

$$\sum_j \pi_j = 1. \quad (27)$$

6) Expected Numbers of Scheduled Packets: We obtain the expected number of scheduled packets as

$$E[X_{l,a}] = \pi_0 \cdot \sum_d P^M_{1,0,0,d} \quad (28)$$

$$E[X_{l,1}] = \pi_0 \cdot \sum_{b,d} b \cdot P^M_{0,b,0,d} + \sum_{l=1}^{\xi-1} \pi_l \cdot \sum_{b,d} b \cdot {}^l Q^M_{b,0,d} + \left(\sum_{l \ge \xi} \pi_l \right) \cdot \sum_{b,d} b \cdot {}^\xi Q^M_{b,0,d} \quad (29)$$

$$E[X_{s,a}] = \pi_0 \cdot \sum_{c,d} c \cdot P^M_{0,0,c,d} + \sum_{l=1}^{\xi-1} \pi_l \cdot \sum_{c,d} c \cdot {}^l Q^M_{0,c,d} + \left(\sum_{l \ge \xi} \pi_l \right) \cdot \sum_{c,d} c \cdot {}^\xi Q^M_{0,c,d} \quad (30)$$

$$E[X_{s,1}] = \pi_0 \cdot \sum_{a,b,c,d} d \cdot P^M_{a,b,c,d} + \sum_{l=1}^{\xi-1} \pi_l \cdot \sum_{b,c,d} d \cdot {}^l Q^M_{b,c,d} + \left(\sum_{l \ge \xi} \pi_l \right) \cdot \sum_{b,c,d} d \cdot {}^\xi Q^M_{b,c,d} \quad (31)$$

where the summations are over all feasible scheduling patterns as given by the respective schedulability conditions (5)–(8) and (15).

D. Network/System Analysis

In this section, we establish equilibrium conditions for the network. These equilibrium conditions result in a system of equations that can be solved by straightforward numerical techniques for the unknowns η, $\hat{p}_{l,a}$, $\hat{p}_{l,1}$, $\hat{p}_{s,a}$, and $\hat{p}_{s,1}$, which in turn give the expected numbers of successfully scheduled packets through the recursive technique given in the preceding section.

For ease of notation let

$$E[X] = E[X_{l,a}] + E[X_{l,1}] + E[X_{s,a}] + E[X_{s,1}]. \quad (32)$$

Also, note that for large S we may reasonably approximate the expected value $E[\eta]$ by η. With this approximation the first equilibrium condition is

$$\frac{\sigma}{D} \cdot \eta = E[X]. \quad (33)$$

This is because $\sigma \cdot \eta$ new packets are generated in each cycle by the nodes attached to a given AWG input port. With probability $1/D$ each of the generated packets is destined to a given (fixed) AWG output port (splitter). On the other hand, $E[X]$ packets are scheduled (and transmitted) on average from a given AWG input port to a given AWG output port in one cycle; in equilibrium as many new packets must be generated.

The other four equilibrium conditions are

$$E[X_{l,a}] = p_{l,a} \cdot E[X] \quad (34)$$

$$E[X_{l,1}] = p_{l,1} \cdot E[X] \quad (35)$$

$$E[X_{s,a}] = p_{s,a} \cdot E[X] \quad (36)$$

$$E[X_{s,1}] = p_{s,1} \cdot E[X]. \quad (37)$$

These hold because in equilibrium the mean number of scheduled packets of a given type from a given AWG input port to a given AWG output port in one cycle (LHS in the equations) is equal to the number of newly generated packets of this type in one cycle (RHS in the equations).

The first equilibrium condition (33) and any three of the four conditions (34)–(37), along with $\hat{p}_{l,a} + \hat{p}_{l,1} + \hat{p}_{s,a} + \hat{p}_{s,1} = 1$ give a system of five linear independent equations, which can be solved by standard numerical techniques for the five unknowns η, $\hat{p}_{l,a}$, $\hat{p}_{l,1}$, $\hat{p}_{s,a}$, and $\hat{p}_{s,1}$. These are then used to calculate the expected numbers of scheduled packets from a given AWG input port to a given AWG output port per cycle $E[X_{l,a}]$, $E[X_{l,1}]$, $E[X_{s,a}]$ and $E[X_{s,1}]$ using the recursive approach given in the preceding section.

Based on the expected numbers of scheduled packets, we evaluate the network performance metrics as follows. The mean aggregate transmitter throughput TH_T is defined as the mean number of transmitting nodes in the network in steady state and is given by

$$TH_T = D^2 \cdot \frac{F \cdot (E[X_{l,a}] + E[X_{l,1}]) + K \cdot (E[X_{s,a}] + E[X_{s,1}])}{F \cdot D}. \quad (38)$$

Note that in the considered scenario where all receivers of a multicast are located at one random splitter the mean aggregate multicast throughput (multicast completions per frame) is equal to the mean aggregate transmitter throughput.

