A Tutorial on Optical Networks

George N. Rouskas and Harry G. Perros

Department of Computer Science, North Carolina State University, Raleigh, NC, USA
rouskas,hp@csc.ncsu.edu

Abstract. In this half-day tutorial, we present the current state-of-the-art in optical networks. We begin by discussing the various optical devices used in optical networks. Then, we present wavelength-routed networks, which is currently the dominant architecture for optical networks. We discuss wavelength allocation policies, calculation of call blocking probabilities, and network optimization techniques. Subsequently, we focus on the various protocols that have been proposed for wavelength-routed networks. Specifically, we present a framework for IP over optical networks, MPLS, LDP, CR-LDP, and GMPLS. Next, we discuss optical packet switching and optical burst switching, two new emerging and highly promising technologies.

1 Introduction

Over the last few years we have witnessed a wide deployment of point-to-point wavelength division multiplexing (WDM) transmission technology in the Internet infrastructure. The corresponding massive increase in network bandwidth due to WDM has heightened the need for faster switching at the core of the network. At the same time, there has been a growing effort to enhance the Internet Protocol (IP) to support traffic engineering [1,2] as well as different levels of Quality of Service (QoS) [3]. Label Switching Routers (LSRs) running Multi-Protocol Label Switching (MPLS) [4,5] are being deployed to address the issues of faster switching, QoS support, and traffic engineering. On one hand, label switching simplifies the forwarding function, thereby making it possible to operate at higher data rates. On the other hand, MPLS enables the Internet architecture, built upon the connectionless Internet Protocol, to behave in a connection-oriented fashion that is more conducive to supporting QoS and traffic engineering.

The rapid advancement and evolution of optical technologies makes it possible to move beyond point-to-point WDM transmission systems to an all-optical backbone network that can take full advantage of the available bandwidth. Such a network consists of a number of optical cross-connects (OXCs) arranged in some arbitrary topology, and its main function is to provide interconnection to a number of IP/MPLS subnetworks. Each OXC can switch the optical signal coming in on a wavelength of an input fiber link to the same wavelength in an output fiber link. The OXC may also be equipped with converters that permit it to switch the optical signal on an incoming wavelength of an input fiber to

E. Gregori et al. (Eds.): Networking 2002 Tutorials, LNCS 2497, pp. 155–193, 2002.

some other wavelength on an output fiber link. The main mechanism of transport in such a network is the lightpath (also referred to as λ-channel), an optical communication channel established over the network of OXCs which may span a number of fiber links (physical hops). If no wavelength converters are used, a lightpath is associated with the same wavelength on each hop. This is the well-known wavelength continuity constraint. Using converters, a different wavelength on each hop may be used to create a lightpath. Thus, a lightpath is an end-to-end optical connection established between two subnetworks attached to the optical backbone.

Currently, there is tremendous interest within both the industry and the research community in optical networks in which OXCs provide the switching functionality. The Internet Engineering Task Force (IETF) is investigating the use of Generalized MPLS (GMPLS) [6] and related signaling protocols to set up and tear down lightpaths. GMPLS is an extension of MPLS that supports multiple types of switching, including switching based on wavelengths usually referred to as Multi-Protocol Lambda Switching (MPλS). With GMPLS, the OXC backbone and the IP/MPLS subnetworks will share common functionality in the control plane, making it possible to seamlessly integrate all-optical networks within the overall Internet infrastructure. Also, the Optical Domain Service Interconnection (ODSI) initiative (which has completed its work) and the Optical Internetworking Forum (OIF) are concerned with the interface between an IP/MPLS subnetwork and the OXC to which it is attached as well as the interface between OXCs, and have several activities to address MPLS over WDM issues [7]. Optical networks have also been the subject of extensive research [8] investigating issues such as virtual topology design [9,10], call blocking performance [11,12], protection and restoration [13,14], routing algorithms and wavelength allocation policies [15,16,17], and the effect of wavelength conversion [18,19,20], among others.

The tutorial is organized as follows. Section 2 introduces the basic elements of the optical network architecture, and Section 3 presents the routing and wavelength assignment problem, the fundamental control problem in optical networks. Section 4 discusses standardization activities under way for optical networks, with an emphasis on control plane issues. Section 5 discusses a framework for IP over optical networks, MPLS, the signaling protocols LDP and CR-LDP, and GMPLS. Section 6 describes optical packet switching, and finally, Section 7 describes an emerging technology, optical burst switching.

2 Wavelength Routing Network Architecture

The architecture for wide-area WDM networks that is widely expected to form the basis for a future all-optical infrastructure is built on the concept of *wavelength routing*. A wavelength routing network, shown in Figure 1, consists of *optical cross-connects (OXCs)* connected by a set of fiber links to form an arbitrary mesh topology. The services that a wavelength routed network offers to attached client subnetworks are in the form of *logical* connections implemented

using *lightpaths*. Lightpaths are clear optical paths which may traverse a number of fiber links in the optical network. Information transmitted on a lightpath does not undergo any conversion to and from electrical form within the optical network, and thus, the architecture of the OXCs can be very simple because they do not need to do any signal processing. Furthermore, since a lightpath behaves as a literally transparent "clear channel" between the source and destination subnetwork, there is nothing in the signal path to limit the throughput of the fibers.

The OXCs provide the switching and routing functions for supporting the logical data connections between client subnetworks. An OXC takes in an optical signal at each of the wavelengths at an input port, and can switch it to a particular output port, independent of the other wavelengths. An OXC with N input and N output ports capable of handling W wavelengths per port can be thought of as W independent $N \times N$ optical switches. These switches have to be preceded by a wavelength demultiplexer and followed by a wavelength multiplexer to implement an OXC, as shown in Figure 2. Thus, an OXC can cross-connect the different wavelengths from the input to the output, where the connection pattern of each wavelength is independent of the others. By appropriately configuring the OXCs along the physical path, logical connections (lightpaths) may be established between any pair of subnetworks.

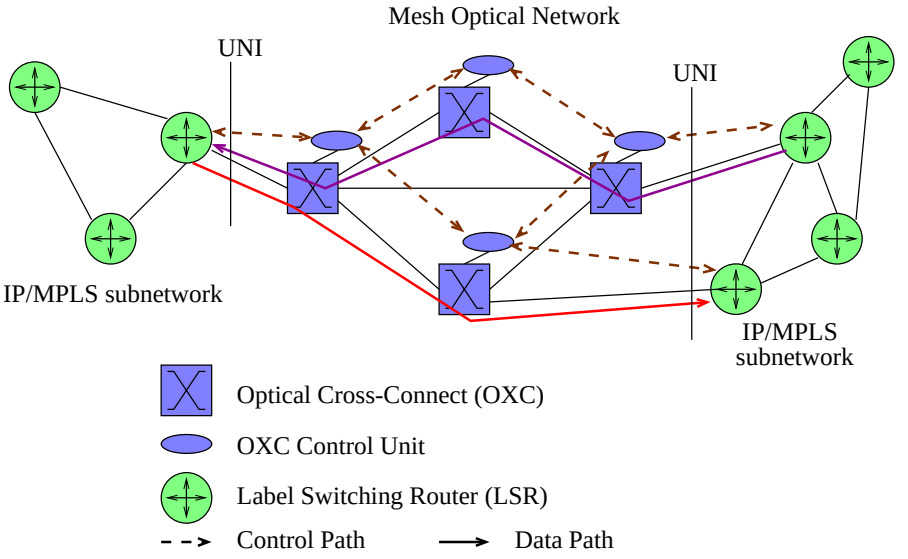

Fig. 1. Optical network architecture

As Figure 1 illustrates, each OXC has an associated *electronic* control unit attached to one of its input/output ports. The control unit is responsible for control and management functions related to setting up and tearing down lightpaths;

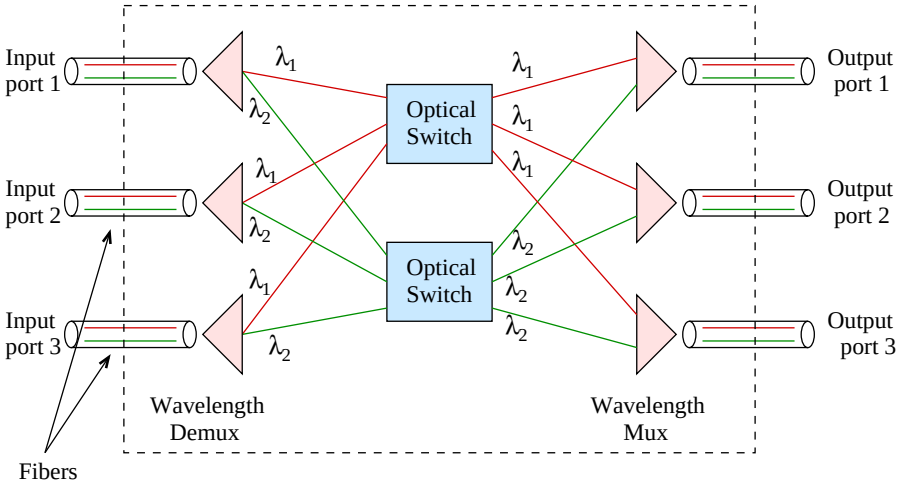

Fig. 2. 3 × 3 optical cross-connect (OXC) with two wavelengths per fiber

these functions are discussed in detail in Section 4. In particular, the control unit communicates directly with its OXC, and is responsible for issuing configuration commands to the OXC in order to implement a desired set of lightpath connections; this communication takes place over a (possibly proprietary) interface that depends on the OXC technology. The control unit also communicates with the control units of adjacent OXCs or with attached client subnetworks over *single-hop* lightpaths as shown in Figure 1. These lightpaths are typically implemented over administratively configured ports at each OXC and use a separate control wavelength at each fiber. Thus, we distinguish between the paths that data and control signals take in the optical network: data lightpaths originate and terminate at client subnetworks and transparently traverse the OXCs, while control lightpaths are electronically terminated at the control unit of each OXC. Communication on the control lightpaths uses a standard signaling protocol (e.g., GMPLS), as well as other standard protocols necessary for carrying out important network functions including label distribution, routing, and network state dissemination. Standardization efforts are crucial to the seamless integration of multi-vendor optical network technology, and are discussed in Section 4.

Client subnetworks attach to the optical network via edge nodes which provide the interface between non-optical devices and the optical core. This interface is denoted as UNI (user-to-network interface) in Figure 1. The edge nodes act as the terminating points (sources and destinations) for the optical signal paths; the communication paths may continue outside the optical network in electrical form. In Figure 1, only the label switching routers (LSRs) of the two IP/MPLS subnetworks which are directly attached to an OXC implement the UNI and may originate or terminate lightpaths. For the remainder of this chapter we will make the assumption that client subnetworks run the IP/MPLS protocols. This assumption reflects the IP-centric nature of the emerging control architecture for

optical networks [21]. However, edge nodes supporting any network technology (including ATM switches and SONET/SDH devices) may connect to the optical network as long as an appropriate UNI is defined and implemented.

In [22,23], the concept of a lightpath was generalized into that of a *light-tree*, which, like a lightpath, is a clear channel originating at a given source node and implemented with a single wavelength. But unlike a lightpath, a light-tree has multiple destination nodes, hence it is a point-to-multipoint channel. The physical links implementing a light-tree form a tree, rooted at the source node, rather than a path in the physical topology, hence the name. Light-trees may be implemented by employing optical devices known as *power splitters* [24] at the OXCs. A power splitter has the ability to split an incoming signal, arriving at some wavelength λ, into up to m outgoing signals, $m \geq 2$; m is referred to as the *fanout* of the power splitter. Each of these m signals is then independently switched to a different output port of the OXC. Note that due to the splitting operation and associated losses, the optical signals resulting from the splitting of the original incoming signal must be amplified before leaving the OXC. Also, to ensure the quality of each outgoing signal, the fanout m of the power splitter may have to be limited to a small number. If the OXC is also capable of wavelength conversion, each of the m outgoing signal may be shifted, independently of the others, to a wavelength different than the incoming wavelength λ. Otherwise, all m outgoing signals must be on the same wavelength λ.

An attractive feature of light-trees is the inherent capability for performing multicasting in the optical domain (as opposed to performing multicasting at a higher layer, e.g., the network layer, which requires electro-optic conversion). Such wavelength routed light-trees are useful for transporting high-bandwidth, real-time applications such as high-definition TV (HDTV). Therefore, OXCs equipped with power splitters will be referred to as *multicast-capable* OXCs (MC-OXCs). Note that, just like with converter devices, incorporating power splitters within an OXC is expected to increase the network cost because of the need for power amplification and the difficulty of fabrication.

