PEGAH SEDDIGHIAN

OPTICAL PACKET SWITCHING USING MULTI-WAVELENGTH LABELS

Thèse de doctorat présentée
à la Faculté des études supérieures de l'Université Laval
dans le cadre du programme de doctorat en Génie Électrique
pour l'obtention du grade de Philosophiæ Doctor (Ph. D.)

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE ET DE GÉNIE INFORMATIQUE
FACULTÉ DES SCIENCES ET DE GÉNIE
UNIVERSITÉ LAVAL
QUÉBEC

2008

To my beloved parents

Résumé

Nous étudions les réseaux optiques pouvant transporter un trafic de données de type Internet. Notre objectif est de transposer les paquets électroniques de données en des paquets optiques et d'effectuer un routage de manière tout-optique. Nous utilisons un multi protocole généralisé par commutation d'étiquettes (GMPLS), où une étiquette optique est assignée à chaque paquet et est utilisée pour le routage. Nous proposons deux structures de réseau différentes basées sur des étiquettes multi longueurs d'onde. Nous contournons les principaux désavantages des scénarios GMPLS optiques proposés précédemment, c'est-à-dire les pertes de fractionnement et les technologies complexes requises. Les structures de réseau proposées sont pratiques, haute-vitesse, simples, redimensionnables et peu coûteuses.

Pour la première approche, nous utilisons un encodage spectral d'amplitude (SAC) pour les étiquettes afin d'accomplir le routage très haut débit des paquets. Nous proposons de superposer des étiquettes SAC pour réaliser un adressage hiérarchique et ainsi réduire la taille des tables de correspondance d'adresses ainsi que les pertes de fractionnement. Nous examinons expérimentalement deux formats de paquets optiques, l'un avec étiquette SAC séparable, et l'autre avec données directement encodées par SAC.

Pour la seconde approche, nous proposons une structure de réseau basée sur des étiquettes multi longueurs d'onde binaires. Les pertes de fractionnement sont éliminées pour cette approche, la rendant ainsi encore plus facilement redimensionnable que notre proposition SAC. Avec cette technique, les bits des étiquettes sont associés à une sélection de cases fréquentielles optiques. Au nœud, les paquets de longueurs variables sont auto routés par un commutateur multi étages alors que chaque bit de l'étiquette contrôle un étage. Le nœud est redimensionnable et possède des pertes d'insertion fixes. Nous proposons également une solution pour alléger les traitements sophistiqués associés à la substitution d'étiquettes dans les réseaux GMPLS. Nous multiplexons temporellement les étiquettes multi longueurs d'onde binaires pour le chemin complet de commutation optique. Comme démonstration de faisabilité, nous examinons expérimentalement les performances des approches proposées.

Finalement, nous proposons une solution pour résoudre les collisions, ce qui n'était pas considéré dans les deux premières structures. Nous considérons qu'une topologie simplifiée, soit une topologie à une liaison à un bond. Les nœuds aux frontières sont coordonnés temporellement avec les nœuds centraux lors de l'établissement du réseau, donc la synchronisation optique en temps continu n'est plus nécessaire. Un algorithme de planification non centralisé aléatoire est utilisé pour éliminer les collisions; aucun tampon optique n'est nécessaire. Un algorithme simple de graphe bipartite est proposé afin de déterminer les connections au commutateur central. Nous simulons le réseau pour un type de trafic réaliste; les résultats confirment la bonne performance de l'algorithme de planification et démontrent que l'architecture proposée est pratique et bien adaptée aux réseaux optiques de commutation par paquets.

Resume

We investigate optical networks capable of carrying data-type traffic. Our objective is to map Internet packets into optical packets and route them all-optically. We employ generalized multi-protocol label switching (GMPLS), where an optical label used for routing is assigned to each packet. We propose two different network structures based on multi-wavelength labels. We resolve the main drawbacks of previously proposed scenarios that are impractical and expensive due to high splitting loss and the complex technologies required. Our proposed network structures are practical, high-speed, simple, scalable, and low-cost.