The mean receiver throughput TH_R is defined as the average number of receiving nodes in the network in steady state and is given by

$$TH_R = D^2 \cdot \frac{F \cdot S \cdot E[X_{l,a}] + F \cdot E[X_{l,1}] + K \cdot S \cdot E[X_{s,a}] + K \cdot E[X_{s,1}]}{F \cdot D}. \quad (39)$$

The mean delay in the analytical network model is defined as the average time in cycles from the generation of the control packet corresponding to a data packet until the successful scheduling of the data packet. Following the arguments in [2] and [43], we obtain

$$\text{Delay} = \frac{S}{D \cdot E[X]} - \frac{1-\sigma}{\sigma}. \quad (40)$$

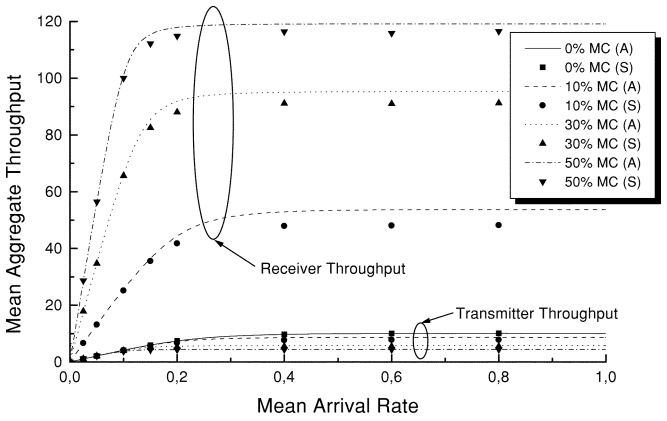


Fig. 13. Mean aggregate transmitter and receiver throughput versus mean arrival rate σ (packet/cycle) for different fractions $p_{l,a} + p_{s,a} = \{0\%, 10\%, 30\%, 50\%\}$ of multicast packets (fraction of short packets $p_{s,a} + p_{s,1} = 0.75$ fixed).

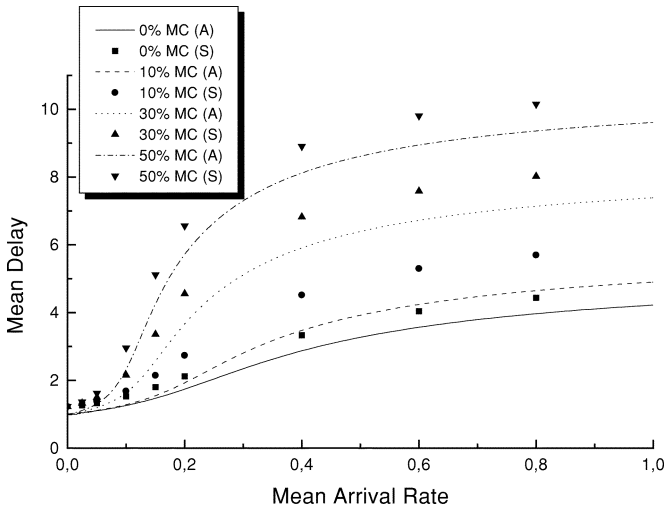


Fig. 14. Mean delay (cycles) versus mean arrival rate σ (packet/cycle) for different fractions $p_{l,a} + p_{s,a} = \{0\%, 10\%, 30\%, 50\%\}$ of multicast packets (fraction of short packets $p_{s,a} + p_{s,1} = 0.75$ fixed).

E. Numerical Results

In this section, we conduct numerical investigations of the interaction between unicast and multicast traffic. This investigation quantifies the benefits of multicasting concurrently with reservation control traffic in conjunction with unicast with spatial wavelength reuse. The default network parameters are set as follows: Number of nodes $N = 200$, number of available wavelengths at each AWG port $D \cdot R = 8$, cycle length $D \cdot F = 800$ slots, number of reservation slots per frame $M = 40$, retransmission probability $p = 0.8$. We have also conducted extensive simulations of a more realistic network in order to verify the accuracy of the analytical model. As opposed to the analysis, in the simulation a given node cannot transmit unicast packets to itself. Furthermore, in the simulation not only the packet type (length, unicast or multicast) but also the destination of a given unicast or multicast packet are not renewed, i.e., are persistent, when retransmitting the corresponding control packet (recall that the analysis assumes that the type of the packet is persistent while the destination is nonpersistent). In addition, the simulation takes all receiver conflicts into account, i.e., a given unicast or multicast packet is not scheduled if the receiver(s) of the intended destination(s) is (are) busy. Each simulation was run for 10^7 slots including a warm-up phase of 10^6 slots. Using the method of batch means, we calculated the 98% confidence intervals for the performance metrics, which were always smaller than 4% of the corresponding sample means.

