Polymorphic Control for Cost-Effective Design of Optical Networks[*]

CHUNMING QIAO†‡, YOUSONG MEI‡, MYUNGSIK YOO‡ AND XIJUN ZHANG‡
Lab for Advanced Network Design, Evaluation and Research (LANDER)
Department of Computer Science and Engineering†
Department of Electrical Engineering‡
University at Buffalo (SUNY)
Buffalo, New York 14260
qiao@computer.org†, {*ymei,myoo,xz2*}*@cse.buffalo.edu*‡

Abstract.

A WDM optical layer can provide differentiated services to various upper layer protocols by forming many virtual optical networks (VONs). Polymorphic control of an optical layer in the form of On-demand-reconfiguration in some VONs and Self-reconfiguration in other VONs is proposed. The former is suitable for bursty traffic and short-lived connections, and the latter is for steady traffic and long-lived connections.

In respect to On-demand-reconfiguration, efficient distributed wavelength reservation protocols, and in particular optical burst switching (OBS) protocols supporting bursty traffic (e.g. Internet traffic) are described. In respect to Self-reconfiguration, optimal algorithms to schedule all-to-all personalized communications (AAPC) in WDM rings and its extensions are described.

1 INTRODUCTION

Optical networks, especially WDM networks, have received and will continue to receive an enormous amount of attention. However, despite the great deal of efforts and progress made at the device, component, and point-to-point transmission subsystem level, all-optical networking research is still in its infancy. To deal with network control issues in all-optical networks, a top-down approach that emphasizes architectural solutions which circumvent current and/or fundamental limits imposed by the devices/components needs to be taken.

In this paper, we will describe a framework under which an optical layer can be used to support different classes of services (CoS) having different traffic characteristics (e.g. either steady or bursty) and performance requirements (e.g. throughput-critical or delay-sensitive). Specifically, an optical network can be sliced into several *virtual optical networks* (VONs) by allocating a subset of limited resources (e.g. fibers, wavelengths, transceivers and wavelength converters) to each VON, and deploying appropriate switches (e.g. slow/fast or small/large) in each VON. Higher level user applications can then be run either *directly* over each VON or *indirectly* through IP, ATM or SONET/SDH. Since each VON will be controlled (i.e. configured and reconfigured) differently in order to support each class of services in a cost-effective way, *polymorphic control* of the optical network (i.e. multiple VONs) is needed.

The paper is organized as follows. Section 2 gives an overview of the proposed framework for polymorphic control. Section 3 describes *On-demand* reconfiguration including distributed wavelength reservation protocols and optical burst switching. Section 4 describes *Self* reconfiguration including optimal scheduling and permutation embedding. Finally, Section 5 concludes the paper.

2 OVERVIEW

In this section, we first describe the principle of cost-effective design, and then apply the principle in developing polymorphic control schemes with a focus on the concepts of network *reconfiguration* in point-to-point WDM networks.

[*] An earlier version of this paper was presented at the Workshop on Multichannel Optical Networks, NJ, U.S.A., March 1998. This research is sponsored in part by grants from U.S. National Science Foundation under contracts MIP-9409864 and ANIR-9801778.

2.1 THE PRINCIPLE OF COST-EFFECTIVE DESIGN

In WDM optical networks, performance (or effectiveness) refers broadly to metrics such as *delay*, *throughput* and *blocking probability* for dynamic traffic, *wavelength requirement* and *schedule length* for static traffic, as well as more abstract ones such as *reliability* (or resilience) and *scalability*. The types of the resources include *network bandwidth* (e.g., in terms of the number of wavelengths), *I/O capacity* (e.g., in terms of the number of simultaneous WDM transmitters/receivers at each node), *control complexity* (e.g., in terms of the amount of processing needed for network operations and management), and *interconnectivity* (e.g., in terms of the ability of the switches to convert or interchange wavelengths), among others. Since implementation cost and/or complexity is always a factor affecting performance, the availability of certain resources, and even the fate of a networking technology, cost-effective design is of prominent interest.

One unique aspect of the cost-effective design in WDM optical networks is to achieve resource-balance. For example, in a WDM ring, one needs to determine the number of wavelengths needed for a given number of transceiver pairs at each node (or vice versa) to support a traffic pattern. Another unique aspect has to do with the relations and trade-offs between bandwidth utilization and control complexity. Specifically, in an electronic network, each link is managed as a unit and its status (e.g. up or down) information is either maintained by a central controller or sent to other nodes under distributed control. However, in an optical network, each wavelength needs to be managed (and allocated/deallocated) as a unit. Accordingly, neither sending the complete usage information of a link (which can have multiple fibers, each carrying multiple wavelengths) to all other nodes, nor maintaining such information at a central controller may be feasible. Instead, distributed control based on local knowledge (e.g. the wavelength usage information on the outgoing links of a node) may be needed [1, 2].

2.2 POLYMORPHIC RECONFIGURATION SCHEMES

Sometimes, new connections need to be established and then released dynamically in order to handle traffic flows injected into the network, or to re-route existing connections in the presence of network faults or congestion. This is an example of *On-demand reconfiguration*, also known as dynamic reconfiguration, which trades increased control complexity for improved bandwidth utilization, and is suitable for *short-lived* connections required by certain applications.

Contrary to On-demand reconfiguration, *Self reconfiguration* is suitable for *long-lived* connections carrying steady traffic. It allows a large number of these connections to time-share the limited bandwidth of the network while maintaining the transparency of all-optical connections without using expensive electronics (e.g. SONET Add-Drop Multiplexers), or requiring complex control as in On-demand reconfiguration. More specifically, the set of desired connections is partitioned into several conflict-free subsets which are then *scheduled* such that each subset of connections is established for a period of time (e.g. a super time-slot) in a round-robin, time-shared (or coarse-grained TDM) fashion. Note that, with sufficient bandwidth, all the required connections (e.g. all-to-all personalized connections or AAPC) can be established simultaneously, at which time scheduling becomes the same as *embedding*.

While On-demand reconfiguration and Self reconfiguration may appear to be two opposing strategies for achieving balance between bandwidth and control, a network must integrate both in a complementary way in order to meet the various requirements of the applications. In the rest of the paper, we will describe *distributed wavelength reservation* and optical burst switching (OBS) protocols for On-demand reconfiguration, as well as connection scheduling and embedding algorithms for Self-reconfiguration.