In the first approach, we use spectral amplitude codes (SAC) as labels, to accomplish ultra-fast packet forwarding. We propose stacking SAC-labels for hierarchical addressing, to reduce the size of lookup tables and splitting loss. We experimentally examine two optical packet formats, one with separable SAC-labels, and the other with SAC-encoded payloads.

In the second approach, we propose a network structure based on binary multi-wavelength labels. Splitting losses are eliminated in this approach, rendering it even more scalable than our SAC proposal. In this scheme, the label is mapped bit-by-bit to a selection of wavelength bins. At the forwarding node, variable-length packets are self-forwarded over a multi-stage switch where each label bit controls a switch stage. The forwarding node is scalable and has fixed insertion loss. We also propose a solution to alleviate the sophisticated label swapping processing required in GMPLS networks. We time-multiplex the binary multi-wavelength labels for the entire optical label-switching path. We examine the performance of the proposed schemes experimentally as a proof of concept.

Finally, we propose a solution for contention resolution, not addressed in the first two structures. We simplify the network topology to single-hop; the edge nodes are time-coordinated with the core nodes, thus optical synchronizers are not required. A non-centralized randomized scheduling algorithm is used to resolve contention; no optical buffer is required. A simple bipartite-graph matching algorithm is proposed to determine

the connections at the core switch. We simulate the network for a realistic traffic type; the results confirm the good performance of the scheduling algorithm and establish that the proposed architecture is practical and desirable for packet-switched optical networks.

Acknowledgement and Thanks

I would like to express my sincere gratitude to my advisor, Professor Leslie A. Rusch, for her endless encouragement, support and guidance. Her insight and knowledge makes her a great professor, but her gentle, caring, compassionate character makes her a remarkable person. In spite of her busy schedule, I never had any difficulty to arrange a meeting with her to discuss scientific or personal problems, and I always enjoyed her intelligence and vision. She was much more than a professor to me and helped me to adapt the myriad of changes I faced in a foreign country. She added more value to my PhD journey by giving me the opportunity to experience other research and life environments such as University of Toronto and TELUS in Calgary. It is my great honor to be her student.

I would also like to thank my co-advisor, Professor Alberto Leon-Garcia, for hosting me at University of Toronto for seven months. I truly appreciate the invaluable discussions about networking aspects of my project, and his bright ideas and comments.

I am grateful to Professor Gabriella Cincotti, Professor Lacra Pavel, and Professor Lawrence Chen for serving on my committee and for their constructive comments on the final thesis.

Many thanks to my talented, professional, experienced colleagues: Simon Ayotte, Mohammad Abtahi, Julien Penon, Walid Mathlouthi, and all the rest, who helped me in various forms to finalize my project. They always put the same amount of time and energy in brainstorming my problems, as they would put for their own. Special thanks to Simon for the motivating discussions, teaching me experimental work, and sharing with me his technical experience. The first packet switching experiment would still be far from completion without his collaboration. The wonderful friendships with my colleagues have made me think of the university as my second home.

I would like to acknowledge and thank the COPL staff and technicians for their assistance, particularly Patrick LaRochelle for designing and implementing the electronics used in the experimental setups.

My deep love and appreciation goes to my parents for their support, kindness and patience in all and every stage of my life. They did everything in their power to advance my education and encouraged me to love books, learning, and science. I owe a great debt of gratitude to them. Many thanks to my sisters and friends for their support and encouragement, and for having an open ear to my complaints on my down days.