Fig. 13 depicts the mean aggregate transmitter throughput (mean number of transmitting nodes) and receiver throughput (mean number of receiving nodes) in steady state for different fractions $p_{l,a} + p_{s,a} = \{0\%, 10\%, 30\%, 50\%\}$ of multicast packets. In all cases, the fraction of short data packets is $p_{s,a} + p_{s,1} = 0.75$. Accordingly, the fraction of long data packets is $p_{l,a} + p_{l,1} = 0.25$. The AWG degree is set to $D = 4$. Hence, the number of used FSRs is $R = 2$, the frame size equals $F = 200$ slots, and short packets are $K = 160$ slots long. If the fraction of multicast packets is equal to 0% all packets are unicast and transmitter throughput is identical to receiver throughput. As shown in Fig. 13, increasing the fraction of multicast packets from 0% up to 50% results in a dramatically larger receiver throughput and a slightly smaller transmitter throughput. This is due to the fact that with an increasing fraction of multicast packets more receivers are used, resulting in a larger receiver throughput. On the other hand, the transmitter throughput is slightly decreased since nodes are less likely to find free receivers, leading to a smaller number of transmissions and thereby smaller transmitter throughput. Note that analysis and simulation results match very well at low traffic loads. At medium to high loads, on the other hand, the analysis provides a slightly larger receiver throughput than the simulation. This is due to the assumed nonpersistency of destination in the analysis. As opposed to the simulation, in the analysis unsuccessful control packets renew the destination of the corresponding multicast packets. Consequently, in the analysis previously conflicting multicast packets are less likely to collide again and can be successfully scheduled resulting in an increased receiver throughput. Overall the results clearly illustrate that scheduling multicast packets concurrently with reservation control in each frame significantly improves the receiver utilization.

Fig. 14 depicts the mean delay (in cycles) for different fractions $p_{l,a} + p_{s,a} = \{0\%, 10\%, 30\%, 50\%\}$ of multicast packets. As expected, with increasing arrival rate the mean delay grows due to more channel and receiver collisions. Moreover, with an increasing fraction of multicast traffic the mean delay becomes larger. Again, this is because with increasing multicast traffic the receiver utilization is higher, resulting in more unsuccessful reservation requests and retransmissions. Note that the analysis yields smaller delay values than the simulation. This is because of two reasons. First, due to the destination nonpersistency in the analysis, control packets are more likely to be successful and have to be retransmitted fewer times resulting in a smaller delay. Second, the definitions of packet delay are slightly different for simulation and analysis. In the simulation, the packet delay is defined as the time interval between packet generation and end of packet transmission. In the analysis, the packet delay is defined as the time interval between packet generation and the time when the packet is successfully scheduled but not yet transmitted.

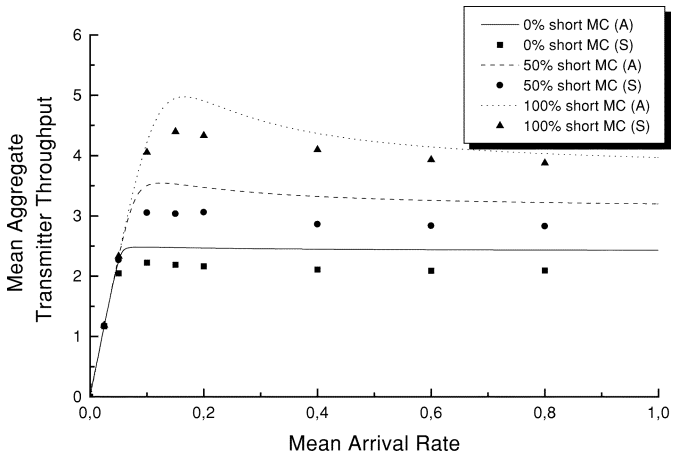


Fig. 15. Mean aggregate transmitter throughput versus mean arrival rate σ (packet/cycle) for different ratios {0%,50%, 100%} of short multicast packets (20% multicast traffic fixed).

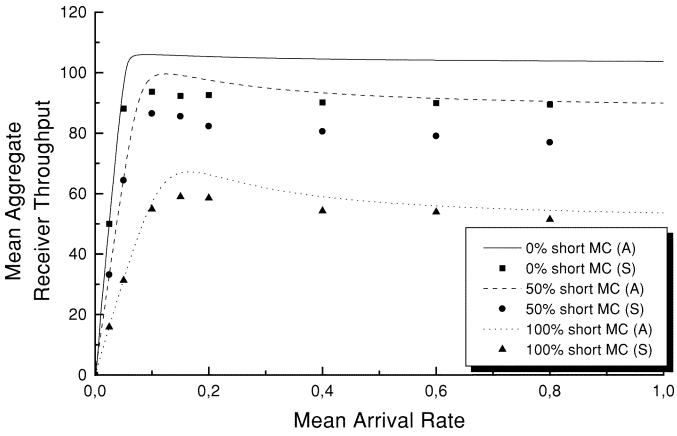


Fig. 16. Mean aggregate receiver throughput versus mean arrival rate σ (packet/cycle) for different ratios {0%, 50%, 100%} of short multicast packets (20% multicast traffic fixed).

In Figs. 15–17, we set the AWG degree to $D = 2$, the fraction of long data packets to $p_{l,a} + p_{l,1} = 0.25$. Long packets are $L = F = 400$ slots and short packets are $K = 120$ slots long. The number of reservation slots per frame is $M = 40$ and $R = 4$ FSRs of the underlying AWG are used. 80% of the data packets are unicast, i.e., $p_{l,1} + p_{s,1} = 0.8$. Accordingly, 20% of the data packets are multicast, i.e., $p_{l,a} + p_{s,a} = 0.2$. The multicast packets can be either only short, both short and long, or only long. Specifically, we consider different ratios $p_{s,a}/(p_{s,a} + p_{l,a}) = 0\%$, 50%, and 100% of short multicast packets.