3 ON-DEMAND RECONFIGURATION

In a large network, On-demand reconfiguration can be accomplished by using distributed control. An optical network with distributed control may be considered as having a *data network* consisting of the optical switches interconnected by several data wavelengths, and a *control network* consisting of the control units (CUs) interconnected by one or more control wavelengths. Each optical switch is controlled by a CU, and each CU exchanges the control information with other CUs by sending and receiving control packets.

We assume that each node (which refers to the combination of a CU and a switch) maintains the *local* usage information of the data wavelengths (or channels) accessible to the switch only [1, 2]. When a node receives a control packet requesting for a connection, it processes the control packet, reserves a channel on the outgoing link based on its local usage information, and then forwards the control packet to the next node on a hop-by-hop basis. This eliminates the potential performance bottleneck caused by a central controller, and also increases the *reliability* of the system when compared to centralized control. In what follows, we will first describe two-way distributed wavelength reservation protocols and then describe optical burst switching protocols based on one-way reservation.

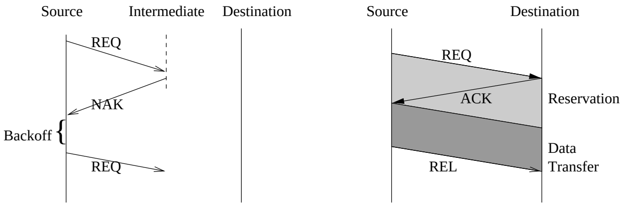
3.1 TWO-WAY WAVELENGTH RESERVATION SCHEMES

Connection establishment based on two-way reservation under distributed control has been studied in multicomputer, telephony and high-speed (non-optical) networks. Similar approaches may be used in optical networks as well. For example, one may let each node broadcast the local wavelength usage information to every other node, so that every node has the global information.

However, as mentioned earlier, since a WDM link may carry many wavelengths, whose usage information changes more often than the up/down status of an electronic link, the above method may not be efficient in terms of the amount of control (or signaling) bandwidth consumed. This argues for a distributed wavelength reservation protocol based on the local usage information only. Several other unique features of the WDM networks also argue for new distributed control protocols. For example, when there is no wavelength conversion, a WDM network is different from a multi-channel electronic network in that a connection has to use the same wavelength (instead of any available channel) on different links along a path. Such a way of establishing a connection was called *path multiplexing* (PM) in [2, 3, 4]. Of course, with all-optical wavelength converters (the technology for which is quite immature), a connection can be established by using different wavelengths (or channels) on different links using the so-called *link multiplexing* (LM) approach [2, 3, 4]. In addition, since a wavelength may support a high bandwidth of several Gigabit/s, the issues of *how to minimize the set-up delay* and *how to minimize the bandwidth wasted* during the set-up period deserve more attention than before.

Note that most of the studies of the *wavelength assignment* methods in WDM networks (see for example, [5, 6, 7, 8, 9]) have assumed *centralized control*. The cost-effectiveness of wavelength converters (or the performance advantage of LM over PM) has been studied for either static communications (e.g. scheduling permutations) [3, 10, 11, 12], or dynamic communications under centralized control [4, 13, 14, 15, 16, 17]. These results show that the performance advantage of LM over PM is limited. In particular, in a TDM network (where a channel corresponds to a time slot instead of a wavelength), the *overall* communication latency can be higher in LM than in PM since interchanging time slots introduces delays [4]. However, these results do not apply to distributed control environments since control overhead such as processing delay has been ignored in all these studies.

We now describe a basic two-way distributed reservation protocol for PM based on the ideas of *distributed time-slot reservation schemes* proposed in [4, 18]. The basic protocol is called *source-initiated-reservation* (SIR) and adopts *parallel* (P) reservation with the *dropping* (D) policy. Specifically, let the nodes along a path (of L-hop long) from a source to its destination be numbered 0 through L. Each node i, where $0 \leq i \leq L$, maintains a list (or set), A_i, of all the available *outgoing* channels (i.e. from node i to node $i+1$). To request for a connection, the source reserves, *in parallel*, all the channels in A_0 (assuming that A_0 is not empty), and sends a request packet (REQ) to the destination on a hop-by-hop basis. REQ includes a field $REQ.cws$ (candidate wavelength set) that is initialized to A_0. When node i, where $1 \leq i \leq L$, receives REQ, it calculates $REQ.cws \cap A_i$, which results in a common subset of channels that are available on all links up to node $i+1$. If the result of set-joint operation is not empty, node i updates $REQ.cws$, reserves all the channels in $REQ.cws$ and forwards REQ to node $i+1$. If the result is empty, REQ is *dropped* and an negative acknowledgement packet (NAK) is sent to the source following the reverse of the partial path taken by REQ, also on a hop-by-hop basis (see Figure 1(a)). The NAK releases all the channels reserved by the corresponding REQ and informs the source of the failure. The source will send another REQ after a random back-off period in an attempt to establish the connection later.



(a). unsuccessful

(b). successful

Figure 1: A basic SIR protocol.

If REQ does arrive at the destination with a non-empty $REQ.cws$, the destination will select one channel, say λ, from $REQ.cws$ for use by the connection and send a positive acknowledgement (ACK) back to the source node identifying the channel. The ACK follows the reverse of the path taken by the corresponding REQ and releases all the channels reserved by the REQ except λ. Once the source gets the ACK, it can start transferring data over λ, and after transferring all the data, the source sends a release packet (REL) to the destination to tear down the connection. This is illustrated in Figure 1(b).

The following variations of the basic protocol may also be considered.

Holding (H): Each REQ, when generated, is assigned a maximum lifetime. If at node i, where $1 \leq i \leq L$, $REQ.cws \cap A_i$ is empty, REQ waits at node i, hoping that at least one channel in $REQ.cws$ will be released by another connection for inclusion by A_i, and thereby it may continue its journey. REQ is dropped only when its lifetime expires (and afterwards, the same process as described earlier takes place). Note that this policy may reduce the set-up delay, but at the same time, waste the bandwidth on the partially reserved path during the holding (or waiting) period.