3 Routing and Wavelength Assignment (RWA)

A unique feature of optical WDM networks is the tight coupling between routing and wavelength selection. As can be seen in Figure 1, a lightpath is implemented by selecting a path of physical links between the source and destination edge nodes, and reserving a particular wavelength on each of these links for the lightpath. Thus, in establishing an optical connection we must deal with both routing (selecting a suitable path) and wavelength assignment (allocating an available wavelength for the connection). The resulting problem is referred to as the *routing and wavelength assignment (RWA)* problem [17], and is significantly more difficult than the routing problem in electronic networks. The additional complexity arises from the fact that routing and wavelength assignment are subject to the following two constraints:

1. *Wavelength continuity constraint:* a lightpath must use the same wavelength on all the links along its path from source to destination edge node.
2. *Distinct wavelength constraint:* all lightpaths using the same link (fiber) must be allocated distinct wavelengths.

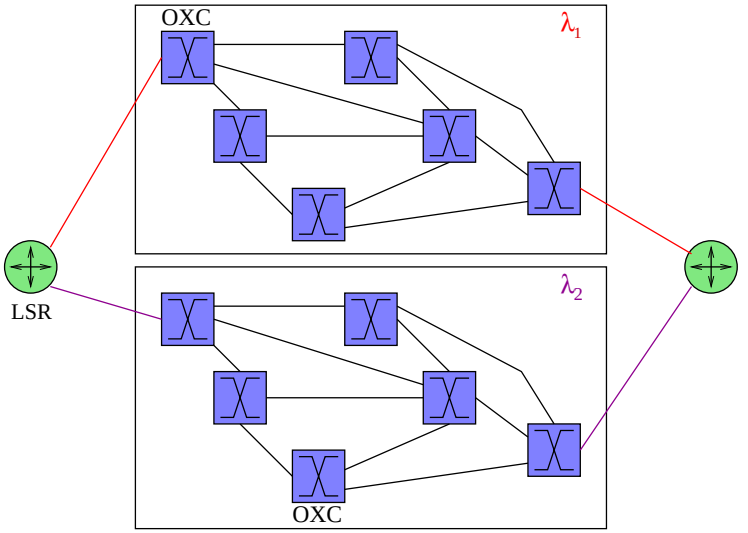

Fig. 3. The RWA problem with two wavelengths per fiber

The RWA problem in optical networks is illustrated in Figure 3, where it is assumed that each fiber supports two wavelengths. The effect of the wavelength continuity constraint is represented by replicating the network into as many copies as the number of wavelengths (in this case, two). If wavelength i is selected for a lightpath, the source and destination edge node communicate over the i-th copy of the network. Thus, finding a path for a connection may potentially involve solving W routing problems for a network with W wavelengths, one for each copy of the network.

The wavelength continuity constraint may be relaxed if the OXCs are equipped with *wavelength converters* [18]. A wavelength converter is a single input/output device that converts the wavelength of an optical signal arriving at its input port to a different wavelength as the signal departs from its output port, but otherwise leaves the optical signal unchanged. In OXCs without a wavelength conversion capability, an incoming signal at port p_i on wavelength λ can be optically switched to any port p_j, but must leave the OXC on the same wavelength λ. With wavelength converters, this signal could be optically switched to any port p_j on some other wavelength λ'. That is, wavelength conversion allows a lightpath to use different wavelengths along different physical links.

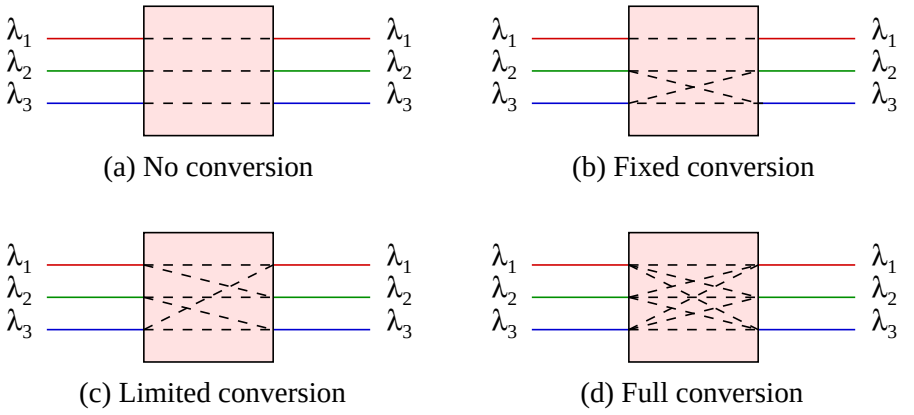

Fig. 4. Wavelength conversion

Different levels of wavelength conversion capability are possible. Figure 4 illustrates the differences for a single input and single output port situation; the case for multiple ports is more complicated but similar. *Full wavelength conversion* capability implies that any input wavelength may be converted to any other wavelength. *Limited wavelength conversion* [25] denotes that each input wavelength may be converted to any of a specific set of wavelengths, which is not the set of all wavelengths for at least one input wavelength. A special case of this is *fixed wavelength conversion*, where each input wavelength can be converted to exactly one other wavelength. If each wavelength is "converted" only to itself, then we have no conversion.

The advantage of full wavelength conversion is that it removes the wavelength continuity constraint, making it possible to establish a lightpath as long as each link along the path from source to destination has a free wavelength (which could be different for different links). As a result, the RWA problem reduces to the classical routing problem, that is, finding a suitable path for each connection in the network. Referring to Figure 3, full wavelength conversion collapses the W copies of the network into a single copy on which the routing problem is solved. On the other hand, with limited conversion, the RWA problem becomes more complex than with no conversion. To see why, note that employing limited conversion at the OXCs introduces links between *some* of the network copies of Figure 3. For example, if wavelength λ_1 can be converted to wavelength λ_2 but not to wavelength λ_3, then links must be introduced from each OXC in copy 1 of the network to the corresponding OXC in copy 2, but not to the corresponding OXC in copy 3. When selecting a path for the connection, at each OXC there is the option of remaining at the same network copy or moving to another one, depending on the conversion capability of the OXC. Since the number of alternatives increases exponentially with the number of OXCs that need to be traversed, the complexity of the RWA problem increases accordingly.

Wavelength conversion (full or limited) increases the routing choices for a given lightpath (i.e., makes more efficient use of wavelengths), resulting in better performance. Since converter devices increase network cost, a possible middle ground is to use *sparse conversion*, that is, to employ converters in some, but not all, OXCs in the network. In this case, a lightpath must use the same wavelength along each link in a segment of its path between OXCs equipped with converters, but it may use a different wavelength along the links of another such segment. It has been shown that implementing full conversion at a relatively small fraction of the OXCs in the network is sufficient to achieve almost all the benefits of conversion [11,19].

With the availability of MC-OXCs and the existence of multicast traffic demands, the problem of establishing light-trees to satisfy these demands arises. We will call this problem the *multicast routing and wavelength assignment (MC-RWA)* problem. MC-RWA bears many similarities to the RWA problem discussed above. Specifically, the tight coupling between routing and wavelength assignment remains, and even becomes stronger: in the absence of wavelength conversion the same wavelength must be used by the multicast connection not just along the links of a single path but along all the links of the light-tree. Since the construction of optimal trees for routing multicast connections is by itself a hard problem [26], the combined MC-RWA problem becomes even harder.

Routing and wavelength assignment is the fundamental control problem in optical WDM networks. Since the performance of a network depends not only on its physical resources (e.g., OXCs, converters, fibers links, number of wavelengths per fiber, etc.) but also on how it is controlled, the objective of an RWA algorithm is to achieve the best possible performance within the limits of physical constraints. The RWA (and MC-RWA) problem can be cast in numerous forms. The different variants of the problem, however, can be classified under one of two broad versions: a static RWA, whereby the traffic requirements are known in advance, and a dynamic RWA, in which a sequence of lightpath requests arrive in some random fashion. The static RWA problem arises naturally in the design and capacity planning phase of architecting an optical network, and is discussed in Section 3.1. The dynamic RWA problem is encountered during the real-time network operation phase and involves the dynamic provisioning of lightpaths; this issue is addressed in Section 3.2.

3.1 Static RWA

If the traffic patterns in the network are reasonably well-known in advance and any traffic variations take place over long time scales, the most effective technique for establishing optical connections (lightpaths) between client subnetworks is by formulating and solving a static RWA problem. Therefore, static RWA is appropriate for provisioning a set of semipermanent connections. Since these connections are assumed to remain in place for relatively long periods of time, it is worthwhile to attempt to optimize the way in which network resources (e.g., physical links and wavelengths) are assigned to each connection, even though optimization may require a considerable computational effort. Because off-line

algorithms have knowledge of the entire set of demands (as opposed to on-line algorithms that have no knowledge of future demands), they make more efficient use of network resources and project a lower overall capacity requirement.

Physical Topology Design. In this phase the network operator has a demand forecast and must decide on a topology to connect client subnetworks through OXCs. This step includes the sizing of links (e.g., determining the number of wavelength channels and the capacity of each channel) and OXCs (e.g, determining the number of ports), as well as the placement of resources such as amplifiers, wavelength converters, and power splitters. Moreover, to deal with link or OXC failures, it is desirable to ensure that there are at least two (or three) paths between any pair of OXCs in the network, i.e., that the graph corresponding to the physical topology of the optical network is two- or three-connected. Often, geographical or administrative considerations may impose further constraints on the physical topology.

If a network does not already exist, the physical topology must be designed from scratch. Obviously, the outcome of this step strongly depends on the accuracy of the demand forecast, and the potential for error is significant when designers have to guess the load in a new network. Therefore, many providers take a cautious approach by initially building a skeleton network and adding new resources as necessary by actual user demand. In this *incremental* network design, it is assumed that sets of user demands arrive over multiple time periods. Resources (e.g., OXCs, fiber links, wavelength channels) are added incrementally to satisfy each new set of demands, in a way that the additional capacity required is minimized.

A physical topology design problem was considered in [27]. Given a number of LSRs and a set of lightpaths to be set up among pairs of LSRs, the objective was to determine the two-connected physical topology with the minimum number of OXCs to establish all the lightpaths (this is a combined physical/virtual topology design problem in that the routing and wavelength assignment for the lightpaths is also determined). An iterative solution approach was considered, whereby a genetic algorithm was used to iterate over the space of physical topologies, and heuristics were employed for routing and wavelength assignment on a given physical topology (refer to the next subsection for details on RWA heuristics). The algorithm was applied to networks with up to 1000 LSRs and tens of thousands of lightpaths, and provided insight into the capacity requirements for realistic optical networks. For example, it was shown that the number of OXCs increases much slower than the number of LSRs, and also that the number of OXCs increases only moderately as the number of lightpaths increases by a factor of two or three. These results indicate that optical networks to interconnect a large number of LSRs can be built to provide rich connectivity with moderate cost.

Other studies related to capacity planning have looked into the problem of optimally placing network resources such as converters or power splitters (for multicast). The problem of converter placement was addressed in [11,28], and

optimal [28] (for uniform traffic only) or near-optimal greedy [11] algorithms (for general traffic patterns) were developed. While both studies established that a small number of converters (approximately 30% of the number of OXCs) is sufficient, the results in [11] demonstrate that (a) the optimal placement of converters is extremely sensitive to the actual traffic pattern, and (b) an incremental approach to deploying converters may not lead to optimal (or near-optimal) results. The work in [29] considered the problem of optimally allocating the multicast-capable OXCs (MC-OXCs) to establish light-trees, and a greedy heuristic was proposed. It was found that there is little performance improvement if more than 50% of the OXCs in the network are multicast-capable, and that the optimal location of MC-OXCs depends on the traffic pattern.

Overall, the physical topology design problem is quite complex because the topology, the link and OXC capacities, and the number and location of optical devices such as converters and amplifiers strongly depends on the routing of lightpaths and the wavelength assignment strategy. If we make the problem less constrained, allowing the topology, routing, wavelength assignment, link capacity, etc., to change, the problem becomes very hard because these parameters are coupled in complicated ways. In practice, the topology may be constrained by external factors making the problem easier to deal with; for instance the existence of a deployed fiber infrastructure may dictate the location of OXCs and the links between them. However, the area of physical topology design for optical networks remains a rich area for future research.

Virtual Topology Design. A solution to the static RWA problem consists of a set of long-lived lightpaths which create a *logical* (or *virtual*) topology among the edge nodes. This virtual topology is embedded onto the physical topology of optical fiber links and OXCs. Accordingly, the static RWA problem is often referred to as the *virtual topology design* problem [9]. In the virtual topology, there is a directed link from edge node s to edge node d if a lightpath originating at s and terminating at d is set up (refer also to Figure 1), and edge node s is said to be "one hop away" from edge node d in the virtual topology, although the two nodes may be separated by a number of physical links. The type of virtual topology that can be created is usually constrained by the underlying physical topology. In particular, it is generally not possible to implement fully connected virtual topologies: for N edge nodes this would require each edge node to maintain $N-1$ lightpaths and the optical network to support a total of $N(N-1)$ lightpaths. Even for modest values of N, this degree of connectivity is beyond the reach of current optical technology, both in terms of the number of wavelengths that can be supported and in terms of the optical hardware (transmitters and receivers) required at each edge node.