Fig. 15 shows how the mean aggregate transmitter throughput is increased by varying the ratio of short and long multicast packets. If 0% of the multicast packets are short, i.e., all multicast packets are long, the transmitter throughput is rather small. By increasing the number of short multicast packets from 0% up to 100% the transmitter throughput is significantly increased; for 100% of short multicast packets the mean aggregate throughput is roughly doubled. This is due to the fact that without long multicast packets nodes ready to send short multicast packets are more likely to find free receivers, which translates into an increased transmitter throughput. However,

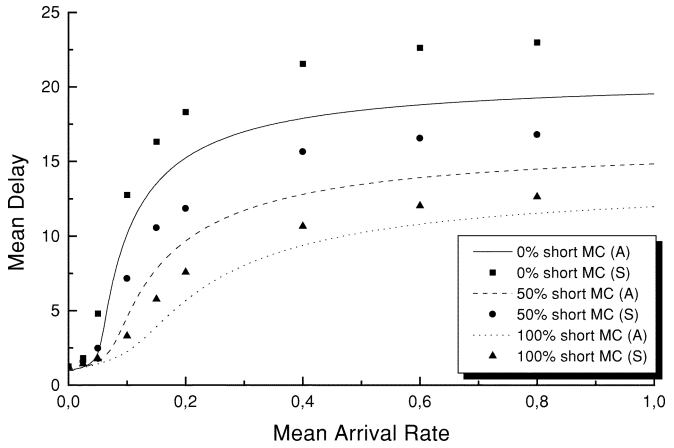


Fig. 17. Mean delay (cycles) versus mean arrival rate σ (packet/cycle) for different ratios {0%, 50%, 100%} of short multicast packets (20% multicast traffic fixed).

increasing the number of short multicast packets leads to a decreased receiver throughput, as depicted in Fig. 16. Thus, there is a tradeoff between channel and receiver utilization. Again, analysis and simulation results match very well at low traffic loads. However, at medium to high loads the analysis and simulation results exhibit some discrepancy. While we observe that the discrepancy is not that large for the case of 100% short multicast packets, the mismatch is more pronounced if the amount of long multicast packets is increased. This is again due to the destination nonpersistency assumption made in the analysis which resolves the destination conflicts as opposed to the simulation resulting in a larger mean aggregate receiver throughput.

Fig. 17 depicts the mean delay versus mean arrival rate σ for different ratios of short and long multicast packets {0%, 50%, 100%}. We observe that with an increasing number of short multicast packets the mean delay is decreased. This is because in the presence of fewer long multicast packets, receivers are more likely to be free. As a consequence, more data packets are scheduled resulting in fewer retransmissions of control packets and decreased delay.

VII. Conclusion

We have investigated multicasting in an AWG-based single-hop WDM network. In the considered network, wavelength-insensitive splitters are attached to each AWG output port, allowing for efficient optical multicasting. For multicast traffic, we have reconfirmed that the partitioning of multicast groups alleviates the receiver conflicts but creates a channel bottleneck. We have demonstrated that the spatial wavelength reuse in the AWG-based network effectively mitigates the channel bottleneck. For typical multicast traffic the AWG-based network achieves more than twice the transmitter throughput and roughly 30% larger receiver and multicast throughput compared with the widely studied single-hop networks based on the PSC, which is a broadcast-and-select device that does not allow for spatial wavelength reuse.

For a typical mix of unicast and multicast traffic we have examined the interplay between multicast transmissions

concurrently with spread control traffic and unicast transmissions with spatial wavelength reuse. A reservation MAC protocol with a periodic reservation phase is employed in the AWG-based network to dynamically allocate wavelengths and receivers, thereby completely avoiding collisions of data packets. During the reservation phase all receivers are tuned to spectrum slices carrying the spread control traffic. We found that multicast transmissions concurrently with the spread control traffic effectively exploit the tuning of the receivers to the control slices, resulting in significantly increased receiver utilization. In addition, exploiting the spatial wavelength reuse for unicast traffic, which typically faces a severe channel bottleneck, but no receiver conflicts, results in an overall improved throughput-delay performance.

There are many interesting avenues for future work on multicasting over the AWG. One such avenue is to investigate logical partitioning in conjunction with (fixed) physical partitioning. Note that in the AWG-based network each splitter locally broadcasts packets to the S attached receivers. The physical partitioning among the different splitters could, thus, be combined with (local) logical partitioning at the individual splitters to further improve the multicasting performance. Another interesting direction for future work is to consider multicasting over a hybrid network formed by operating an AWG in parallel with a PSC. (A preliminary study of such an AWG||PSC for unicast traffic is provided in [42].) The AWG||PSC network offers interesting tradeoffs for multicasting in that a multicast destined to receivers at one AWG output port could be conducted over the AWG, while a multicast destined to receivers at several AWG output ports may be more efficiently conducted over the PSC.