Sequential (S) Reservation: Parallel reservation of all the channels in A_0 initially and in $REQ.cws$ subsequently seems to increase the chances of success for a specific connection. However, it wastes a lot of bandwidth and may decrease the chances of success for other connections. An alternative is to reserve only one channel in A_0 initially (i.e. set $REQ.cws$ to include that channel only). If $REQ.cws$ ever becomes empty, the source can try a different channel in A_0 *immediately* (or after a random back-off period as in [2]).

Destination-Initiated-Reservation (DIR): Each node simply forwards REQ without reserving any channels (implying that REQ will not wait nor be dropped). $REQ.cws$ is set and updated as in SIR with parallel reservation just to collect usage information for the destination. After the destination receives REQ, it sends ACK and starts the reservation process. This process is similar to those described earlier in that either parallel or sequential reservation (with either dropping or holding) may be used. Specifically, ACK also carries a field $ACK.cws$, which is initially set to be equal to $REQ.cws$ when using parallel reservation, or one of the channels in $REQ.cws$ when using sequential reservation. In addition, a node i, where $1 \leq i \leq L$, determines $ACK.cws \cap A_{i-1}$ upon receiving ACK (note that this implies that node i needs to maintain the set of available *incoming* channels, A_{i-1}, as well). Finally, if node i decides to drop ACK either immediately after $ACK.cws$ becomes empty (when using the dropping policy), or only after ACK's lifetime expires (when using the holding policy), it sends a NAK to the destination to release the partially established path, but may also send a NAK to inform the source of the failure so that the source may try again later, as illustrated in Figure 2 (a).

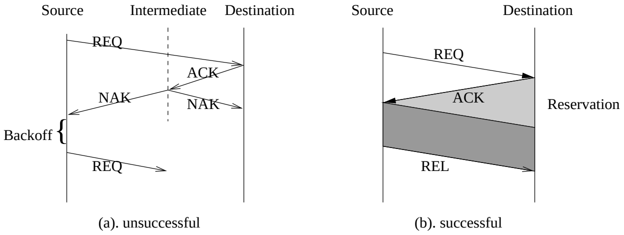


Figure 2: A basic DIR protocol.

Note that such a DIR protocol has two useful features when compared to a SIR protocol. One is that it can reduce the amount of bandwidth wasted during the connection set-up (or reservation) period by half, as can be seen by comparing Figure 2 (b) with Figure 1 (b). The other is that the destination can make a more informed decision based on $REQ.cws$ than can a SIR protocol based on A_0 as an initial set of candidate wavelengths. In fact, even if the destination receives an empty $REQ.cws$, it may still take advantage of this knowledge (e.g. by trying to establish the connection along an alternate path). Note that, an alternative to the above DIR protocol is to let an intermediate node to drop REQ as soon as $REQ.cws$ becomes empty [19].

We also note that although parallel reservation does not apply to LM since any available channel on a link can be used, both SIR and DIR using sequential reservation under either the dropping or holding policy do. Performance studies [1] show that in PM, parallel reservation results in a higher throughput than sequential reservation if the average end-to-end propagation delay is small relative to the average connection duration (but performs worse otherwise). In addition, DIR outperforms SIR, especially for LM and when the propagation delay is large. Finally, LM performs much better than PM under all cases and in some specific situations where the propagation delay is large and a DIR protocol is used, LM can achieve twice the throughput of PM. This suggests that the cost of extra hardware for wavelength conversion required by LM may be justified for dynamic traffic under distributed control (at least more so than for static traffic and/or under centralized control).

So far, only a few papers have addressed the basic issues related to distributed control in WDM networks. In [20], a protocol in which each node maintains a globle information on wavelength usage as well as topology, and sends a request for establishing a connection to all intermediate nodes simultaneously was proposed but performance issues related to the connection set-up delays and bandwidth utilization were not discussed. Hop-by-hop based distributed reservation protocols for both PM and LM based on local wavelength usage information were first proposed and evaluated in [1, 2]. Similar protocols were discussed and evaluated in [19, 21] for PM only.

3.2 Optical Burst Switching (OBS) and Just Enough Time (JET)

Both SIR and DIR protocols are based on two-way reservation, and thus have a set-up latency which is equal to the sum of the round-trip propagation and the total processing delay of REQ and ACK. In this subsection, we describe a novel paradigm called optical burst switching (or OBS) which is based on one-way reservation. Note that at 2.5 Gb/s, a burst of 500 Kbytes can be transmitted in about 1.6ms. However, it would take ACK 2.5ms just to propagate over a distance of merely 500km. This explains why one-way reservation protocols are generally better than their two-way counterparts for bursty traffic over a relatively long distance.

OBS is suitable for supporting portion of the Internet traffic (especially WWW traffic) which is self-similar, or in other words, bursty at all time scales [22, 23, 24, 25]. It can be used to streamline both software (e.g. ATM signaling) and hardware (e.g. SONET equipment) in the next generation Optical Internet. More specifically, OBS can support IP over WDM by running IP software, along with other control software as a part of the interface between the network layer and the WDM layer, on top of every optical (WDM) switch. In the WDM layer, a dedicated control wavelength is used to provide the “static/physical” links between these IP entities, which maintain topology and routing tables.

Figure 3 illustrates the basic concept of an OBS protocol called *Just-Enough-Time* (or JET) [26, 27]. To send a data burst (of many IP packets), a control packet, which is treated as an ordinary IP packet, is routed from a source to its destination (on a hop-by-hop basis) to set up an all-optical connection. More specifically, each node chooses an appropriate wavelength on the outgoing link, reserves it for the duration of the following burst (starting at the expected arrival time of the data burst), and sets up the optical switch (for simplicity, we have assumed that the total processing time is δ at each node). Meanwhile, the burst waits at the source in the *electronic* domain. After an *offset time*, T_o, (but without having to wait for an acknowledgement from the destination), the burst is sent in optical signals. Let L be the number of hops along the path (e.g., in Figure 3, $L = 3$), then T_o is chosen to be at least $\delta \cdot L$ in order to ensure that there is an enough time for each node to complete the processing of the control packet before the burst arrives. As a result, once a burst is sent, it *passes through* the intermediate nodes without going through any buffer, O/E/O conversions, or intermediate IP entities.