In its most general form, the RWA problem is specified by providing the physical topology of the network and the traffic requirements. The physical topology corresponds to the deployment of cables in some existing fiber infrastructure, and is given as a graph $G_p(V, E_p)$, where V is the set of OXCs and E_p is the set of fibers that interconnect them. The traffic requirements are specified in a

traffic matrix $\mathbf{T} = [\rho p_{sd}]$, where ρp_{sd} is a measure of the long-term traffic flowing from source edge node s to destination edge node d [30]. Quantity ρ represents the (deterministic) total offered load to the network, while the p_{sd} parameters define the distribution of the offered traffic.

Routing and wavelength assignment are considered together as an optimization problem using integer programming formulations. Usually, the objective of the formulation is to minimize the maximum congestion level in the network subject to network resource constraints [9,10]. While other objective functions are possible, such as minimizing the average weighted number of hops or minimizing the average packet delay, minimizing network congestion is preferable since it can lead to linear programming (ILP) formulations. While we do not present the RWA problem formulation here, the interested reader may refer to [30,9,10]. These formulations turn out to have extremely large numbers of variables, and are intractable for large networks. This fact has motivated the development of heuristic approaches for finding good solutions efficiently.

Before we describe the various heuristic approaches, we note that the static RWA problem can be logically decomposed into four subproblems. The decomposition is approximate or inexact, in the sense that solving the subproblems in sequence and combining the solutions may not result in the optimal solution for the fully integrated problem, or some later subproblem may have no solution given the solution obtained for an earlier subproblem, so no solution to the original problem may be obtained. However, the decomposition provides insight into the structure of the RWA problem and is a first step towards the design of effective heuristics. Assuming no wavelength conversion, the subproblems are as follows.

1. **Topology Subproblem:** Determine the logical topology to be imposed on the physical topology, that is, determine the lightpaths in terms of their source and destination edge nodes.
2. **Lightpath Routing Subproblem:** Determine the physical links which each lightpath consists of, that is, route the lightpaths over the physical topology.
3. **Wavelength Assignment Subproblem:** Determine the wavelength each lightpath uses, that is, assign a wavelength to each lightpath in the logical topology so that wavelength restrictions are obeyed for each physical link.
4. **Traffic Routing Subproblem:** Route packet traffic between source and destination edge nodes over the logical topology obtained.

A large number of heuristic algorithms have been developed in the literature to solve the general static RWA problem discussed here or its many variants. Overall, however, the different heuristics can be classified into three broad categories: (1) algorithms which solve the overall ILP problem sub-optimally, (2) algorithms which tackle only a subset of the four subproblems, and (3) algorithms which address the problem of embedding regular logical topologies onto the physical topology.

Suboptimal solutions can be obtained by applying classical tools developed for complex optimization problems directly to the ILP problem. One technique

is to use LP-relaxation followed by rounding [31]. In this case, the integer constraints are relaxed creating a non-integral problem which can be solved by some linear programming method, and then a rounding algorithm is applied to obtain a new solution which obeys the integer constraints. Alternatively, genetic algorithms or simulated annealing [32] can be applied to obtain locally optimal solutions. The main drawback of these approaches is that it is difficult to control the quality of the final solution for large networks: simulated annealing is computationally expensive and thus, it may not be possible to adequately explore the state space, while LP-relaxation may lead to solutions from which it is difficult to apply rounding algorithms.

Another class of algorithms tackles the RWA problem by initially solving the first three subproblems listed above; traffic routing is then performed by employing well-known routing algorithms on the logical topology. One approach for solving the three subproblems is to maximize the amount of traffic that is carried on one-hop lightpaths, i.e., traffic that is routed from source to destination edge node directly on a lightpath. A greedy approach taken in [33] is to create lightpaths between edge nodes in order of decreasing traffic demands as long as the wavelength continuity and distinct wavelength constraints are satisfied. This algorithm starts with a logical topology with no links (lightpaths) and sequentially adds lightpaths as long as doing so does not violate any of the problem constraints. The reverse approach is also possible [34]: starting with a fully connected logical topology, an algorithm sequentially removes the lightpath carrying the smallest traffic flows until no constraint is violated. At each step (i.e., after removing a lightpath), the traffic routing subproblem is solved in order to find the lightpath with the smallest flow.

The third approach to RWA is to start with a given logical topology, thus avoiding to directly solve the first of the four subproblems listed above. Regular topologies are good candidates as logical topologies since they are well understood and results regarding bounds and averages (e.g., for hop lengths) are easier to derive. Algorithms for routing traffic on a regular topology are usually simple, so the traffic routing subproblem can be trivially solved. Also, regular topologies possess inherent load balancing characteristics which are important when the objective is to minimize the maximum congestion.

Once a regular topology is decided on as the one to implement the logical topology, it remains to decide which physical node will realize each given node in the regular topology (this is usually referred to as the *node mapping* subproblem), and which sequence of physical links will be used to realize each given edge (lightpath) in the regular topology (this *path mapping* subproblem is equivalent to the lightpath routing and wavelength assignment subproblems discussed earlier). This procedure is usually referred to embedding a regular topology in the physical topology. Both the node and path mapping subproblems are intractable, and heuristics have been proposed in the literature [34,35]. For instance, a heuristic for mapping the nodes of shuffle topologies based on the gradient algorithm was developed in [35].

Given that all the algorithms for the RWA problem are based on heuristics, it is important to be able to characterize the quality of the solutions obtained. To this end, one must resort to comparing the solutions to known bounds on the optimal solution. A comprehensive discussion of bounds for the RWA problem and the theoretical considerations involved in deriving them can be found in [9]. A simulation-based comparison of the relative performance of the three classes of heuristic for the RWA problem is presented in [10]. The results indicate that the second class of algorithms discussed earlier achieve the best performance.

The study in [22] also focused on virtual topology design (i.e., static RWA) for point-to-point traffic but observed that, since a light-tree is a more general representation of a lightpath, the set of virtual topologies that can be implemented using light-trees is a superset of the virtual topologies that can be implemented only using lightpaths. Thus, for any given virtual topology problem, an optimal solution using light-trees is guaranteed to be at least as good and possibly an improvement over the optimal solution obtained using only lightpaths. Furthermore, it was demonstrated that by extending the lightpath concept to a light-tree, the network performance (in terms of average packet hops) can be improved while the network cost (in terms of the number of optical transmitters/receivers required) decreases.

The static MC-RWA problem has been studied in [36,37]. The study in [36] focused on demonstrating the benefits of multicasting in wavelength routed optical networks. Specifically, it was shown that using light-trees (spanning the source and destination nodes) rather than individual parallel lightpaths (each connecting the source to an individual destination) requires fewer wavelengths and consumes a significantly lower amount of bandwidth. In [37] an ILP formulation that maximizes the total number of multicast connections was presented for the static MC-RWA problem. Rather than providing heuristic algorithms for solving the ILP, bounds on the objective function were presented by relaxing the integer constraints.

3.2 Dynamic RWA

During real-time network operation, edge nodes submit to the network requests for lightpaths to be set up as needed. Thus, connection requests are initiated in some random fashion. Depending on the state of the network at the time of a request, the available resources may or may not be sufficient to establish a lightpath between the corresponding source-destination edge node pair. The network state consists of the physical path (route) and wavelength assignment for all active lightpaths. The state evolves randomly in time as new lightpaths are admitted and existing lightpaths are released. Thus, each time a request is made, an algorithm must be executed in real time to determine whether it is feasible to accommodate the request, and, if so, to perform routing and wavelength assignment. If a request for a lightpath cannot be accepted because of lack of resources, it is blocked.

Because of the real-time nature of the problem, RWA algorithms in a dynamic traffic environment must be very simple. Since combined routing and

wavelength assignment is a hard problem, a typical approach to designing efficient algorithms is to decouple the problem into two separate subproblems: the routing problem and the wavelength assignment problem. Consequently, most dynamic RWA algorithms for wavelength routed networks consist of the following general steps:

1. Compute a number of candidate physical paths for each source-destination edge node pair and arrange them in a path list.
2. Order all wavelengths in a wavelength list.
3. Starting with the path and wavelength at the top of the corresponding list, search for a feasible path and wavelength for the requested lightpath.

The specific nature of a dynamic RWA algorithm is determined by the number of candidate paths and how they are computed, the order in which paths and wavelengths are listed, and the order in which the path and wavelength lists are accessed.

Route Computation. Let us first discuss the routing subproblem. If a *static* algorithm is used, the paths are computed and ordered independently of the network state. With an *adaptive* algorithm, on the other hand, the paths computed and their order may vary according to the current state of the network. A static algorithm is executed off-line and the computed paths are stored for later use, resulting in low latency during lightpath establishment. Adaptive algorithms are executed at the time a lightpath request arrives and require network nodes to exchange information regarding the network state. Lightpath set up delay may also increase, but in general, adaptive algorithms improve network performance.

The number of path choices for establishing an optical connection is another important parameter. A *fixed* routing algorithm is a static algorithm in which every source-destination edge node pair is assigned a single path. With this scheme, a connection is blocked if there is no wavelength available on the designated path at the time of the request. In *fixed-alternate* routing, a number $k, k > 1$, of paths are computed and ordered off-line for each source-destination edge node pair. When a request arrives, these paths are examined in the specified order and the first one with a free wavelength is used to establish the lightpath. The request is blocked if no wavelength is available in any of the k paths. Similarly, an adaptive routing algorithm may compute a single path, or a number of alternate paths at the time of the request. A hybrid approach is to compute k paths off-line, however, the order in which the paths are considered is determined according to the network state at the time the connection request is made (e.g., least to most congested).

In most practical cases, the candidate paths for a request are considered in increasing order of *path length* (or *path cost*). Path length is typically defined as the sum of the weights (costs) assigned to each physical link along the path, and the weights are chosen according to some desirable routing criterion. Since weights can be assigned arbitrarily, they offer a wide range of possibilities for selecting path priorities. For example, in a static (fixed-alternate) routing algorithm, the weight of each link could be set to 1, or to the physical distance of

the link. In the former case, the path list consists of the k minimum-hop paths, while in the latter the candidate paths are the k minimum-distance paths (where distance is defined as the geographic length). In an adaptive routing algorithm, link weights may reflect the load or "interference" on a link (i.e., the number of active lightpaths sharing the link). By assigning small weights to least loaded links, paths with larger number of free channels on their links rise to the head of the path list, resulting in a *least loaded* routing algorithm. Paths that are congested become "longer" and are moved further down the list; this tends to avoid heavily loaded bottleneck links. Many other weighting functions are possible.

When path lengths are sums of of link weights, the k-shortest path algorithm [38] can be used to compute candidate paths. Each path is checked in order of increasing length, and the first that is feasible is assigned the first free wavelength in the wavelength list. However, the k shortest paths constructed by this algorithm usually share links. Therefore, if one path in the list is not feasible, it is likely that other paths in the list with which it shares a link will also be infeasible. To reduce the risk of blocking, the k shortest paths can be computed so as to be pairwise link-disjoint. This can be accomplished as follows: when computing the i-th shortest path, $i = 1, \cdots, k$, the links used by the first $i-1$ paths are removed from the original network topology and Dijkstra's shortest path algorithm [39] is applied to the resulting topology. This approach increases the chances of finding a feasible path for a connection request.

The problem of determining algorithms for routing multicast optical connections has also been studied in [37,40]. The problem of constructing trees for routing multicast connections was considered in [40] independently of wavelength assignment, under the assumption that not all OXCs are multicast capable, i.e., that there is a limited number of MC-OXCs in the network. Four new algorithms were developed for routing multicast connections under this *sparse light splitting* scenario. While the algorithms differ slightly from each other, the main idea to accommodate sparse splitting is to start with the assumption that all OXCs in the network are multicast capable and use an existing algorithm to build an initial tree. Such a tree is infeasible if a non-multicast-capable OXC is a branching point. In this case, all but one branches out of this OXC are removed, and destination nodes in the removed branches have to join the tree at a MC-OXC. In [37], on the other hand, the MC-RWA problem was solved by decoupling the routing and wavelength assignment problems. A number of *alternate* trees are constructed for each multicast connection using existing routing algorithms. When a request for a connection arrives, the associated trees are considered in a fixed order. For each tree, wavelengths are also considered in a fixed order (i.e., the first-fit strategy discussed in the next subsection). The connection is blocked if no free wavelength is found in any of the trees associated with the multicast connection.

We note that most of the literature (and the preceding discussion) has focused on the problem of obtaining paths that are optimal with respect to total path cost. In transparent optical networks, however, optical signals may suffer from physical layer impairments including attenuation, chromatic dispersion,

polarization mode dispersion (PMD), amplifier spontaneous emission (ASE), cross-talk, and various nonlinearities [41]. These impairments must be taken into account when choosing a physical path. In general, the effect of physical layer impairments may be translated into a set of constraints that the physical path must satisfy; for instance, the total signal attenuation along the physical path must be within a certain power budget to guarantee a minimum level of signal quality at the receiver. Therefore, a simple shortest path first (SPF) algorithm (e.g., Dijkstra's algorithm implemented by protocols such as OSPF [42]) may not be appropriate for computing physical paths within a transparent optical network. Rather, constraint-based routing techniques such as the one employed by the constraint-based shortest path first (CSPF) algorithm [5] are needed. These techniques compute paths by taking into account not only the link cost but also a set of constraints that the path must satisfy. A first step towards the design of constraint-based routing algorithms for optical networks has been taken in [41] where it was shown how to translate the PMD and ASE impairments into a set of linear constraints on the end-to-end physical path. However, additional work is required to advance our understanding of how routing is affected by physical layer considerations, and constraint-based routing remains an open research area [43].