APPENDIX

In this Appendix, we analyze the packet scheduling with $\Gamma \geq 1$ occupied columns if

$$\left\lfloor \frac{M}{K} \right\rfloor + \left\lfloor \frac{F-M}{K} \right\rfloor + 1 = \left\lfloor \frac{F}{K} \right\rfloor. \tag{41}$$

If (41) holds, as opposed to (14), then there are situations where the feasibility of a scheduling pattern depends not only on the number of scheduled packets of the different types, but also the order in which these packets are scheduled. Specifically, if (41) holds and $\Gamma \geq 1$, then the number of (s,a) and $(s,1)$ packets that can be scheduled depends on the specific order in which these packets appear in the control slots. Consider an example with $F = 200$, $M = 30$, $K = 20$, $R = 5$, and $\Gamma = 1$. In this example, $\lfloor M/K \rfloor + \lfloor (F-M)/K \rfloor = 9$ and $\lfloor F/K \rfloor = 10$. Now, suppose that for a given scheduling window of a given port we first have 15 $(s,1)$-packets to schedule and then 6 (s,a)-packets. Since we can reasonably ignore potential receiver collisions due to $(s,1)$-packets from other ports and $(s,1)$-packets from the considered port in our analytical model (see Section VI-E), the 15 $(s,1)$-packets are scheduled in the first 60 slots of the considered scheduling window. The six (s,a)-packets are scheduled in the 120 subsequent slots. Thus, there is room for one more (s,a)-packet in this scenario. Next, consider a scenario where we first have the 6 (s,a)-packets to schedule and then the 15 $(s,1)$-packets. To avoid receiver collisions of the (s,a)-packets with the $(s,1)$-packets from the other ports (which occupy the first column, i.e., slots 31 through 50, in the considered scheduling window), one (s,a)-packet is scheduled in slots 1 through 20 and the other five (s,a)-packets are scheduled in slots 51 through 150. Then, following the adopted first-come-first-served and first-fit scheduling policy, five $(s,1)$-packets are scheduled in slots 21 through 40 and the remaining ten $(s,1)$-packets are scheduled in slots 151 through 190. Thus, there is no room for any additional packet. Note that if we had $F = 199$ and all other parameters as given in the example, then (14) would hold and we would not have the situation where the feasibility of a scheduling pattern depends on the order of the packets.

More generally, whenever (41) holds, $\Gamma \geq 1$, and an (s,a)-packet is scheduled in slot $M + K \cdot \Gamma + 1$ and onwards, then there are slots "wasted" due to the packet ordering, as illustrated in the second scenario in the above example. Formally, we add an indicator e to the scheduling pattern to capture this effect. The indicator e is set to one if an (s,a)-packet is scheduled in slot $M + K \cdot \Gamma + 1$ and onwards; e is set to zero otherwise. The scheduling pattern (Γ, i, b, c, d, e) indicates that given Γ occupied columns, the control slots up to index i, $i = 1, \ldots, M$, have been scanned and b $(l,1)$-packets, c (s,a)-packets, and d $(s,1)$-packets have been scheduled. Also, if $e = 1$ then an (s,a)-packet is scheduled starting in slot $M + K \cdot \Gamma + 1$, and if $e = 0$ this is not the case. For the case considered in this Appendix, the first schedulability condition is given by (5) and the second original schedulability condition (6) is not considered. The third schedulability is as given by (7). The fourth schedulability condition is replaced by

$$a = 0,\ b = 0,\ 0 \leq c \leq \left\lfloor \frac{F}{K} \right\rfloor - \Gamma,\ 0 \leq d \leq \left(\left\lfloor \frac{F}{K} \right\rfloor - c \right) \cdot R + (D-1) \cdot \left\lfloor \frac{F-M}{K} \right\rfloor \cdot R,\ e = 0 \tag{42}$$

and

$$a = 0,\ b = 0,\ 1 \leq c \leq \left\lfloor \frac{M}{K} \right\rfloor + \left\lfloor \frac{F-M}{K} \right\rfloor - \Gamma,\ \max\left\{0, \left(\left\lfloor \frac{M}{K} \right\rfloor - 1 \right) \cdot R + 1 \right\} \leq d \leq \left(\left\lfloor \frac{M}{K} \right\rfloor + \left\lfloor \frac{F-M}{K} \right\rfloor - c \right) \cdot R + (D-1) \cdot \left\lfloor \frac{F-M}{K} \right\rfloor \cdot R,\ e = 1. \tag{43}$$

We let ${}^{\Gamma}Q^{i}_{b,c,d,e}$ denote the probability that the scheduling pattern (Γ, i, b, c, d, e) arises. For all feasible scheduling patterns, we calculate ${}^{\Gamma}Q^{i}_{b,c,d,e}$ with the recursion

$$\begin{aligned} {}^{\Gamma}Q^{i}_{b,c,d,e} = {}& {}^{\Gamma}Q^{i-1}_{b,c,d,e} \cdot \left\{ \left(1 - \frac{\kappa}{D}\right) + \frac{\kappa}{D}(\hat{p}_{l,a} + \beta + \gamma + \delta) \right\} \\ & + {}^{\Gamma}Q^{i-1}_{b-1,c,d,e} \cdot \frac{\kappa \hat{p}_{l,1}}{D} + {}^{\Gamma}Q^{i-1}_{b,c-1,d,e} \cdot \frac{\kappa \hat{p}_{s,a}}{D} \\ & + {}^{\Gamma}Q^{i-1}_{b,c,d-1,e} \cdot \frac{\kappa \hat{p}_{s,1}}{D} + {}^{\Gamma}Q^{i-1}_{b,c-1,d,e-1} \cdot \frac{\kappa \hat{p}_{s,a}}{D} \end{aligned} \tag{44}$$