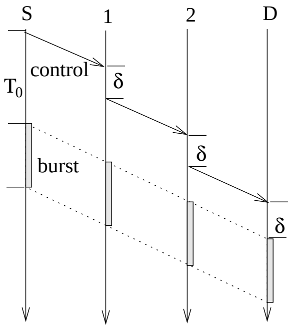


Figure 3: OBS using Just-Enough-Time (JET) protocol.

OBS can leverage the attractive properties of optical communications, and at the same time, take into account its limitations, in order to provide flexible and efficient high-bandwidth physical transport services. To certain extent, switching optical bursts achieves a balance between switching coarse-grained optical circuits and switching fine-grained optical packets/cells, and thus combines the best of both paradigms. Specifically, since in OBS, the wavelength on a link used by the burst will be released as soon as the burst passes through the link, bursts from different sources to different destinations can effectively utilize the bandwidth of the same wavelength on a link in a time-shared, statistical multiplexed fashion. This results in much more efficient bandwidth utilizations than wavelength-routing, which is suitable for long-lived connections. It also overcomes the problems of limited connectivity in wavelength-routed networks where the number of “lightpaths” that can be established is limited by the number of wavelengths available.

In addition, due to the limited “opaqueness” of the control packet, OBS can achieve a high degree of adaptivity to congestions or faults (e.g. by using deflection-routing), and support priority-based routing as in optical packet/cell switching. However, since OBS switches bursts, whose size can be much larger than that of IP packets (or ATM cells), OBS results in a much lower overhead. By using out-of-band control and especially the offset time as in JET, the coupling between the control packet and the burst is not as tight as it has to be in optical packet/cell switching. Finally, optical packet/cell switching requires the use of fiber-delay lines (FDLs) to delay the payload while the header is being processed, while in JET-based OBS, FDLs are not required. Note that, as long as the minimum value of T_o is used, a burst would encounter the same end-to-end latency even if it is sent along with the control packet as in optical packet switching.

A critical issue in any one-way reservation protocol is the data loss rate. Specifically, in case a control packet fails to reserve the bandwidth at an intermediate node, the corresponding burst may have to be dropped, and a negative acknowledgement may be sent back to the source so that it may retransmit the control packet and the burst later. This wastes the bandwidth on the partially established path. However, such bandwidth has been reserved exclusively for the burst, it would be wasted even if one does not send out the burst (as in two-way reservation). In order to completely eliminate the possibility of such bandwidth waste, a burst (or an optical packet) would have to be stored in an electronic buffer (after going through O/E conversions) and later relayed (after going through E/O conversions) to its destination. FDLs providing limited delays at intermediate nodes, which are not mandatory in JET-based OBS, can be used to reduce the bandwidth waste. Note that, in JET-based OBS, 100% of the buffering capacity of the FDLs is available for resolving conflicts while in optical packet/cell switching, less than 100% is (some FDLs are only used to delay data burst while the corresponding control packets are being processed). In addition, JET can not only improve the bandwidth utilization, but also facilitate intelligent buffer management, resulting in a significantly reduced burst dropping probability [26, 27].

Note that, the dropping probability of some bursts can also be reduced without using any FDLs. Specifically, some burst can be assigned a higher priority (and thus guaranteed a lower dropping probability) by simply using an additional offset time. The corresponding control packet can then, in effect, reserve the bandwidth much in advance than others, thus resulting in a higher success probability. The additional offset time needed by higher priority bursts to achieve several orders of magnitude of reduction in the dropping probability is comparable to the average length of the lower priority bursts, and could be small (e.g. less than a few milliseconds) relative to the end-to-end propagation delay in a wide-area network [28].

Although JET uses one-way reservation, its idea may also be applied to protocols using two-way reservation, e.g. the DIR protocols mentioned earlier. Specifically, the bandwidth wasted during the set-up period in DIR protocols can be further reduced by reserving the bandwidth for the period beginning at the time the first bit of the data is expected to arrive, instead of at the time an ACK arrives (see Figure 2(a)).

The idea of JET may also be applied to SIR protocols, making it similar to ERVC (*Efficient Reservation Virtual Circuit*), which was proposed for high-speed TDM networks (e.g. the Thunder and Lightning) [29]. In other words, we can have JET-DIR or JET-SIR. Since an important issue is how accurately one can predict the time at which the first and last bits of the data will arrive and leave, respectively, on each hop, JET-DIR is better since the only uncertain timing factor is the delay encounted by ACK, not that encounted by both REQ and ACK as in JET-SIR. Furthermore, while the source may not know the exact number of hops to be taken (or the propagation delay per hop), such information can be easily collected by REQ, and made available to the destination when using JET-DIR. Note that, if a burst is long relative to the round-trip propagation delay, JET-DIR (or other two-way reservation protocols) may be better than JET (or other one-way reservation protocols). Similarly, if a burst is short, then it can be sent as a part of a control packet. Finally, when JET is used to send out a burst of unknown length, an estimated length may be assumed. If it is an over-estimation, another control (release) packet may be sent to release the extra bandwidth reserved. If it is an under-estimation, then the remaining data will be sent as one or more additional bursts.

Note that, as discussed in [26, 30, 27], in addition to JET, OBS may use other one-way reservation protocols based on the idea of tell-n-go (TAG), which is also known as Fast-reservation protocol (FRP), or ATM block transfer with immediate transmission (ABT-IT) [31, 32, 33, 34, 35]. As in optical packet/cell switching, using TAG-based OBS protocols, FDLs will be needed at each intermediate node to delay the burst, while its corresponding control packet is being processed. Accordingly, less than 100% of the FDLs can be used to help resolve conflicts and improve performance (as in optical packet/cell switching).

4 Self Reconfiguration

As mentioned earlier, for permanent and semi-permanent connections, Self-reconfiguration is a way to avoid complex control and its associated overhead involved in On-demand-reconfiguration. A special instance of Self reconfiguration is to let the network maintain the set of all-to-all personalized connections (AAPC) (that is, one connection from every node to every other node).