Wavelength Assignment. Let us now discuss the wavelength assignment subproblem which is concerned with the manner in which the wavelength list is ordered. For a given candidate path, wavelengths are considered in the order in which they appear in the list to find a free wavelength for the connection request. Again, we distinguish between the static and adaptive cases. In the static case, the wavelength ordering is fixed (e.g., the list is ordered by wavelength number). The idea behind this scheme, also referred to as *first-fit*, is to pack all the in-use wavelengths towards the top of the list so that wavelengths towards the end of the list will have higher probability of being available over long continuous paths. In the adaptive case, the ordering of wavelengths is typically based on usage. Usage can be defined either as the number of links in the network in which a wavelength is currently used, or as the number of active connections using a wavelength. Under the *most used* method, the most used wavelengths are considered first (i.e., wavelength are considered in order of decreasing usage). The rationale behind this method is to reuse active wavelengths as much as possible before trying others, packing connections into fewer wavelengths and conserving the spare capacity of less-used wavelengths. This in turn makes it more likely to find wavelengths that satisfy the continuity requirement over long paths. Under the *least used* method, wavelengths are tried in the order of increasing usage. This scheme attempts to balance the load as equally as possible among all the available wavelengths. However, least used assignment tends to "fragment" the availability of wavelengths, making it less likely that the same wavelength is available throughout the network for connections that traverse longer paths.

The most used and least used schemes introduce communication overhead because they require global network information in order to compute the usage

of each wavelength. The first-fit scheme, on the other hand, requires no global information, and since it does not need to order wavelengths in real-time, it has significantly lower computational requirements than either the most used or least used schemes. Another adaptive scheme that avoids the communication and computational overhead of most used and least used is *random* wavelength assignment. With this scheme, the set of wavelengths that are free on a particular path is first determined. Among the available wavelengths, one is chosen randomly (usually with uniform probability) and assigned to the requested lightpath.

We note that in networks in which all OXCs are capable of wavelength conversion, the wavelength assignment problem is trivial: since a lightpath can be established as long as at least one wavelength is free at each link and different wavelengths can be used in different links, the order in which wavelengths are assigned is not important. On the other hand, when only a fraction of the OXCs employ converters (i.e., a sparse conversion scenario), a wavelength assignment scheme is again required to select a wavelength for each segment of a connection's path that originates and terminates at an OXC with converters. In this case, the same assignment policies discussed above for selecting a wavelength for the end-to-end path can also be used to select a wavelength for each path segment between OXCs with converters.

Performance of Dynamic RWA Algorithms. The performance of a dynamic RWA algorithm is generally measured in terms of the call blocking probability, that is, the probability that a lightpath cannot be established in the network due to lack of resources (e.g., link capacity or free wavelengths). Even in the case of simple network topologies (such as rings) or simple routing rules (such as fixed routing), the calculation of blocking probabilities in WDM networks is extremely difficult. In networks with arbitrary mesh topologies, and/or when using alternate or adaptive routing algorithms, the problem is even more complex. These complications arise from both the link load dependencies (due to interfering lightpaths) and the dependencies among the sets of active wavelengths in adjacent links (due to the wavelength continuity constraint). Nevertheless, the problem of computing blocking probabilities in wavelength routed networks has been extensively studied in the literature, and approximate analytical techniques which capture the effects of link load and wavelength dependencies have been developed in [11,19,16]. A detailed comparison of the performance of various wavelength assignment schemes in terms of call blocking probability can be found in [44].

Though important, average blocking probability (computed over all connection requests) does not always capture the full effect of a particular dynamic RWA algorithm on other aspects of network behavior, in particular, *fairness*. In this context, fairness refers to the variability in blocking probability experienced by lightpath requests between the various edge node pairs, such that lower variability is associated with a higher degree of fairness. In general, any network has the property that longer paths are likely to experience higher blocking than

shorter ones. Consequently, the degree of fairness can be quantified by defining the *unfairness factor* as the ratio of the blocking probability on the longest path to that on the shortest path for a given RWA algorithm. Depending on the network topology and the RWA algorithm, this property may have a cascading effect which can result in an unfair treatment of the connections between more distant edge node pairs: blocking of long lightpaths leaves more resources available for short lightpaths, so that the connections established in the network tend to be short ones. These shorter connections "fragment" the availability of wavelengths, and thus, the problem of unfairness is more pronounced in networks without converters, since finding long paths that satisfy the wavelength continuity constraint is more difficult than without this constraint.

Several studies [11,19,16] have examined the influence of various parameters on blocking probability and fairness, and some of the general conclusions include the following:

- Wavelength conversion significantly affects fairness. Networks employing converters at all OXCs sometimes exhibit orders of magnitude improvement in fairness (as reflected by the unfairness factor) compared to networks with no conversion capability, despite the fact that the improvement in overall blocking probability is significantly less pronounced. It has also been shown that equipping a relatively small fraction (typically, 20-30%) of all OXCs with converters is sufficient to achieve most of the fairness benefits due to wavelength conversion.
- Alternate routing can significantly improve the network performance in terms of both overall blocking probability and fairness. In fact, having as few as three alternate paths for each connection may in some cases (depending on the network topology) achieve almost all the benefits (in terms of blocking and fairness) of having full wavelength conversion at each OXC with fixed routing.
- Wavelength assignment policies also play an important role, especially in terms of fairness. The random and least used schemes tend to "fragment" the wavelength availability, resulting in large unfairness factors (with least used having the worst performance). On the other hand, the most used assignment policy achieves the best performance in terms of fairness. The first-fit scheme exhibits a behavior very similar to most used in terms of both fairness and overall blocking probability, and has the additional advantage of being easier and less expensive to implement.

4 Control Plane Issues and Standardization Activities

So far we have focused on the application of network design and traffic engineering principles to the control of traffic in optical networks with a view to achieving specific performance objectives, including efficient utilization of network resources and planning of network capacity. Equally important to an operational network are associated control plane issues involved in automating the process of lightpath establishment and in supporting the network design and

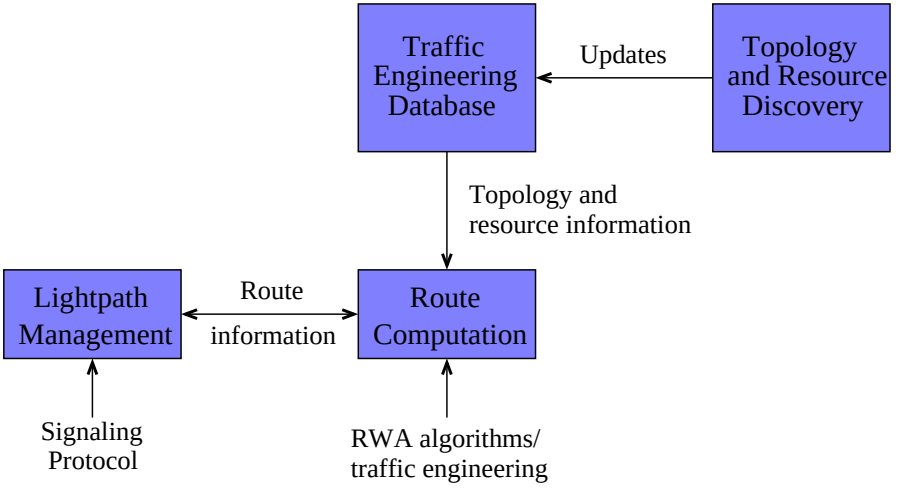

Fig. 5. Control plane components

traffic engineering functions. Currently, a number of standardization activities addressing the control plane aspects of optical networks are underway [45,46, 47] within the Internet Engineering Task Force (IETF) [48], the Optical Domain Service Interconnection (ODSI) coalition [49], and the Optical Internetworking Forum (OIF) [50]. In this section we review the relevant standards activities and discuss how they fit within the traffic engineering framework; we note, however, that these are ongoing efforts and will likely evolve as the underlying technology matures and our collective understanding of optical networks advances.

Let us return to Figure 1 which illustrates the manner in which client subnetworks (IP/MPLS networks in the figure) attach to the optical network of OXCs. The figure corresponds to the vision of a future optical network which is capable of providing a bandwidth-on-demand service by dynamically creating and tearing down lightpaths between client subnetworks. There are two broad issues that need to be addressed before such a vision is realized. First, a signaling mechanism is required at the user-network interface (UNI) between the client subnetworks and the optical network control plane. The signaling channel allows edge nodes to dynamically request bandwidth from the optical network, and supports important functions including service discovery and provisioning capabilities, neighbor discovery and reachability information, address registration, etc. Both the ODSI coalition [51] and the OIF [52] have developed specifications for the UNI; the OIF specifications are based on GMPLS [6].

Second, a set of signaling and control protocols must be defined within the optical network to support dynamic lightpath establishment and traffic engineering functionality; these protocols are implemented at the control module of each OXC. Currently, most of the work on defining control plane protocols in the optical network takes place under the auspices of IETF, reflecting a convergence

of the optical networking and the IP communities to developing technology built around a single common framework, namely, GMPLS, for controlling both IP and optical network elements [53]. There are three components of the control plane that are crucial to setting up lightpaths within the optical network (refer to Figure 5):

- **Topology and resource discovery.** The main purpose of discovery mechanisms is to disseminate network state information including resource usage, network connectivity, link capacity availability, and special constraints.
- **Route Computation.** This component employs RWA algorithms and traffic engineering functions to select an appropriate route for a requested lightpath.
- **Lightpath Management.** Lightpath management is concerned with setup and tear-down of lightpaths, as well as coordination of protection switching in case of failures.

Topology and resource discovery includes neighbor discovery, link monitoring, and state distribution. The link management protocol (LMP) [54] has been proposed to perform neighbor discovery and link monitoring. LMP is expected to run between neighboring OXC nodes and can be used to establish and maintain control channel connectivity, monitor and verify data link connectivity, and isolate link, fiber, or channel failures. Distribution of state information is typically carried out through link state routing protocols such as OSPF [42]. There are currently several efforts under way to extend OSPF to support GMPLS [55] and traffic engineering [56]. In particular, the link state information that these protocols carry must be augmented to include optical resource information including: wavelength availability and bandwidth, physical layer constraints (discussed in Section 3.2), and link protection information, among others. This information is then used to build and update the optical network traffic engineering database (see Figure 5) which guides the route selection algorithm.

Once a lightpath is selected, a signaling protocol must be invoked to set up and manage the connection. Two protocols have currently been defined to signal a lightpath setup: RSVP-TE [57] and CR-LDP [58]. RSVP-TE is based on the resource reservation protocol (RSVP) [59] with appropriate extensions to support traffic engineering, while CR-LDP is an extension of the label distribution protocol (LDP) [60] augmented to handle constraint-based routing. The protocols are currently being extended to support GMPLS [61,62]. Besides signaling the path at connection time, both protocols can be used to automatically handle the switchover to the protection path once a failure in the working path has occurred. In the next section, we describe in detail the operation of some of these control plane protocols.

We note that the control plane elements depicted in Figure 5 are independent of each other and, thus, separable. This modularity allows each component to evolve independently of others, or to be replaced with a new and improved protocol. As the optical networking and IP communities come together to define standards, the constraints and new realities (e.g., the explosion in the number of channels in the network) imposed by the optical layer and WDM technology

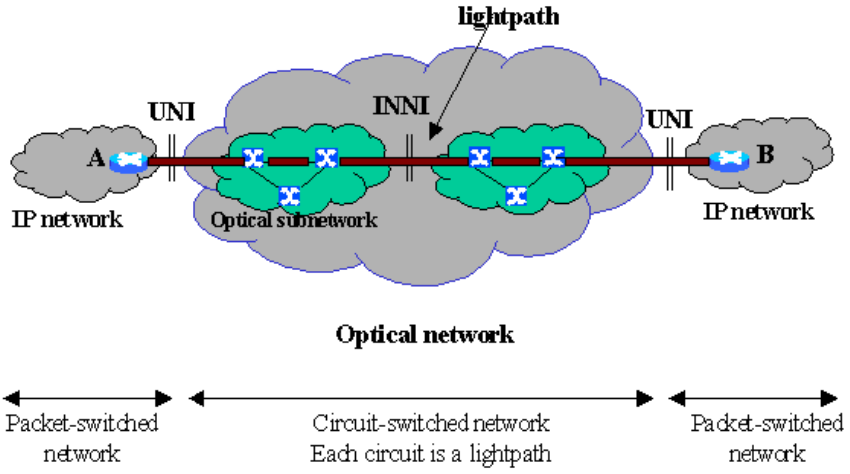

Fig. 6. IP networks interconnected by an optical network

will certainly affect our long-held assumptions regarding issues such as routing, control, discovery, etc., which have been developed for mostly opaque electronic networks. As we carefully rethink these issues in the context of transparent (or almost-transparent) optical networks, protocol design will certainly evolve to better accommodate the new technology. Therefore, we expect that the control plane protocols will continue to be refined and/or replaced by new, more appropriate ones. The interested reader should frequently check with the activities within IETF and OIF for the most recent developments.