where

$$\beta = \begin{cases} 0, & \text{if } (\Gamma,\ i,\ b+1,\ c,\ d,\ e) \text{ is feasible} \\ \hat{p}_{l,1}, & \text{otherwise} \end{cases} \tag{45}$$

$$\gamma = \begin{cases} 0, & \text{if } (\Gamma,\ i,\ b,\ c+1,\ d,\ e) \text{ or} \\ & (\Gamma,\ i,\ b,\ c+1,\ d,\ e+1) \text{ is feasible} \\ \hat{p}_{s,a}, & \text{otherwise} \end{cases} \tag{46}$$

$$\delta = \begin{cases} 0, & \text{if } (\Gamma,\ i,\ b,\ c,\ d+1,\ e) \text{ is feasible} \\ \hat{p}_{s,1}, & \text{otherwise.} \end{cases} \tag{47}$$

We initialize this recursion with ${}^{\Gamma}Q^{0}_{b,c,d,e} = 1$ if $b = c = d = e = 0$, and ${}^{\Gamma}Q^{0}_{b,c,d,e} = 0$, otherwise and note that all undefined ${}^{\Gamma}Q^{i}_{b,c,d,e}$ (e.g., those with negative b, c, d, or e) are set to zero.

ACKNOWLEDGMENT

The authors are indebted to Prof. H. Mittelmann of Arizona State University for providing us with valuable background on numerical analysis. They are grateful to H.-S. Yang of Arizona State University for assisting in running some of the simulations.

REFERENCES

[1] B. Mukherjee, "WDM optical communication networks: Progress and challenges," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1810–1824, Oct. 2000.

[2] M. Scheutzow, M. Maier, M. Reisslein, and A. Wolisz, "Wavelength reuse for efficient packet-switched transport in an AWG-based metro WDM network," *J. Lightwave Technol.*, vol. 21, pp. 1435–1455, June 2003.

[3] A. Okada, T. Sakamoto, Y. Sakai, and K. N. *et al.*, "All-optical packet routing by an out-of-band optical label and wavelength conversion in a full-mesh network based on a cyclic-frequency AWG," in *Proc. OFC 2001 Tech. Dig.*, Anaheim, CA, Mar. 2001, paper ThG5.

[4] D. Stoll, P. Leisching, H. Bock, and A. Richter, "Metropolitan DWDM: A dynamically configurable ring for the KomNet field trial in Berlin," *IEEE Commun. Mag.*, vol. 39, pp. 106–113, Feb. 2001.

[5] K. V. Shrikhande, I. M. White, D. Wonglumsom, and S. M. Gemelos *et al.*, "HORNET: a packet-over-WDM multiple access metropolitan area ring network," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 2004–2016, Oct. 2000.

[6] I. M. White, K. Shrikhande, M. S. Rogge, and M. Gemelos *et al.*, "Architecture and protocols for HORNET: A novel packet-over-WDM multiple-access MAN," in *Proc. IEEE GLOBECOM*, San Francisco, CA, Nov./Dec. 2000.

[7] S. Yao, S. J. B. Yoo, and B. Mukherjee, "All-optical packet switching for metropolitan area networks: opportunities and challenges," *IEEE Commun. Mag.*, vol. 39, pp. 142–148, Mar. 2001.

[8] H. Woesner, M. Maier, and A. Wolisz, "Comparison of single-hop and multihop AWG-based WDM networks," in *Proc. Optical Network Design and Modeling (ONDM)*, Torino, Italy, Feb. 2002.

[9] M. Bandai, S. Shiokawa, and I. Sasase, "Performance analysis of multicasting protocol in WDM-based single-hop lightwave networks," in *Proc. IEEE GLOBECOM*, Phoenix, AZ, Nov. 1997.

[10] M. S. Borella and B. Mukherjee, "Limits of multicasting in a packet-switched WDM single-hop local lightwave network," *J. High Speed Networks*, vol. 4, pp. 155–167, 1995.

[11] ——, "A reservation-based multicasting protocol for WDM local lightwave networks," in *Proc. IEEE ICC '95*, Seattle, WA, June 1995, pp. 1277–1281.

[12] J. P. Jue and B. Mukherjee, "The advantages of partitioning multicast transmissions in a single-hop optical WDM network," in *Proc. ICC '97*, Montreal, Canada, June 1997.

[13] H.-C. Lin, P.-S. Liu, and H. Chu, "A reservation-based multicast scheduling algorithm with reservation window for single-hop WDM network," in *Proc. IEEE Int. Conf. Networks*, Singapore, Sept. 2000.

[14] H.-C. Lin and C.-H. Wang, "Minimizing the number of multicast transmissions in single-hop WDM networks," in *Proc. IEEE Int. Conf. Communications (ICC)*, New Orleans, LA, June 2000, pp. 1645–1649.