Because of the limited resources (e.g., wavelengths and transceivers at each node) in a WDM network, not all the connections in AAPC may be established at the same time. However, if the traffic over each connection requires less bandwidth than that of one wavelength, and the aggregated traffic required does not exceed the total bandwidth available in the network, all these connections may still be supported by letting them share the limited number of wavelengths available (in the time domain). This can be accomplished by multiplexing the traffic through electronics using SONET Add-Drop Multiplexers (ADMs) [36, 37, 38, 39, 40], for example. Or, if a high degree of transparency, in other words, an all-optical path is desired for each connection, we can partition AAPC into several conflict-free subsets, and then *schedule* (or establish) the connections in each subset for a period of a fixed duration, which we refer to as a *round*. The number of rounds needed to schedule AAPC will be called the *schedule length*. With AAPC scheduling, message routing is simple since each node needs only to *wait* for an appropriate round to transmit/receive at a predetermined wavelength. In addition, the maximum connection latency between any pair of nodes, which is proportional to the schedule length, can also be guaranteed along with the throughput (or traffic) of each connection.

4.1 Scheduling in WDM Rings

We first consider the problem of optimal scheduling of all-to-all personalized connections in bidirectional WDM rings [41, 42]. Assume that, in a ring of N nodes, each connection will be established either clockwise or counter-clockwise under shortest-path routing, such that the stride (i.e. the number of hops) of any connection is no greater than $\frac{N}{2}$ hops. The problem of optimal scheduling is complicated under the assumption that each node can use *multiple* (but a limited number of) transmitters and receivers to simultaneously communicate to several other nodes at different wavelengths. Having multiple transceivers per node is necessary to fully utilize the bandwidth provided by multiple wavelengths as discussed below.

4.1.1 Balance I/O Capacity and Network Bandwidth

Our study has shown that the schedule length has a lower bound determined by not only the number of nodes (denoted by N) and wavelengths (denoted by K) in the ring, but also by the number of simultaneous transmitters/receivers at each node (denoted by T). More specifically, based on analysis, the theoretical lower bound (LB) on the schedule length with a unlimited K is $LB(T,-) = \lceil \frac{N-1}{T} \rceil$ (this is because in AAPC, each node needs to originate $N-1$ connections), and with a unlimited T is $LB(-,K) = \lceil \frac{N^2}{8K} \rceil$ (this is because the total number of channels available in the bidirectional ring is $2NK$ and that required by AAPC is $2N\sum_{s=1}^{N/2-1} s + N\cdot N/2$ or $N^3/4$). Hence, for a given T and K, the schedule length has a lower bound of $LB(T,K) = max\{LB(T,-), LB(-,K)\}$.

This LB is useful not only in guiding the search for optimal scheduling algorithms, but also in achieving cost-effective design with resource-balance. Specifically, a system with K wavelengths and T transceivers per node is considered cost-effective if using one fewer wavelength or one fewer transceiver per node will increase the schedule length. In other words, the following conditions need to be satisfied in a cost-effective system: $LB(T,K) < LB(T,K-1)$ and $LB(T,K) < LB(T-1,K)$.

For a given K (or T), there may be 0, 1 or more values of T (or K) that satisfy these cost-effective conditions. The points in Figure 4 (a) show the appropriate value(s) of T for a given K, as well as the appropriate value(s) of K for a given T in a 16-node ring. In addition, the LB at all possible points (i.e. K and T values) is drawn in Figure 4 (b) for reference. Note that a point, e.g. $K = 3, T = 1$ in Figure 4 (a), (or the performance at that point), is said to be *I/O-limited* if $LB(T,-) > LB(-,K)$ since in such a case, it is possible to shorten the schedule length *only* when T increases (see Figure 4 (b)). Similarly, a point, e.g. $K = 6, T = 3$, is said to be *bandwidth-limited* if $LB(T,-) < LB(-,K)$ since in such a case, it is possible to shorten the schedule length *only* when K increases. Other points, e.g. $K = 8$, $T = 4$, at which $LB(T,-) = LB(-,K)$ are said to be *jointly-limited*. As can be seen, a system is cost-effective at only a few points which are either I/O-limited, bandwidth-limited or jointly-limited, but is not cost-effective at most of the points. For instance, from Figure 4 (b), one should not use anywhere between 17 to 31 wavelengths since the schedule length remains the same as using 16 wavelengths (regardless of T). In addition, T should not exceed 8 unless $K = 32$ (in fact, this is true as long as $K \leq N$ regardless of N).

Given that the LB described above is based on theoretical analysis, and as such, may not be achievable for an arbitrary T and K even with the best possible (often the most complicated) scheduling algorithm. This is especially true when we assume PM. As a result, the cost-effective points shown in Figure 4 (a) may not always be accurate in practice. However, as to be shown later, our heuristic scheduling algorithm will be able to achieve the LB in most cases. Even in the cases where the LB is not achieved because we have assumed PM and/or used the hueristics, a schedule length very close to the LB can be achieved, and the cost-effective points can still be determined in a similar way based on the schedule length achieved by such an algorithm.

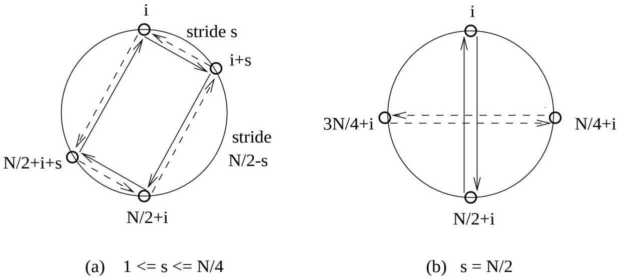
We also note that based on the analysis of the lower bounds on the schedule length, to establish all the connections in AAPC simultaneously, i.e., to make $LB(T,K) = 1$, the minimum resources required will be

$$T^* = N - 1 \quad \text{and} \quad K^* = \lceil \tfrac{N^2-1}{8} \rceil \qquad (1)$$

Although these requirements are derived from theoretical analysis[1], scheduling algorithms that can schedule AAPC in only one round given these minimum resources exist, as to be described next. In the following discussions, we will assume $T \leq T^*$ and $K \leq K^*$.

4.1.2 *Optimal Scheduling Algorithms*

A basic strategy called *Complementary Assembly with Dual Strides* (CADS) has been proposed for an even N [42], which groups up to four connections to form a *circle* of N links. Two of these connections have a stride of s and the other two have the complementary stride, that is, $\frac{N}{2} - s$ as shown in Figure 5(a). A special case is $s = \frac{N}{2}$, in which only two connections with the same stride are combined in a circle as shown in Figure 5(b).



(a) 1 <= s <= N/4

(b) s = N/2

Figure 5: The CADS method to form circles.