We now proceed to describe some of the proposed control plane protocols. We first describe a proposed framework for transporting IP traffic over optical networks, and subsequently we present MPLS, LDP, CR-LDP, and GMPLS.

5 IP over Optical – A Framework

The issue of how IP traffic will be transported over an optical network has been addressed by IETF's IP over optical (IPO) working group in the [63]. The main features of this internet draft are summarized in this section.

An optical network is assumed to consist of interconnected optical sub-networks, where an optical sub-network consists of OXCs built by the same vendor. An optical network is a single administrative network. Optical networks can be combined to form an optical internetwork. An optical network is used as a backbone to interconnect a number of different IP networks, as well as other packet networks such as ATM and frame relay. In Figure 6, we show two IP networks interconnected via an optical network. We note that the edge IP router A is connected to the edge IP router B via a lightpath.

Three different methods have been proposed for the control plane, namely, the *peer* model, the *overlay* model, and the *augmented* model. In the peer model, the IP and optical networks are treated together as a single network. Each OXC is equipped with an IP address, and all IP routers and OXCs use a single control plane based on GMPLS. In view of this, there are no special user-network interface (UNI) or network-node interface (NNI). The IP and the optical networks run the same IP routing protocol, such as OSPF with suitable "optical" extensions, and the topological and link state information maintained by all IP and OXC nodes is identical. LSPs can be established using CR-LDP or RSVP-TE extended.

The overlay model is closer to the classical IP and ARP over ATM scheme which is used to transport IP traffic over ATM [64]. The optical network and the IP networks are independent of each other, and an edge IP router interacts with its ingress OXC over a well-defined UNI. The optical network is responsible for setting up a lightpath between two edge IP routers. A lightpath may be either switched or permanent. Switched lightpaths are established in real-time using signaling procedures, and they may last for a short or a long period of time. Permanent lightpaths are setup administratively by subscription and typically they last for a very long time. An edge IP router requests a switched lightpath from its ingress OXC using a signaling protocol over the UNI. Signaling messages are provided for creating, deleting, and modifying a switched lightpath. Routing within the optical network is independent of the routing within the IP networks.

Finally, in the augmented model the IP and optical networks use separate routing protocols, but information from one routing protocol is passed through the other routing protocol. For instance, external IP addresses could be carried within the optical routing protocol to allow reachability information to be passed to IP clients. The inter-domain IP routing protocol BGP may be used for exchanging information between IP and optical domains. Addressing of an OXC is identified by a unique IP address and a selector. The selector identifies further fine-grained information of relevance at the OXC, such as port, channel, sub-channel, etc. Typically, the setting up of a lightpath will be done in a distributed fashion similar to setting up a connection in ATM networks and also in MPLS-ready IP networks. Recently, it has been proposed to use a centralized scheme for setting up lightpaths. This requires a centralized server which has complete knowledge of the physical topology and wavelength availability [65,66]. The Common Open Policy Service (COPS) signaling protocol is used by the ingress switch of an edge router to request the establishment of a connection from the Policy Decision Point (PDP), a remote server, which calculates the path and downloads the information to all the nodes along the path.

5.1 Multiprotocol Label Switching (MPLS)

In order to understand the signalling protocols that have been proposed to control a wavelength-routed optical network, we first need to examine the Multiprotocol Label Switching (MPLS) scheme.

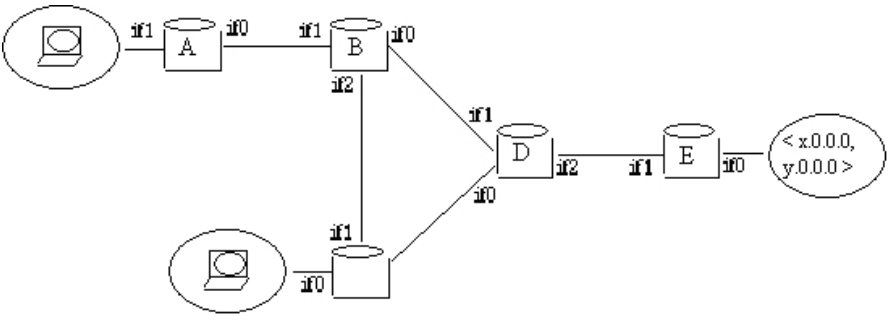

Fig. 7. The shim label header

MPLS was developed as a means of introducing connection oriented features in an IP network. A router forwards an IP packet according to its prefix. In a given router, the set of all addresses that have the same prefix, is referred to as the forwarding equivalent class (FEC). IP packets belonging to the same FEC have the same output interface. In MPLS, a FEC is associated with a label. This label is used to determine the output interface of an IP packet without having to do the traditional look-up its address in the routing table. In IPv6, the label can be carried in the flow label field. In IPv4, however, there is no space for such a label in the IP header. If the IP network runs on top of an ATM network, the label is carried in the VPI/VCI field of an ATM cell. If it is running over frame relay, the label is carried in the DLCI field. For Ethernet, token ring, and point-to-point connections running a link layer protocol such as PPP, the label is carried in a special shim label header, which is inserted between the LLC header and the IP header, as shown in Figure 7. The first field of the shim label header is a 20-bit field used to carry the label. The second field is a 3-bit field used for the class-of-service (CoS) indication. This field is used to indicate the priority of the IP packet. The S field is used in conjunction with the label stack. Finally, the time-to-live (TTL) field is similar to the TTL field in the IP header. A label switching network consists of label switching routers (LSR), which are IP routers that run MPLS, they forward IP packets based on their labels, and they can also carry the customary IP forwarding decision based on the prefix of an IP addresses. An MPLS node is an LSR which may not necessarily forward IP packets based on the prefixes.

To see how label switching works, let us consider a network consisting of 5 LSRs, A, B, C, D, and E, linked with point-to-point connections as shown in Figure 8. We assume that a new set of hosts with the prefix <x.0.0.0,y.0.0.0>, where x.0.0.0 is the base network address and y.0.0.0 is the mask, is directly connected to E. The flow of IP packets with this prefix from A to E is via B and D. That is, A's next-hop router for this prefix is B, B's next-hop router is D, and D's next-hop router is E. Likewise, the flow of IP packets with the same prefix from C to E is via D. That is, C's next-hop router for this prefix is D, and D's next-hop router is E. The interfaces in Figure 8 show how these routers

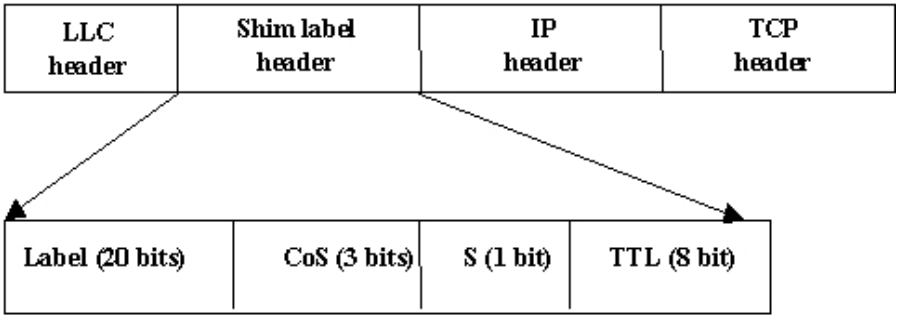

Fig. 8. An example of label switching

are interconnected. For instance, A is connected to B via if0, B is connected to A via if1, to C via if2 and to D via if0, and so on. When an LSR identifies the FEC associated with this new prefix <x.0.0.0,y.0.0.0>, it selects a label from a pool of free labels and it makes an entry into a table known as the *label forward information base* (LFIB). This table contains information regarding the incoming and outgoing labels associated with a FEC, the output interface, i.e., the FEC's next-hop router, and the operation that needs to be performed on the label stack. Incoming IP packets belonging to this particular FEC have to be labeled with the value selected by the LSR. In view of this, the LSR has to notify its neighbours about its label selection for the particular FEC. In the above example, LSR B sends its information to A, D, and C. A recognizes that it is upstream from B, and it uses the information to update the entry for this FEC in its LFIB. D sends its information to B, C, and E. Since B and C are both upstream of D, they use this information to update the entries in their LFIB. E sends its information to D, which uses it to update its entry in its LFIB. As a result, in each LSR each incoming label associated with a FEC is bound to an outgoing label in the LFIB entry. In Figure 9, we show the labels allocated by the LSRs. The sequence of labels 62, 15, 60 forms a path, referred to as the label switched path (LSP). Typically, there may be several label switched paths associated with the same FEC which form a tree, as shown in Figure 9.

Once the labels have been distributed and the entries have been updated in the LFIBs, the forwarding of an IP packet belonging to the FEC associated with the prefix <x.0.0.0, y.0.0.0> is done using solely the labels. Let us assume that A receives an IP packet from one of its local hosts with a prefix <x.0.0.0, y.0.0.0>. A identifies that the packet's IP address belongs to the FEC, and it looks up its LFIB to obtain the label value and the outgoing interface. It creates a shim label header, sets the label value to 62, and forwards it to the outgoing interface if0. When the IP packet arrives at LSR B, its label is extracted and looked up in B's LFIB. The old label is replaced by the new one, which is 15, and the IP packet is forwarded to interface if0. LSR D follows exactly the same procedure. When it receives the IP packet from B, it replaces its incoming label with the outgoing

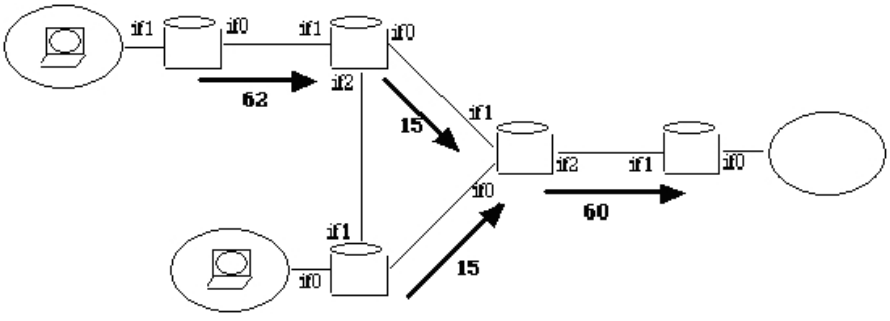

Fig. 9. Label switched paths

label, which is 60, and forwards it to interface if2. Finally, E forwards the IP packet to its local destination. The same procedure applies for an IP packet with a prefix <x.0.0.0, y.0.0.0> that arrives at C. Labeled IP packets within an LSR are served according to their priority, carried in the CoS field of he shim header. Specifically, an IP router maintains different quality-of-service queues for each output interface. These queues are served using a scheduling algorithm, so that different classes of IP packets can be served according to their requested quality of service. Another interesting feature of label switching is that it can be used to create a dedicated route, known as an explicit route, between two IP routers. Explicit routing is used primarily in optical networks, and it is described below.

Label allocation. In the example described above, a label is generated by the LSR which is at the downstream end of the link, with respect to the flow of the IP packets. In view of this, this type of label allocation is known as *downstream label allocation*. In addition to this scheme, labels can be allocated using *downstream label allocation on demand*. In this case, each LSR allocates an incoming label to a FEC and creates an appropriate entry in its LFIB. However, it does not advertise its label to its neighbours as in the case of downstream allocation. Instead, an upstream LSR obtains the label information by issuing a request.

Explicit routing. As we have discussed above, a router makes a forwarding decision by using the IP address in its routing table in order to determine the next-hop router. Typically, each IP router calculates the next-hop router for a particular destination using the shortest path algorithm. Label switching follows the same general approach, only it uses labels. This routing scheme is known as hop-by-hop routing. An alternative way of routing a packet is to use source routing. In this case, the originating (source) LSR selects the path to the destination LSR. Other LSRs on the path simply obey the source's routing instructions. Source routing can be used in an IP network for a variety of reasons, such as to evenly distribute traffic among links by moving some of the traffic from highly utilized links to less utilized links (load balancing), create tunnels for MPLS-based VPNs, and introduce routes based on a quality-of-service criterion such as minimize the number of hops, minimize the total end-to-end delay, and max-

imize throughput. Label switching can be used to set-up such routes, referred to as CR-LSP. In optical networks, only explicit routing is used.

Set-up of an LSP. The setup of an LSP can be done in one of the two following ways: *independent LSP control* and *ordered LSP control*. In independent control, when an LSR recognizes a new FEC, it binds a label to it and advertises it to its neighbors. In ordered control, the allocation of labels proceeds backwards starting from the egress LSP LSR. That is, an LSR only binds a label to a FEC if it is the egress LSR for that FEC or it has already received a label binding for that FEC from its next hop LSR for that FEC.