[15] E. Modiano, "Unscheduled multicasts in WDM broadcast-and-select networks," in *Proc. IEEE INFOCOM'98*, San Francisco, CA, Mar. 1998, pp. 86–93.

[16] ——, "Random algorithms for scheduling multicast traffic in WDM broadcast-and-select networks," *IEEE/ACM Trans. Networking*, vol. 7, pp. 425–434, June 1999.

[17] Z. Ortiz, G. N. Rouskas, and H. G. Perros, "Maximizing multicast throughput in WDM networks with tuning latencies using the virtual receiver concept," *Eur. Trans. Telecommun.*, vol. 11, no. 1, pp. 63–72, Jan./Feb. 2000.

[18] G. N. Rouskas and M. H. Ammar, "Multi-destination communication over single-hop lightwave WDM networks," in *Proc. IEEE INFOCOM'94*, Toronto, Canada, June 1994, pp. 1520–1527.

[19] ——, "Multidestination communication over tunable-receiver single-hop WDM networks," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 501–511, Apr. 1997.

[20] L. Sahasrabuddhe and B. Mukherjee, "Probability distribution of the receiver busy time in a multicasting local lightwave network," in *Proc. ICC'97*, Montreal, Canada, June 1997, pp. 116–120.

[21] S.-T. Sheu and C.-P. Huang, "An efficient multicasting protocol for WDM star-coupler networks," in *Proc. IEEE Int. Symp. Computers and Communications*, Alexandria, Egypt, July 1997, pp. 579–583.

[22] W.-Y. Tseng and S.-Y. Kuo, "A combinational media access protocol for multicast traffic in single-hop WDM LANs," in *Proc. IEEE GLOBECOM*, Sydney, Australia, Nov. 1998, pp. 294–299.

[23] W.-Y. Tseng, C.-C. Sue, and S.-Y. Kuo, "Performance analysis for unicast and multicast traffic in broadcast-and-select WDM networks," in *Proc. IEEE Int. Symp. Computers and Communications*, Red Sea, Egypt, July 1999, pp. 72–78.

[24] A. Bianco, G. Galante, E. Leonardi, F. Neri, and A. Nucci, "Scheduling algorithms for multicast traffic in TDM/WDM networks with arbitrary tuning latencies," in *Proc. IEEE GLOBECOM'2001*, San Antonio, TX, Nov. 2001, pp. 1551–1556.

[25] A. Ding and G.-S. Poo, "A survey of optical multicast over WDM networks," *Comput. Commun.*, vol. 26, no. 2, pp. 193–200, Feb. 2003.

[26] A. M. Hamad and A. E. Kamal, "A survey of multicasting protocols for broadcast-and-select single-hop networks," *IEEE Network*, vol. 16, no. 4, pp. 36–48, July/Aug. 2002.

[27] J. He, S.-H. G. Chan, and D. H. Tsang, "Multicasting in WDM networks," *IEEE Commun. Surveys and Tutorials*, Dec. 2002.

[28] T. Kitamura, M. Iizuka, and M. Sakuta, "A new partition scheduling algorithm by prioritizing the transmission of multicast packets with less destination address overlap in WDM single-hop networks," in *Proc. IEEE GLOBECOM*, San Antonio, TX, Nov. 2001, pp. 1469–1473.

[29] H.-C. Lin and C.-H. Wang, "A hybrid multicast scheduling algorithm for single-hop WDM networks," in *Proc. IEEE INFOCOM'01*, Anchorage, AK, Apr. 2001, pp. 169–178.

[30] ——, "A hybrid multicast scheduling algorithm for single-hop WDM networks," *J. Lightwave Technol.*, vol. 19, pp. 1654–1664, Nov. 2001.

[31] T.-L. Liu, C.-F. Hsu, and N.-F. Huang, "Multicast QoS traffic scheduling with arbitrary tuning latencies in single-hop WDM networks," in *Proc. ICC 2002*, New York, Apr. 2002, pp. 2886–2890.

[32] M. Maier, M. Reisslein, and A. Wolisz, "A hybrid MAC protocol for a metro WDM network using multiple free spectral ranges of an arrayed-waveguide grating," *Comput. Networks*, vol. 41, no. 4, pp. 407–433, Mar. 2003.

[33] L. Giehmann, A. Gladisch, and N. Hanik *et al.*, "The application of code division multiple access for transport overhead information in transparent optical networks," in *Proc. OFC 1998 Tech. Dig., paper WM42*, San Jose, CA, Feb. 1998, pp. 228–229.

[34] B. Mukherjee, *Optical Communication Networks*. New York: McGraw-Hill, 1997.

[35] A. Similjanic, "An efficient channel access protocol for an optical star network," in *Proc. IEEE ICC '98*, 1998, pp. 514–519.

[36] R. Ramaswami and K. N. Sivarajan, *Optical Networks—A Practical Perspective*. San Mateo, CA: Morgan Kaufmann, 1998.

[37] C. Fujihashi and T. Hirasawa, "All error detecting and 1 bit error correcting coding for zero-error high speed optical transmission," in *Proc. Annu. Meeting Lasers and Electro-Optics Society*, vol. 2, Dec. 1998, pp. 301–302.