A similar strategy, called Complementary Assembly with Triadic Stride (CATS), can be used when N is odd. Using CATS, a circle consists of up to four (sometimes only three) connections with strides s, $\frac{N-1}{2} - s$, $s + 1$, and $\frac{N-1}{2} - s$, where $0 \leq s < \frac{N-1}{2}$. It is shown that using CADS or CATS, the connections in AAPC are grouped into a minimal number of circles (which is equal to $\lceil \frac{N^2-1}{8} \rceil$).

If $K = 1$ and $T = 1$, one circle is simply scheduled in a round, and this can achieve the LB. In a multiplexed ring where $K > 1$, multiple circles can be scheduled in a round. Heuristic scheduling algorithms that generate scheduling length approaching the LB have been proposed [41, 42]. Note that, since each circle of connections can be established on a single wavelength using PM, such results imply that LM, which requires wavelength converters, may not be cost-effective for this application.

4.1.3 *Other Extensions*

The methods to partition AAPC into a minimal number of circles and the optimal scheduling algorithms described above can be extended in several ways. For example, to effectively tolerate the propagation delay and thereby reducing the scheduling latency, pipelined transmission may be used [45]. In addition, to accommodate non-uniform traffic, one may apply CADS or CATS to form "full" circles

[1] The value of K^* was first reported in [43] (for odd N only), and then in [41, 44].

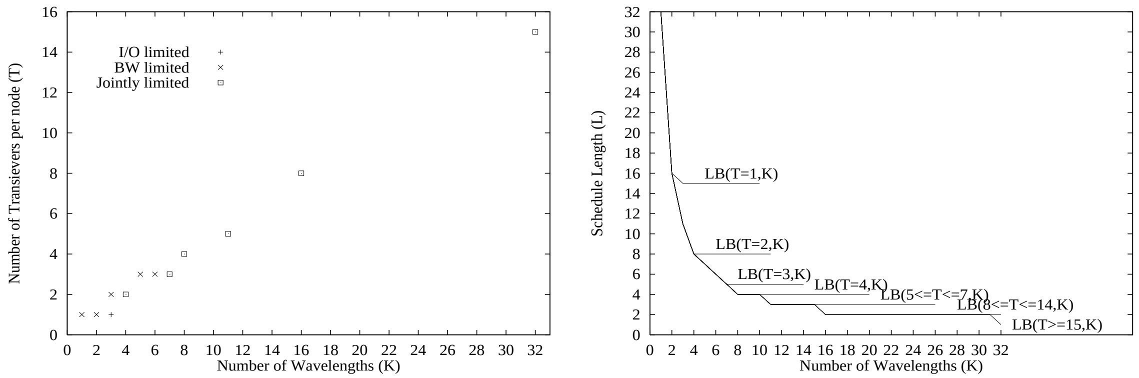


Figure 4: (a) Appropriate values of T and K in a 16-node ring; (b) the LB for all possible K and T.

first, and then use heuristics to form "partial" circles. Finally, to groom traffic in a SONET/WDM ring, one may first construct circles using connections of basic rate (e.g. OC-3), and then groom multiple (e.g. 16) circles onto the same wavelength (e.g. operating at OC-48). To reduce the number of SONET ADMs, a heuristic algorithm whereby a circle is groomed with other circles if it involves the least number of additional end-nodes (SONET ADMs) has been proposed [36].

4.2 Permutation Embedding

We now examine a special case of Self-reconfiguration called *embedding*. In particular, we study the effect of the PM and LM approaches on permutation embedding (and scheduling) by determining the minimum number of wavelengths (or time slots) in PM *as well as* LM, denoted by K_{LM} and K_{PM} respectively, for an N-node network to be rearrangeably nonblocking and wide-sense nonblocking.

Table 1 below summarizes the bounds on (or sometimes the values of) K_{LM} and K_{PM} in a class of networks with regular topologies. Note that although regular topologies are more likely to be used in metropolitan and local area networks than in wide area networks, the results obtained are also indicative of those for irregular topologies, for which precise analytic results are difficult, if not impossible, to obtain.

Our results show that LM and PM are equally effective in linear arrays, and LM is slightly more effective than PM in rings, meshes, tori and hypercubes, especially for wide-sense nonblocking ones. These results suggest that *PM may be more cost-effective than LM* for this type of applications, given that LM requires sophisticated and costly hardware for interchanging wavelengths (or time slots).

As far as related work is concerned, a large body of research has been devoted to the subject of permutation routing and scheduling including past work on non-multiplexed networks [46, 47, 48]. Several recent studies of WDM networks have considered the bounds on the number of wavelengths required to make a network either nonblocking (rearrangeably and/or wide-sense) (see for example, [5, 49]), or be able to establish an arbitrary set of connections [50, 51, 52]. In these studies, however, PM has been assumed almost exclusively, and in addition, these studies either obtained *asymptotic* bounds for general networks with a bounded nodal degree or for specific ones such as hypercube-based networks, or obtained probablistic or approximate bounds as a function of the optimal number of wavelengths needed (or the maximal number of connections sharing a common link, also called the *maximum load*) for specific networks such as trees and meshes.

Another aspect in which our work differs from others is that we have considered the case in which a network is blocking as a result of having an insufficient multiplexing degree, and compared the schedule lengths of a permutation resulted from using LM and PM. Additional results on comparing the effectiveness of PM and LM for both off-line and on-line permutation embedding and scheduling were reported in [3, 12].