Label distribution protocol. A label distribution protocol is required to reliably establish and maintain label bindings. As mentioned previously, the RSVP protocol and its extension RSVP-TE have been proposed to be used as label distribution protocol. In addition, we new protocol, the label distribution protocol (LDP), has been proposed for MPLS. LDP has been extended to CR-LDP for the establishment, maintenance, and tearing down of explicit routes.

5.2 The Label Distribution Protocol (LDP)

For reliability purposes, the LDP protocol runs over TCP [60]. Two LSRs that run LDP and they are directly connected are known as LDP peers.

An LSR discovers potential LDP peers by sending periodically LDP link hello messages out of each interface. The receipt of an LDP hello message triggers the establishment of an LDP session between two LDP peers. When the LDP session is initialized, the two LDP peers negotiate session parameters such as label distribution method, timer values, range of VPI/VCI values for ATM, and range of DLCI values for frame relay. An LDP session is soft-state and it needs to be continuously refreshed. A session is maintained as long as traffic flows (in the form of LDP PDUs) over the session. In the absence of LDP PDUs, keepAlive messages are sent. LDP supports independent label distribution control and ordered label distribution control. It also provides functionality for detection of loops in the LSP. Information in LDP is sent in the form of LDP PDUs, which consists of a header followed by one or more LDP messages. An LDP message consists of a header followed by mandatory and optional parameters. The header and the parameters are all encoded using the type-length-value (TLV) scheme. The type specifies how the value field is to be interpreted, the length gives the length of the value, and the value field contains the actual information. The value field contains one or more TLVs.

The following LDP messages have been defined: notification, hello, initialization, keepAlive, address, address withdraw, label mapping, label request, label abort, label withdraw, and label release. The notification message is used to inform an LDP peer of a fatal error or to provide advisory information regarding the outcome of processing an LDP message. The hello messages are used to discover peer LDPs, and the initialization message is used to initialize a new session between two peer LDPs. The address message is used by an LSR to advertise the address of its interfaces. Previously advertised addresses can be withdrawn using the address withdraw message. The label mapping message is used by an LSR

to advertise a binding of a label to a FEC to its LDP peers. The label request message is used by an LSR to request a label from a peer LDP to a FEC. An LSR may transmit a request message under the following conditions:

- The LSR recognizes a new FEC via the forwarding table, and the next hop is an LDP peer, and the LSR does not already have a mapping from the next hop for the given FEC.
- The next hop to the FEC changes, and the LSR does not already have a mapping from the next hop for the given FEC.
- The LSR receives a label request for a FEC from an upstream LDP peer, the FEC next hop is an LDP peer, and the LSR does not already have a mapping from the next hop.

5.3 Constrained Routing Label Distribution Protocol (CR-LDP)

CR-LDP is a signaling protocol based on LDP, and it runs over TCP. It is used to set-up a point-to-point LSP, referred to as CR-LSP. A CR-LSP, unlike an LSP, is a point-to-point path through an MPLS network, which is set-up based on criteria not limited to routing information, such as explicit routing and QoS based routing. A CR-LDP may be used for a variety of reasons, such as, to evenly distribute traffic among links (load balancing), create tunnels for MPLS-based VPNs, and introduce routes based on a QoS criteria, such as minimization of the number of hops, minimization of the total end-to-end delay, and maximization of throughput.

A CR-LSP is setup as follows. A request at an ingress LSR to setup a CR-LSP originates from a management system or an application. The ingress LSR calculates the explicit route using information provided by the management system, or the application, or form a routing table. The explicit route is a series of nodes or groups of nodes (referred to as abstract nodes), which is signalled to nodes or abstract nodes along the path using the label request message. CR-LSPs are set up using ordered control with downstream on demand label allocation. Strict and loose explicit routes can be used. In a strict route all the LSRs through which the CR-LSP must pass are indicated. In loose routing, some LSRs are specified, and the exact path between two such LSRs is determined using conventional routing based on IP addresses. Route pinning is a feature that can be used to fix the path through a loosely defined route, so that it does not change when a better next hop becomes available.

As in ATM networks, CR-LDP permits the specification of traffic parameters for a CR-LSP. The following five traffic parameters have been specified: peak data rate (PDR), peak burst size (PBS), committed data rate (CDR), committed bucket size (CBS), and excessive bucket size (EBS). PBS and PDR are used to specify the peak rate. This is the maximum rate at which traffic is sent to the CR-LSP, and it is expressed in bytes/sec. It is defined in terms of a token bucket whose maximum bucket size is set equal to the peak burst size (PBS) and the rate at which it is replenished is equal to the peak data rate (PDR). CBS and CDR are used to specify the committed rate, which is the amount of

bandwidth allocated to a CR-LSP by an LSR. It is defined by a token bucket whose maximum bucket size is set equal to CBS and the rate at which the bucket is replenished is equal to CDR. Finally, the excess bucket size (EBS) is used to define the maximum size of a third token bucket, the excess token bucket, which is replenished at the rate of CDR. By appropriately manipulating the values of these five traffic parameters, it is possible to establish different classes of service.

The establishment of an CR-LSP is achieved using the label request message and label mapping message. The label request message carries the list of all nodes and abstract nodes which are on the path of the CR-LSP, the traffic parameters, FEC, and other relevant parameters. It is propagated from the ingress LSR to the egress LSR. The label mapping message is used to advertise the labels, which is done using ordered control, that is from the egress LSR back towards the ingress LSR.

5.4 Generalized MPLS (GMPLS)

GMPLS [6] extends the label switching architecture proposed in MPLS to other types of non-packet based networks, such as SONET/SDH based networks and wavelength-routed networks. Specifically, the GMPLS architecture supports the following types of switching: packet switching (IP, ATM, and frame relay), wavelength switching in a wavelength-routed network, port or fiber switching in a wavelength-routed network, and time slot switching for a SONET/SDH cross-connect.

A GMPLS LSR may support the following five interfaces: packet switch interfaces, layer-2 switch interfaces, time-division multiplex interfaces, lambda switch interfaces, and fiber switch interfaces. A packet switch interface recognizes packet boundaries and it can forward packets based on the content of the IP header or the content of the shim header. A layer-2 switch interface recognizes frame/cell boundaries and can forward data based on the content of the frame/cell header. Examples include interfaces on ATM-LSRs that forward cells based on their VPI/VCI value, and interfaces on Ethernet bridges that forward data based on the MAC header. A time-division multiplex interface forwards data based on the data's time slot in a repeating cycle (frame). Examples of this interface is that of a SONET/SDH cross-connect, terminal multiplexer, and add-drop multiplexer. Other examples include interfaces implementing the digital wrapper (G.709) and PDH interfaces. A lamda-switch interface forwards the optical signal from an incoming wavelength to an outgoing wavelength. An example of such an interface is the optical cross-connect (OXC) that operates at the level of an individual wavelength or a group of wavelengths (waveband). Finally, a fiber switch interface forwards the signals from one (or more) incoming fibers to one (or more) outgoing fibers. An example of this interface is an OXC that operates at the level of a fiber or group of fibers.

GMPLS extends the control plane of MPLS to support each of the five classes of interfaces. The GMPLS supports the peer model, the overlay model and the augmented model. In GMPLS, downstream on-demand label allocation is used with ordered control initiated by an ingress node. There is no restriction on the

route selection. Explicit routing is normally used, but hop-by-hop routing can be also used. There is also no restriction on the way an LSP is set-up. It could be set-up as described in the example above in the MPLS section (control driven), or it could be set-up as a result of a user issuing a request to establish an LSP. The latter approach is suitable for circuit-switching technologies. Several new forms of labels are required to deal with the widened scope of MPLS into the optical and time division multiplexing domain. The new label not only allows for the familiar label that travels in-band with the associated packet, but it also allows for labels which identify time-slots, wavelengths, or fibers. The generalized label may carry a label that represents a single fiber in a bundle, a single wavelength within a fiber, a single wavelength within a waveband or a fiber, a set of time-slots within a the SONET/SDH payload carried over a wavelength, and the MPLS labels for IP packets, ATM cells and frame relay frames. This new label is known as the generalized label.

CR-LDP [62] and RSVP-TE [61] have both been extended to allow the signalling and instantiation of lightpaths. A UNI signalling protocol has been proposed by OIF based on GMPLS. The interior gateway protocols IS-IS and OSPF have been extended to advertise availability of optical resources (i.e., bandwidth on wavelengths, interface types) and other network attributes and constraints. Also, a new link management protocol (LMP) has been developed to address issues related to the link management in optical networks.

6 Optical Packet Switching

Optical packet switching has been proposed as a solution to transporting packets over an optical network. Optical packet switching is sometimes referred to as "optical ATM," since it resembles ATM, but it takes place in the optical domain.

A WDM optical packet switch consists of four parts, namely, the input interface, the switching fabric, the output interface, and the control unit. The input interface is mainly used for packet delineation and alignment, packet header information extraction and packet header removal. The switch fabric is the core of the switch and it is used for switching packets optically. The output interface is used to regenerate the optical signals and insert the packet header. The control unit controls the switch using the information in the packet headers. Because of synchronization requirements, optical packet switches are typically designed for fixed-size packets.

When a packet arrives at a WDM optical packet switch, it is first processed by the input interface. The header and the payload of the packet are separated, and the header is converted into the electrical domain and processed by the control unit electronically. The payload remains as an optical signal throughout the switch. After the payload passes through the switching fabric, it is re-combined with the header, which has been converted back into the optical domain, at the output interface.

In the following, we briefly describe some issues of optical packet switches. For more information about synchronization and contention resolution, the reader is referred to [67].

Packet coding techniques. Several optical packet coding techniques have been studied. There are three basic categories, namely, bit-serial, bit-parallel, and out-of-band-signaling. Bit-serial coding can be implemented using optical code division multiplexing (OCDM), or optical pulse interval, or mixed rate techniques. In OCDM, each bit carries its routing information, while in the latter two techniques, multiple bits are organized into a packet payload with a packet header that includes routing information. The difference between the latter two techniques is that in optical pulse interval the packet header and payload are transmitted at the same rate, whereas in mixed rate technique the packet header is transmitted at a lower rate than the payload so that the packet header can be easily processed electronically. In bit-parallel coding, multi-bits are transmitted at the same time but on separate wavelengths. Out-of-band-signaling coding includes sub-carrier multiplexing (SCM) and dual wavelength coding. In SCM, the packet header is placed in an electrical subcarrier above the baseband frequencies occupied by the packet payload, and both are transmitted at the same time slot. In dual wavelength coding, the packet header and payload are transmitted in separate wavelengths but at the same time slot.

Contention resolution. Contention resolution is necessary in order to handle the case where more than one packet are destined to go out of the same output port at the same time. This is a problem that commonly arises in packet switches, and it is known as *external blocking* [68]. It is typically resolved by buffering all the contending packets, except one which is permitted to go out. In an optical packet switch, techniques designed to address the external blocking problem include *optical buffering*, *exploiting the wavelength domain*, and using *deflection routing*. Whether these prove to be adequate to address the external blocking problem is still highly doubtful. Below we discuss each of these solutions.

Optical buffering currently can only be implemented using optical delay lines (ODL). An ODL can delay a packet for a specified amount of time, which is related to the length of the delay line. Currently, optical buffering is the Achilles' heel of optical packet switches! Delay lines may be acceptable in prototype switches, but they are not commercially viable. The alternative, of course, is to convert the optical packet to the electrical domain and store it electronically. This is not an acceptable solution, since electronic memories cannot keep up with the speeds of optical networks.

There are many ways that an ODL can be used to emulate an electronic buffer. For instance, a buffer for N packets with a FIFO discipline can be implemented using N delay lines of different lengths. Delay line i delays a packet for i timeslots. A counter keeps track of the number of the packets in the buffer. It is decremented when a packet leaves the buffer, and it is incremented when a packet enters the buffer. Suppose that the value of the counter is j when a packet arrives at the buffer, then the packet will be routed to the j-th delay line.

Limited by the length of the delay lines, this type of buffer is usually small, and it does not scale up.

An alternative solution to optical buffering is to use the wavelength domain. In WDM, several wavelengths run on a fiber link that connects two optical switches. This can be exploited to minimize external blocking as follows. Let us assume that two packets are destined to go out of the same output port at the same time. Then, they can be still transmitted out but on two different wavelengths. This method may have some potential in minimizing external blocking, particularly since the number of wavelengths that can be coupled together onto a single fiber continues to increase. For instance, it is expected that in the near future there will be as many as 200 wavelengths per fiber. This method requires wavelength converters.

Finally, deflection routing is another alternative to solving the external blocking problem. Deflection routing is ideally suited to switches that have little buffer space. When there is a conflict between two packets, one will be routed to the correct output port, and the other will be routed to any other available output port. In this way, no or little buffer is needed. However, the deflected packet may end up following a longer path to its destination. As a result, the end-to-end delay for a packet may be unacceptably high. Also, packets will have to be re-ordered at the destination since they are likely to arrive in an out-of-sequence manner.