[38] T. Duman, Personal Communication, Communication Systems Lab, Dept. Elec. Eng., Arizona State Univ., May 2003.

[39] V. Sivaraman and G. N. Rouskas, "HiPeR-l: a high performance reservation protocol with look-ahead for broadcast WDM networks," in *Proc. IEEE INFOCOM '97*, Kobe, Japan, Apr. 1997, pp. 1270–1277.

[40] M. Maier, *Metropolitan Area WDM Networks–An AWG Based Approach*. Norwell, MA: Kluwer, 2003.

[41] J. Lu and L. Kleinrock, "A wavelength division multiple access protocol for high-speed local area networks with a passive star topology," *Perform. Evaluat.*, vol. 16, no. 1–3, pp. 223–239, Nov. 1992.

[42] C. Fan, M. Maier, and M. Reisslein, "The AWG‖PSC network: a performance enhanced single-hop WDM network with heterogeneous protection," in *Proc. IEEE INFOCOM'03*, San Francisco, CA, Apr. 2003, pp. 2279–2289.

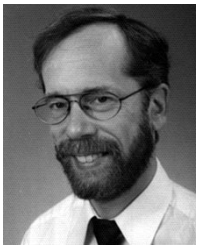
[43] M. Maier, M. Scheutzow, M. Reisslein, and A. Wolisz, "Wavelength reuse for efficient transport of variable-size packets in a metro WDM network," in *Proc. IEEE INFOCOM'02*, New York, July 2002, pp. 1432–1441.

[44] M. Maier, M. Reisslein, and A. Wolisz, "High performance switchless WDM network using multiple free spectral ranges of an arrayed-waveguide grating," in *Proc. SPIE Terabit Optical Networking: Architecture, Control, and Management Issues*, Boston, MA, Nov. 2000, (Paper received the Best Paper Award of the Conference), pp. 101–112.

Martin Maier received the Dipl.-Ing. and Dr.-Ing. degrees (both with distrinction) in electrical engineering from the Technical University Berlin, Berlin, Germany, in 1998 and 2003, respectively.

He was a Visiting Researcher at University of Southern California (USC), Los Angeles, Arizona State University (ASU), Tempe, and Massachusetts Institute of Technology (MIT), Cambridge. He also participates in the national research project TransiNet. His research interests include network and node architectures, routing and switching paradigms, protection, multicasting, and the design, performance evaluation, and optimization of MAC protocols for optical WDM networks, with particular focus on metro networks.

Dr. Maier was a recipient of the two-year Deutsche Telekom Doctoral Scholarship from June 1999 through May 2001. He is also a corecipient of the Best Paper Award presented at the SPIE Photonics East 2000–Terabit Optical Networking Conference. He is the author of Metropolitan Area WDM Networks-*An AWG Based Approach* (Norwell, MA: Kluwer, 2003).



Michael Scheutzow received the Diploma in mathematics from the Johann-Wolfgang-Goethe University, Frankfurt/Main, Germany, in 1979, the Ph.D. degree in mathematics (*magna cum laude*) and the Habilitation in mathematics from the University of Kaiserslautern, Kaiserslautern, Germany, in 1983 and 1988, respectively.

He is a Professor in the Department of Mathematics, Technical University Berlin, Berlin, Germany. From 1988 to 1990, he was Project Leader with Tecmath GmbH, Kaiserslautern, Germany. Since 1990, he has been a Professor of stochastics in the Department of Mathematics at the Technical University Berlin. From 1997 to 1999, he was Associate Chair of the department. He has visited the University of Carbondale, Carbondale, IL, Rutgers University, Piscataway, NJ, University of Rochester, Rochester, NY, Warwick University, Coventry, U.K., and The Mathematical Sciences Research Institute (MSRI), University of California, Berkeley.

Martin Reisslein (A'96–S'97–M'98) received the Dipl.-Ing. (FH) degree from the Fachhochschule Dieburg, Germany, in 1994, the M.S.E. degree from the University of Pennsylvania, Philadelphia, in 1996, both in electrical engineering, and the Ph.D. degree in systems engineering from the University of Pennsylvania in 1998.

He is an Assistant Professor in the Department of Electrical Engineering, Arizona State University, Tempe. During the academic year 1994–1995, he visited the University of Pennsylvania as a Fulbright Scholar. From July 1998 to October 2000, he was a Scientist with the German National Research Center for Information Technology (GMD FOKUS), Berlin, Germany. While in Berlin, he was teaching courses on performance evaluation and computer networking at the Technical University Berlin, Berlin, Germany. He maintains an extensive library of video traces for network performance evaluation, including frame size traces of MPEG-4 and H.263 encoded video, at http://trace.eas.asu.edu. His research interests are in the areas of Internet quality-of-service, video traffic characterization, wireless networking, and optical networking.

Dr. Reisslein is the Editor-in-Chief of the *IEEE Communications Surveys and Tutorials* and has served on the Technical Program Committees of *IEEE INFOCOM*, *IEEE GLOBECOM*, and the *IEEE International Symposium on Computer and Communications*. He has organized sessions at the *IEEE Computer Communications Workshop (CCW)*. He is corecipient of the Best Paper Award of the *SPIE Photonics East 2000—Terabit Optical Networking* Conference.