5 Conclusion

Because there are various voice and data communication applications, a WDM optical layer needs to provide different class of services. In this paper, it is suggested that this be accomplished by partitioning and allocating resources appropriately to form several virtual optical networks (VONs). Specifically, a VON supporting dynamic traffic can be allocated with a small subset of wavelengths, fast switches and wavelength converters. On the other hand, a VON supporting static traffic can use relatively slower switches and do without wavelength converters. We have proposed a new framework for polymorphic control which includes both On-demand-reconfiguration in some virtual otical networks (VONs) and Self-reconfiguration in other VONs. In respect to the former, we have described several unique distributed wavelength reservation protocols

	Rearrangeably Nonblocking		Wide-sense Nonblocking	
	LM	PM	LM	PM
$Linear\ Arrays)$	$\frac{N}{2}$	$\frac{N}{2}$	$\frac{N}{2}$	$\frac{N}{2}$
$Unidirectional\ Rings$	$N-1$	N	$N-1$	N
$Bidirectional\ Rings$	$\frac{N}{2}-1$	$\left[\frac{N}{2}-1, \frac{N}{2}\right]$	$\frac{N}{2}-1$	$\left[\frac{N}{2}-1, \frac{N}{2}\right]$
$n \times n\ Meshes$	$[n-1, n]$	$[n-1, n]$	$[n-1, n]$	$[n-1, 2n-3]$
$n \times n\ Tori$	$\left[\frac{n}{2}-1, \frac{n}{2}\right]$	$\left[\frac{n}{2}-1, \frac{n}{2}\right]$	$\left[\frac{n}{2}-1, \frac{n}{2}\right]$	$\left[\frac{n}{2}-1, n-1\right]$
n$-dim.\ hypercubes$	$2^{\frac{n}{2}}$	$2^{\frac{n}{2}}$	$2^{\frac{n}{2}}$	$\left[2^{\frac{n}{2}}, \frac{3}{2} \cdot 2^{\frac{n}{2}} - 1\right]$

Table 1: The values of K_{LM} and K_{PM}.

and as well as the novel concept of optical burst switching (OBS) which are suitable for bursty traffic and short-lived connections. In respect to the latter, we have presented efficient scheduling algorithms for WDM rings and obtained concrete results on the problem of embedding permutations in a class of regular networks including rings and meshes.

In addition, we have addressed several issues related to the principle of cost-effective design of WDM networks. For example, we have identified the trade-offs between distributed and centralized control, as well as between control complexity and bandwidth utilization, discussed the benefit (and cost) of wavelength conversions as well as E/O and O/E conversions, and analyzed ways to achieve resource balance. The proposed framework of polymorphic control and the principle of cost-effective design can help realize the vision of building a flexible, efficient and bandwidth-abundant fiber-optic network infrastructure, which is capable of eliminating redundant and expensive hardware and software, and providing ubiquitous services to applications, IP, ATM and other existing (e.g. SONET) and future protocols.

REFERENCES

[1] Y. Mei and C. Qiao, "Efficient distributed control protocols for WDM optical networks," in *Proc. Int'l Conference on Computer Communication and Networks*, pp. 150–153, Sept. 1997.

[2] C. Qiao and Y. Mei, "Wavelength reservation under distributed control," in *IEEE/LEOS Broadband Optical Networks*, pp. 45–46, Aug. 1996.

[3] C. Qiao and Y. Mei, "A comparative study of cost-effective multiplexing approaches in optical networks," in *International Conference on Massively Parallel Processing Using Optical Interconnections (MPPOI)*, pp. 24–31, Oct. 1996.

[4] C. Qiao and R. Melhem, "Reducing communication latency with path multiplexing in optically interconnected multiprocessor systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 8, no. 2, pp. 97–108, 1997.

[5] A. Aggarwal et al, "Efficient routing and scheduling in optical networks," *Proc. of the ACM-SIAM Symp. on discrete algorithms*, pp. 412–423, 1993.

[6] K. Bala, T. Stern, and K. Bala, "Algorithms for routing in a linear lightwave network," in *Proceedings of the IEEE InfoCom*, pp. 1–9, 1991.

[7] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: an approach to high-bandwidth optical WANs," *IEEE Transactions on Communications*, vol. 40, pp. 1171–1182, July 1992.

[8] K. Lee and V. O. Li, "A circuit rerouting algorithm for all-optical wide-area networks," in *Proceedings of IEEE Infocom*, pp. 954–961, 1994.

[9] R. Ramaswami and K. Sivarajan, "Optimal routing and wavelength assignment in all-optical networks," in *Proceedings of IEEE Infocom*, pp. 970–979, June 1994.

[10] R. Barry and P. Humblet, "On the number of wavelengths and switches in all-optical networks," *IEEE Transactions on Communications*, vol. 42, pp. 583–591, 1994.

[11] R. Pankaj and R. Gallager, "Wavelength requirements of all-optical networks," *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 269–280, 1995.

[12] C. Qiao and Y. Mei, "On the multiplexing degree required to embed permutations in a class of interconnection networks," in *Proceedings of the IEEE Symp. High Performance Computer Architecture*, pp. 118–129, Feb. 1996. (A comprehensive version is to appear in IEEE/ACM Trans. on Networking (*ToN*)).

[13] R. A. Barry and P. A. Humblet, "Models of blocking probability in all-optical networks with and without wavelength changers," in *Proceedings of IEEE Infocom*, pp. 402–412, Apr. 1995.

[14] A. Birman, "Computing approximate blocking probabilities for a class of all-optical networks," in *Proceedings of IEEE Infocom*, pp. 651–657, Apr. 1995.

[15] M. Kovacevic and A. Acampora, "On wavelength translation in all-optical networks," in *Proceedings of IEEE Infocom*, pp. 413–422, Apr. 1995.

[16] S. Subramaniam, M. Azizoglu, and A. Somani, "Connectivity and sparse wavelength conversion in wavelength-routing networks," in *Proceedings of IEEE Infocom*, pp. 148–155, Mar. 1996.

[17] J. Yates et. al, "Limited-range wavelength translation in all-optical networks," in *Proceedings of IEEE Infocom*, pp. 954–961, Mar. 1996.

[18] C. Qiao and R. Melhem, "Reconfiguration with time-division multiplexed MINs for multiprocessor communications," *IEEE Transactions on Parallel and Distributed Systems*, vol. 5, no. 4, pp. 337–352, 1994.

[19] X. Yuan, R. Gupta, and R. Melhem, "Distributed control in optical WDM networks," in *IEEE MILCOM*, pp. 100–104, Oct. 1996.

[20] R. Ramaswami and A. Segall, “Distributed network control for wavelength routed optical networks,” in *Proceedings of IEEE Infocom*, pp. 138–147, Mar. 1996.

[21] A. Sengupta et al., “On an adaptive algorithm for routing in all-optical networks,” in *SPIE Proceedings, All Optical Communication Systems: Architecture, Control and Network Issues*, vol. 3230, pp. 288–297, Nov. 1997.