6.1 Optical Packet Switch Architectures

Various optical packet switch architectures that have been proposed in the literature. For a review of some of these architectures see [69] . Based on the switching fabric used, they have been classified in the following three classes: space switch fabrics, broadcast-and-select switch fabrics, and wavelength routing switch fabrics. For presentation purposes, we only give an example below based on a space switch fabric.

An architecture with a space switch fabric. A space switch fabric architecture is shown in Figure 10. The performance of this switch was analyzed in [70]. The switch consists of N incoming and N outgoing fiber links, with n wavelengths running on each fiber link. The switch is slotted, and the length of the slot is such that an optical packet can be transmitted and propagated from an input port to an output optical buffer.

The switch fabric consists of three parts: optical packet encoder, space switch, and optical packet buffer. The optical packet encoder works as follows. For each incoming fiber link, there is an optical demultiplexer which divides the incoming optical signal to the n different wavelengths. Each wavelength is fed to a different tunable wavelength converter (TWC) which converts the wavelength of the optical packet to a wavelength that is free at the destination optical output buffer. Then, through the space switch fabric, the optical packet can be switched to any of the N output optical buffers. Specifically, the output of a TWC is fed to a splitter which distributes the same signal to N different output fibers, one per output buffer. The signal on each of these output fibers goes through another

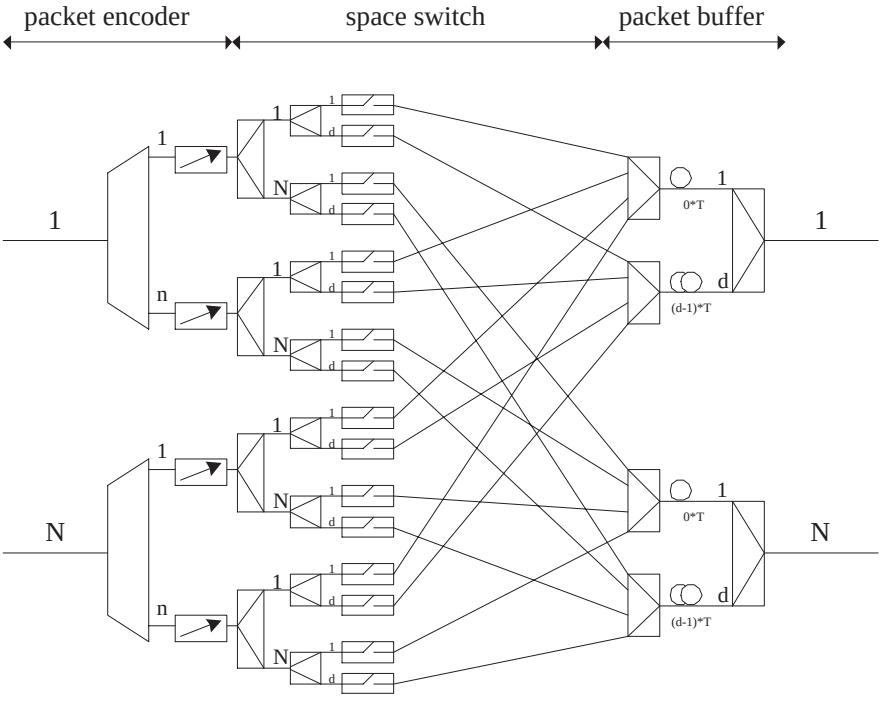

Fig. 10. An architecture with a space switch fabric

splitter which distributes it to $d+1$ different output fibers, and each output fiber is connected through an optical gate to one of the ODLs of the destination output buffer. The optical packet is forwarded to an ODL by appropriately keeping one optical gate open, and closing the remaining. The information regarding which wavelength a TWC should convert the wavelength of an incoming packet and the decision as to which ODL of the destination output buffer the packet will be switched to is provided by the control unit, which has knowledge of the state of the entire switch.

Each output buffer is an optical buffer implemented as follows. It consists of $d+1$ ODLs, numbered from 0 to d. ODL i delays an optical packet for a fixed delay equal to i slots. ODL 0 provides zero delay, and a packet arriving at this ODL is simply transmitted out of the output port. Each ODL can delay optical packets on each of the n wavelengths. For instance, at the beginning of a slot, ODL 1 can accept up to n optical packets, one per wavelength, and delay them for 1 slot. ODL 2 can accept up to n optical packets at the beginning of each time slot, and delay them for 2 slots. That is, at slot t, it can accept up to n packets (one per wavelength) and delay them for 2 slots, in which case, these packets will exit at the beginning of slot $t+2$. However, at the beginning of slot $t+1$, it can also accept another batch of n optical packets. Thus, a maximum of

$2n$ packets may be in transit within ODL 2. Similarly for ODL 3 through d. Let c_i denote the number of optical packets on wavelength i, where $i = 1, 2, \cdots, n$. We note that these c_i optical packets may be on a number of different ODLs. To insert an optical packet into the buffer, we first check all the c_i counters to find the smallest one, say c_s, then we set the TWC associated with this optical packet to convert the packet's wavelength to s, increase c_s by one, and switch the optical packet to ODL c_s. If the smallest counter c_s is larger than d, the packet will be dropped.

7 Optical Burst Switching

Optical burst switching (OBS) is a technique for transmitting bursts of traffic through an optical transport network by reserving resources through the optical network for only one burst. This technique is an adaptation of an ITU-T standard for burst switching for ATM networks, known as ATM block transfer (ABT) [64]. It is a new technology that has not as yet been commercialized. The main idea of OBS is shown in Figure 11. End-devices A and B communicate via a network of OBS nodes by transmitting data in bursts. An OBS node can be seen as consisting of a switch fabric and a CPU which controls the switch fabric and also processes signalling messages. The switch fabric is an $N \times N$ switch, where each incoming or outgoing fiber has W wavelengths, and it switches incoming bursts to their requested output ports. It may or may not be equipped with converters. Early proposals for OBS required an OBS node to be equipped with optical buffers. However, more recently it has been proposed to use bufferless OBS nodes.

Let us consider now the flow of bursts from end-device A to B. For each burst, A first sends a SETUP message via a signaling channel [71] to its ingress switch announcing its intention to transmit a burst. Transmission of the burst takes place after a delay known as offset. The ingress switch processes the SETUP message and allocates resources in its switch fabric so that to switch the burst out of the destination output port. The SETUP message is then forwarded to the next OBS node, which processes the SETUP message and allocates resources to switch the burst through its switch fabric. This goes one until the SETUP message reaches the destination end-device B. Each node in the path of the burst allocates resources to switch the burst through its switch for just a single burst, and it frees these resources after the burst has come through. A burst is dropped if an OBS node does not have enough capacity to switch it through its switch fabric.

The burst may last a short period of time and it may contain several packets, such as IP packets, ATM cells, and frame relay frames. It may also last for a long time, like a lightpath. In view of this, OBS can be seen as lying in-between packet switching and circuit switching.

Several variants of OBS have been proposed, such as tell-and-go (TAG), tell-and-wait (TAW), just-enough-time (JET) [72], and just-in-time (JIT) [73,71]. In the tell-and-go scheme, the source transmits the SETUP message and imme-

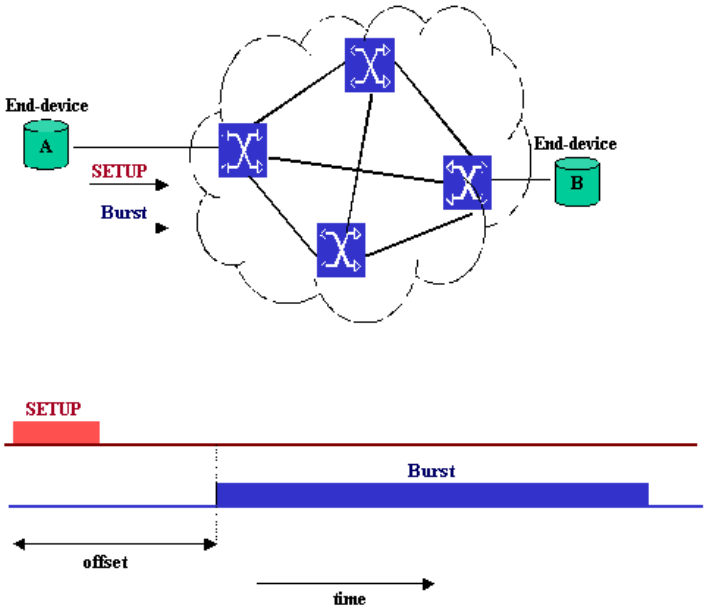

Fig. 11. An example of optical burst switching

diately after it transmits the optical burst. The tell-and-go scheme is inspired by one of the variants of the ATM block transfer (ABT) scheme in ATM network. However, within the setting of OBS, it is a rather idealistic scheme since there is no time for the receiving OBS node to process the SETUP message and to configure its switch fabric on time so that to transmit the incoming burst to its destination output port. In order to implement this scheme, either the OBS node has already been configured to switch the burst or some input optical buffering may be required to hold the burst until the node can process the SETUP message.

The tell-and-wait (TAW) scheme is the opposite of TAG, and it was inspired by another variant of the ATM block transfer (ABT) scheme. In this case, the SETUP message is propagated all the way to the receiving end-device, and each OBS node along the path processes the SETUP message and allocates resources within its switch fabric. A positive acknowledgement is returned to the transmitting end-device, upon receipt of which the end-device transmits its bursts. In this case, the burst will go through without been dropped at any OBS node. The offset can be seen as being equal to the round trip propagation delay plus the sum of the processing delays of the SETUP message at each OBS node along the path. In the just-enough-time (JET) and just-in-time (JIT) schemes, there is a delay between the transmission of the control packet and the transmission of the optical burst. This delay has to be larger than the sum of the

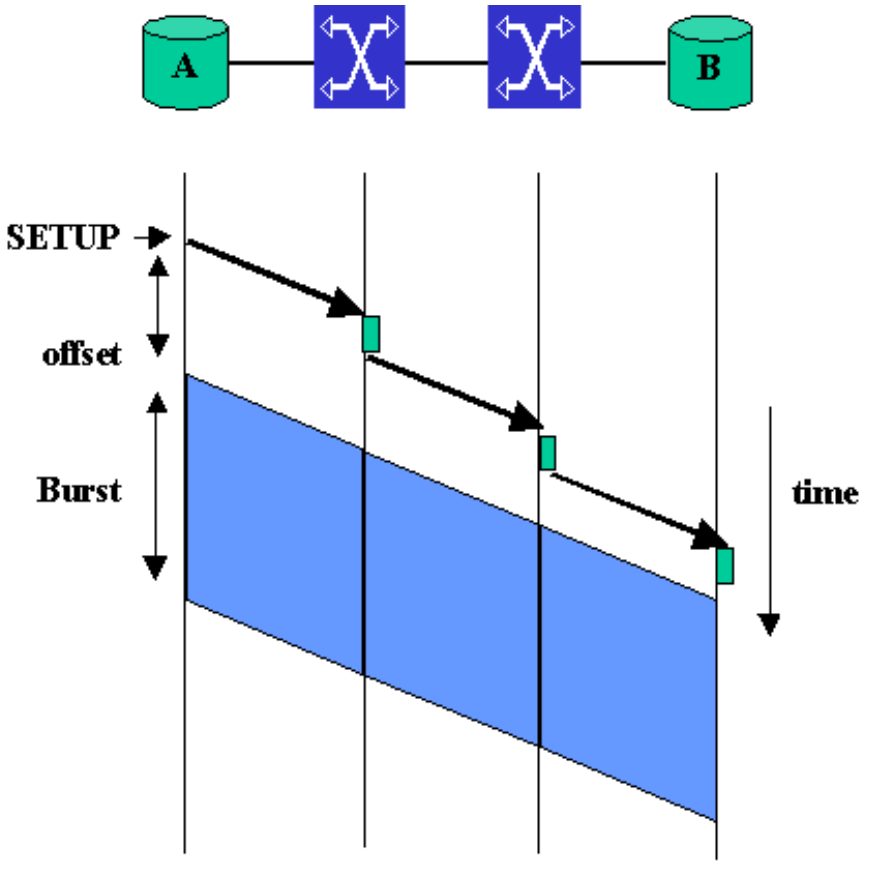

Fig. 12. The offset has to be larger than the sum of processing times

total processing times of the SETUP message at all the OBS nodes along the path. In this way, when the burst arrives at an OBS node, the SETUP message has already been processed and resources have been allocated so as to switch the burst through the switch fabric. An example is shown in Figure 12. The two schemes JET and JIT vary significantly in their proposed implementation. One of the main design issues has to do with the time at which an OBS node should configure its switch fabric to receive the pending burst. There are two alternatives, namely, *immediate configuration* and *estimated configuration*. In the former case, the OBS node allocates resources to the incoming burst immediately after it processes the SETUP message, whereas in the latter case, it allocates the necessary resources later on at a time when it estimates that the burst will arrive at the node. Obviously, in the case of immediate configuration, resources go unused until the burst arrives, whereas in the estimated configuration scheme the resources are better utilized. However, the immediate configuration scheme is considerably simpler to implement. JET uses estimated configuration, whereas

JIT proposes to use immediate configuration. Another design issue has to do with how long does an OBS node keep the resources allocated to a burst. Again, we distinguish two alternatives, namely, timed bursts and explicit release bursts. In the former case, the length of the burst is indicated in the SETUP message, and as a result, the OBS node calculate how long it will keep its resources allocated to the burst. In the latter case, the OBS node keeps the resources allocated to the burst until it receives an explicit release message. In JET it was also proposed a scheme to supports quality of service. Specifically, two traffic classes were defined, namely, real-time and non-real-time. A burst belonging to the real-time class is allocated a higher priority than a burst belonging to the non-real-time class, by simply using an additional delay between the transmission of the control packet and the transmission of the burst. The effect of this additional delay is that it reduces the blocking probability of the real-time burst at the optical burst switch.