[22] A. Erramilli, O. Narayan, and W. Willinger, “Experimental queueing analysis with long-range dependent packet traffic,” *IEEE/ACM Transactions on Networking*, vol. 4, no. 2, pp. 209–223, 1996.

[23] W. Leland, M. Taqqu, M. Willinger, and D. Wilson, “On the self-similar nature of Ethernet traffic (extended version),” *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 1–15, 1994.

[24] V. Paxon and S. Floyd, “Wide area traffic: the failure of Poisson modeling,” *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 226–244, 1995.

[25] M. Willinger, M. Taqqu, R. Sherman, and D. Wilson, “Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level,” *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 71–86, 1997.

[26] C. Qiao and M. Yoo, “Optical burst switching (OBS) - a new paradigm for an Optical Internet,” *J. High Speed Networks (JHSN)*, vol. 8, no. 1, pp. 69–84, 1999.

[27] M. Yoo and C. Qiao, “Just-enough-time(JET): a high speed protocol for bursty traffic in optical networks,” in *IEEE/LEOS Technologies for a Global Information Infrastructure*, Aug. 1997.

[28] M. Yoo and C. Qiao, “A new optical burst switching protocol for supporting quality of service,” in *SPIE Proceedings, All Optical Networking: Architecture, Control and Management Issues*, vol. 3531, pp. 396–405, Nov. 1998.

[29] E. Varvarigos and V. Sharma, “The ERVC protocol for the Thunder and Lightning network: operation, formal description and proof of correctness,” Tech. Rep. CIPR 95-05, ECE Dept, UC Santa Barbara, June 1995.

[30] M. Yoo, M. Jeong, and C. Qiao, “A high-speed protocol for bursty traffic in optical networks,” in *SPIE Proceedings, All Optical Communication Systems: Architecture, Control and Network Issues*, vol. 3230, pp. 79–90, Nov. 1997.

[31] P. E. Boyer and D. P. Tranchier, “A reservation principle with applications to the ATM traffic control,” *Computer Networks and ISDN Systems*, vol. 24, pp. 321–334, 1992.

[32] G. C. Hudek and D. J. Muder, “Signaling analysis for a multi-switch all-optical network,” in *IEEE International Conferenece in Communications*, pp. 1206–1210, June 1995.

[33] H. Shimonishi, T. Takine, M. Murata, and H. Miyahara, “Performance analysis of fast reservation protocol with generalized bandwidth reservation method,” in *Proceedings of IEEE Infocom*, vol. 2, pp. 758–767, 1996.

[34] J. S. Turner, “Managing bandwidth in ATM networks with bursty traffic,” *IEEE Network*, pp. 50–58, Sept. 1992.

[35] I. Widjaja, “Performance analysis of burst admission-control protocols,” *IEE proceedings-communications*, vol. 142, pp. 7–14, Feb. 1995.

[36] X. Zhang and C. Qiao, “An effective and comprehensive solution to traffic glooming and wavelength assignment in SONET/WDM rings,” in *SPIE Proceedings, All Optical Networking: Architecture, Control and Management Issues*, vol. 3531, pp. 221–231, Nov. 1998.

[37] J.Simmons, E. Goldstein, and A. Saleh, “On the value of wavelength-add/drop in WDM rings with uniform traffic,” in *Proc. Optical Fiber Communication Conference*, pp. 361–362, Mar. 1998.

[38] O. Gerstel, P. Lin, and G. Sasaki, “Wavelength assignment in a WDM ring to minimize cost of embedded SONET rings,” in *Proceedings of IEEE Infocom*, pp. 94–101, Mar. 1998.

[39] O. Gerstel, R. Ramaswami, and G. Sasaki, “Cost effective traffic grooming in WDM rings,” in *Proceedings of IEEE Infocom*, pp. 69–77, Mar. 1998.

[40] E. Modiano and A. Chiu, “Traffic grooming algorithms for minimizing electronic multiplexing costs in unidirectional SONET/WDM ring networks,” in *CISS'98*, Mar. 1998.

[41] C. Qiao and X. Zhang, “Optimal design of WDM ring networks via resource-balance,” in *IEEE/LEOS Broadband Optical Networks*, Aug. 1996. (a comprehensive version is to appear in IEEE/ACM ToN).

[42] C. Qiao, X. Zhang, and L. Zhou, “Scheduling all-to-all connections in WDM rings,” in *SPIE Proceedings, All Optical Communication Systems: Architecture, Control and Network Issues*, vol. 2919, pp. 218–229, Nov. 1996.

[43] A. Elrefaie, “Multiwavelength survivable ring network architectures,” in *Proc. Int'l Conference on Communication*, pp. 1245–1251, 1993.

[44] J. Bermond et. al, “Efficient collective communication in optical networks,” in *Proc. of ICALP*, pp. 574–585, 1996.

[45] X. Zhang and C. Qiao, “Pipelined transmission scheduling in all-optical TDM/WDM rings,” in *Proc. Int'l Conference on Computer Communication and Networks*, pp. 144–149, Sept. 1997.

[46] S. Choi and A. Somani, “Rearrangeable circuit-switched hypercube architecture for routing permutations,” *Journal of Parallel and Distributed Computing*, vol. 19, pp. 125–133, 1993.

[47] T. Szymanski, “On the permutation capability of a circuit-switched hypercube,” in *Proceedings of International Conference on Parallel Processing*, pp. 103–110, 1989.

[48] A. Youssef, “Off-line permutation scheduling on circuit-switched fixed routing networks,” in *Proceedings of the Symp. on Frontiers of Massively Parallel Computation*, pp. 389–396, 1992.

[49] Y. Aumann and Y. Rabani, “Improved bounds for all optical routing,” in *Proc. of ACM-SIAM Symp. on Discrete Algorithms*, pp. 567–576, 1995.

[50] V. Kumar and E. Schwabe, “Improved access to optical bandwidth in trees,” in *Proc. of ACM-SIAM Symp. on Discrete Algorithms*, pp. 437–444, Jan. 1997.

[51] M. Mihail, C. Kaklamanis, and S. Rao, “Efficient access to optical bandwidth,” in *Proceedings of ACM Symp. on Theory of Computing*, pp. 548–557, 1995.

[52] P. Raghavan and E. Upfal, “Efficient routing in all-optical networks,” in *Proceedings of ACM Symp. on Theory of Computing*, pp. 134–143, 1994.