References

1. D. O. Awduche. MPLS and traffic engineering in IP networks. *IEEE Communications*, 37(12):42–47, December 1999.
2. D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus. Requirements for traffic engineering over MPLS. RFC 2702, September 1999.
3. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated services. RFC 2475, December 1998.
4. E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol label switching architecture. RFC 3031, January 2001.
5. B. Davie and Y. Rekhter. *MPLS Technology and Applications*. Morgan Kaufmann Publishers, San Diego, California, 2000.
6. P. Ashwood-Smith *et al.* Generalized MPLS – signaling functional description. IETF Draft <draft-ietf-mpls-generalized-signaling-06.txt>, April 2001. Work in progress.
7. D. H. Su and D. W. Griffith. Standards activities for MPLS over WDM networks. *Optical Networks*, 1(3), July 2000.
8. O. Gerstel, B. Li, A. McGuire, G. N. Rouskas, K. Sivalingam, and Z. Zhang (Eds.). Special issue on protocols and architectures for next generation optical WDM networks. *IEEE Journal Selected Areas in Communications*, 18(10), October 2000.
9. R. Dutta and G. N. Rouskas. A survey of virtual topology design algorithms for wavelength routed optical networks. *Optical Networks*, 1(1):73–89, January 2000.
10. E. Leonardi, M. Mellia, and M. A. Marsan. Algorithms for the logical topology design in WDM all-optical networks. *Optical Networks*, 1(1):35–46, January 2000.
11. Y. Zhu, G. N. Rouskas, and H. G. Perros. A path decomposition approach for computing blocking probabilities in wavelength routing networks. *IEEE/ACM Transactions on Networking*, 8(6):747–762, December 2000.
12. L. Li and A. K. Somani. A new analytical model for multifiber WDM networks. *IEEE Journal Selected Areas in Communications*, 18(10):2138–2145, October 2000.
13. S. Ramamurthy and B. Mukherjee. Survivable WDM mesh networks, part I – protection. In *Proceedings of INFOCOM '99*, pages 744–751, March 1999.
14. S. Ramamurthy and B. Mukherjee. Survivable WDM mesh networks, part II – restoration. In *Proceedings of ICC '99*, pages 2023–2030, June 1999.

15. A. Mokhtar and M. Azizoglu. Adaptive wavelength routing in all-optical netowrks. *IEEE/ACM Transactions on Networking*, 6(2):197–206, April 1998.
16. E. Karasan and E. Ayanoglu. Effects of wavelength routing and selection algorithms on wavelength conversion gain in WDM optical networks. *IEEE/ACM Transactions on Networking*, 6(2):186–196, April 1998.
17. H. Zang, J. P. Jue, and B. Mukherjee. A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks. *Optical Networks*, 1(1):47–60, January 2000.
18. B. Ramamurthy and B. Mukherjee. Wavelength conversion in WDM networking. *IEEE Journal Selected Areas in Communications*, 16(7):1061–1073, September 1998.
19. S. Subramaniam, M. Azizoglu, and A. Somani. All-optical networks with sparse wavelength conversion. *IEEE/ACM Transactions on Networking*, 4(4):544–557, August 1996.
20. T. Tripathi and K. Sivarajan. Computing approximate blocking probabilities in wavelength routed all-optical networks with limited-range wavelength conversion. In *Proceedings of INFOCOM '99*, pages 329–336, March 1999.
21. N. Ghani. Lambda-labeling: A framework for IP-over-WDM using MPLS. *Optical Networks*, 1(2):45–58, April 2000.
22. L. H. Sahasrabuddhe and B. Mukherjee. Light-trees: Optical multicasting for improved performance in wavelength-routed networks. *IEEE Communications*, 37(2):67–73, February 1999.
23. D. Papadimitriou *et al.* Optical multicast in wavelength switched networks – architectural framework. IETF Draft <draft-poj-optical-multicast-01.txt>, July 2001. Work in progress.
24. B. Mukherjee. *Optical Communication Networking*. McGraw-Hill, 1997.
25. V. Sharma and E. A. Varvarigos. Limited wavelength translation in all-optical WDM mesh networks. In *Proceedings of INFOCOM '98*, pages 893–901, March 1999.
26. S. L. Hakimi. Steiner's problem in graphs and its implications. *Networks*, 1:113–133, 1971.
27. Y. Xin, G. N. Rouskas, and H. G. Perros. On the design of MPλS networks. Technical Report TR-01-07, North Carolina State University, Raleigh, NC, July 2001.
28. S. Subramaniam, M. Azizoglu, and A. K. Somani. On the optimal placement of wavelength converters in wavelength-routed networks. In *Proceedings of INFOCOM '98*, pages 902–909, April 1998.
29. M. Ali and J. Deogun. Allocation of splitting nodes in wavelength-routed networks. *Photonic Network Communications*, 2(3):245–263, August 2000.
30. R. Ramaswami and K. N. Sivarajan. Design of logical topologies for wavelength-routed optical networks. *IEEE Journal Selected Areas in Communications*, 14(5):840–851, June 1996.
31. D. Banerjee and B. Mukherjee. A practical approach for routing and wavelength assignment in large wavelength-routed optical networks. *IEEE Journal Selected Areas in Communications*, 14(5):903–908, June 1996.
32. B. Mukherjee *et al.* Some principles for designing a wide-area WDM optical network. *IEEE/ACM Transactions on Networking*, 4(5):684–696, October 1996.
33. Z. Zhang and A. Acampora. A heuristic wavelength assignment algorithm for multihop WDM networks with wavelength routing and wavelength reuse. *IEEE/ACM Transactions on Networking*, 3(3):281–288, June 1995.

34. I. Chlamtac, A. Ganz, and G. Karmi. Lightnets: Topologies for high-speed optical networks. *Journal of Lightwave Technology*, 11:951–961, May/June 1993.
35. S. Banerjee and B. Mukherjee. Algorithms for optimized node placement in shufflenet-based multihop lightwave networks. In *Proceedings of INFOCOM '93*, March 1993.
36. R. Malli, X. Zhang, and C. Qiao. Benefit of multicasting in all-optical networks. In *Proceedings of SPIE*, volume 3531, pages 209–220, November 1998.
37. G. Sahin and M. Azizoglu. Multicast routing and wavelength assignment in wide-area networks. In *Proceedings of SPIE*, volume 3531, pages 196–208, November 1998.
38. E. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, 1976.
39. D. Bertsekas and R. Gallager. *Data Networks*. Prentice Hall, Inc., Englewood Cliffs, NJ, 1992.
40. X. Zhang, J. Y. Wei, and C. Qiao. Constrained multicast routing in WDM networks with sparse light splitting. *Journal of Lightwave Technology*, 18(12):1917–1927, December 2000.
41. J. Strand, A. L. Chiu, and R. Tkach. Issues for routing in the optical layer. *IEEE Communications*, pages 81–96, February 2001.
42. J. Moy. OSPF version 2. RFC 2328, April 1998.
43. A. Chiu *et al.* Impairments and other constraints on optical layer routing. IETF Draft <draft-ietf-ipo-impairments-00.txt>, May 2001. Work in progress.
44. Y. Zhu, G. N. Rouskas, and H. G. Perros. A comparison of allocation policies in wavelength routing networks. *Photonic Network Communications*, 2(3):265–293, August 2000.
45. Z. Zhang, J. Fu, D. Guo, and L. Zhang. Lightpath routing for intelligent optical networks. *IEEE Network*, 15(4):28–35, July/August 2001.
46. C. Assi, M. Ali, R. Kurtz, and D. Guo. Optical networking and real-time provisioning: An integrated vision for the next-generation internet. *IEEE Network*, 15(4):36–45, July/August 2001.
47. S. Sengupta and R. Ramamurthy. From network design to dynamic provisioning and restoration in optical cross-connect mesh networks: An architectural and algorithmic overview. *IEEE Network*, 15(4):46–54, July/August 2001.
48. The internet engineering task force. http://www.ietf.org.
49. Optical domain service interconnect. http://www.odsi-coalition.com.
50. The optical internetworking forum. http://www.oiforum.com.
51. G. Bernstein, R. Coltun, J. Moy, A. Sodder, and K. Arvind. ODSI functional specification version 1.4. ODSI Coalition, August 2000.
52. User network interface (UNI) 1.0 signaling specification. OIF2000.125.6, September 2001.
53. B. Rajagopalan *et al.* IP over optical networks – a framework. IETF Draft <draft-many-ip-optical-framework-03.txt>, March 2001. Work in progress.
54. J. P. Lang *et al.* Link management protocol (LMP). IETF Draft <draft-ietf-mpls-lmp-02.txt>, September 2001. Work in progress.
55. K. Kompella *et al.* OSPF extensions in support of generalized MPLS. IETF Draft <draft-ietf-ccamp-ospf-gmpls-extensions-00.txt>, September 2001. Work in progress.
56. D. Katz, D. Yeung, and K. Kompella. Traffic engineering extensions to OSPF. IETF Draft <draft-katz-yeung-ospf-traffic-06.txt>, October 2001. Work in progress.

57. D. Awduche *et al.* RSVP-TE: Extensions to RSVP for LSP tunnels. IETF Draft <draft-ietf-mpls-rsvp-lsp-tunnel-08.txt>, February 2001. Work in progress.
58. O. Aboul-Magd *et al.* Constraint-based LSP setup using LDP. IETF Draft <draft-ietf-mpls-cr-ldp-05.txt>, February 2001. Work in progress.
59. R. Braden *et al.* Resource reservation protocol – version 1. RFC 2205, September 1997.
60. L. Andersson, P. Doolan, N. Feldman, A. Fredette, and B. Thomas. LDP specification. RFC 3036, January 2001.
61. P. Ashwood-Smith *et al.* Generalized MPLS signaling – RSVP-TE extensions. IETF Draft <draft-ietf-mpls-generalized-rsvp-te-05.txt>, October 2001. Work in progress.
62. P. Ashwood-Smith *et al.* Generalized MPLS signaling – CR-LDP extensions. IETF Draft <draft-ietf-mpls-generalized-cr-ldp-04.txt>, July 2001. Work in progress.
63. B. Rajagopalan, J. Luciani, D. Awduche, B. Cain, B. Jamoussi, and D. Saha. IP over optical networks: A framework. IETF Draft <draft-ietf-ipo-framework-01.txt>, February 2002. Work in progress.
64. H. Perros. *An Introduction to ATM Networks.* Wiley, 2001.
65. D. Durham (Ed.), J. Boyle, R. Cohen, S. Herzog, R. Rajan, and A. Sastry. The COPS (common open policy service) protocol. RFC 2748, January 2000.
66. S. Herzog (Ed.), J. Boyle, R. Cohen, D. Durham, R. Rajan, and A. Sastry. COPS usage for RSVP. RFC 2749, January 2000.
67. S. Yao, S. Dixit, and B. Mukherjee. Advances in photonic packet switching: An overview. *IEEE Communications*, 38(2):84–94, February 2000.
68. H. Perros. *Queueing Networks with Blocking: Exact and Approximate Solutions.* Oxford University Press, 1994.
69. L. Xu, H. G. Perros, and G. N. Rouskas. Techniques for optical packet switching and optical burst switching. *IEEE Communications*, 39(1):136–142, January 2001.
70. S. L. Danielsen *et al.* Analysis of a WDM packet switch with improved performance under bursty traffic conditions due to tunable wavelength converters. *IEEE/OSA Journal of Lightwave Technology*, 16(5):729–735, May 1998.
71. I. Baldine, G. N. Rouskas, H. G. Perros, and D. Stevenson. `JumpStart`: A just-in-time signaling architecture for WDM burst-switched networks. *IEEE Communications*, 40(2):82–89, February 2002.
72. C. Qiao and M. Yoo. Optical burst switching (OBS)-A new paradigm for an optical Internet. *Journal of High Speed Networks*, 8(1):69–84, January 1999.
73. J. Y. Wei and R. I. McFarland. Just-in-time signaling for WDM optical burst switching networks. *Journal of Lightwave Technology*, 18(12):2019–2037, December 2000.