My sincere gratitude to you all,

Pegah Seddighian

Table of Contents

List of Figures

List of Acronyms

APC	Autocorrelation Peak
ASE	Amplified Spontaneous Emission
ATM	Asynchronous Transfer Mode
ATT	Attenuator
AWG	Arrayed Waveguide Grating
BBS	Broadband Source
BER	Bit Error Rate
BERT	Bit Error Rate Tester
BIBD	Balanced Incomplete Block Design
BPF	Band Pass Filter
BPSK	Binary Phase Shift Keying
CCP	Cross-Correlation Peak
C-DEC	Complementary Decoder
DCF	Dispersion Compensating Fiber
DEC	Decoder
DFB	Distributed Feedback
DFG	Difference Frequency Generation
EAM	Electro Absorption Modulator
ED	Envelope Detector
E-DELAY	Electronic Delay
EDFA	Erbium Doped Fiber Amplifier
ENC	Encoder
E-TH	Electrical Thresholder
EXP	Experimental
FBG	Fiber Bragg Gratings
FDL	Fiber Delay Line
FEC	Forwarding Equivalence Class
FWHM	Full-Width At Half-Maximum
FWM	Four Wave Mixing
GMPLS	Generalized MPLS
ISO	Isolator
LN-SW	$Linbo_3$ Optical Gate Switch
LPF	Low-Pass Filter
LSP	Label Switched Path
MLL	Mode Locked Laser
MMB	Markov-Modulated Bernoulli
MOD	Modulator
MPLS	Multi-Protocol Label Switching
M-λ	Multi-Wavelength
NRZ	Non-Return To Zero
N-SOA	Nonlinear SOA
O/E	Optical To Electrical Converter

OC	Optical Code
OCDMA	Optical Code Division Multiple Access
OEO	Optical-Electronic-Optical
O-FILTER	Optical Filters
OOK	On-Off Keying
OXC	Optical Cross Connects
PC	Polarization Controllers
PD	Photodiode
PLC	Planar Lightwave Circuit
PLSR	Photonic Label Switching Router
PPLN	Periodically-Poled Lithium-Niobate
PRBS	Pseudo Random Bit Sequence
PSK	Phase-Shift Keying
SAC	Spectral Amplitude Code
SCM	Subcarrier Multiplexing
SE	Switching Elements
SFRL	Semiconductor Fiber Ring Laser
SMF	Single Mode Fiber
SOA	Semiconductor Optical Amplifier
SW	Switch
TDM	Time-Division Multiplexed
T-FF	T-Flip-Flop
TTL	Time-To-Live
VCI	Virtual Channel Identifier
VOA	Variable Optical Attenuator
VPI	Virtual Path Identifier
WDM	Wavelength Division Multiplexing
WRS	Wavelength Recognizing Switch
XGM	Cross Gain Modulation

Chapter 1

Introduction

1.1. Motivation

Today's information networks carry a larger and larger percentage of data compared to voice traffic. The optical network infrastructure must adapt to optimally deliver predominantly data traffic, while still providing the quality of service required for voice and video transmission. The transmission of data in most of today's optical networks is performed by passing through several layers: 1) IP for applications, 2) asynchronous transfer mode (ATM) for traffic engineering, 3) SONET or SDH for transmission, and 4) wavelength division multiplexing (WDM) for capacity (see Figure 1.1) [1]. A multi-layer

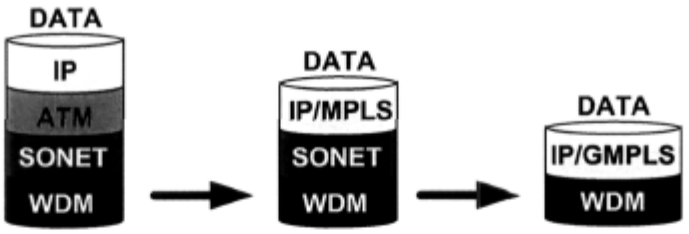

Figure 1.1. The evolution toward optical network.

approach is not optimum for data transmission as each layer needs to be managed separately, running its own addressing, routing, and management standards and resulting in inefficient overhead and a waste of resources. Furthermore, optical-electronic-optical (OEO) signal conversion is required at each node, which, along with passage through several layers, slows down the throughput. Accordingly, some effort has been made to reduce the number of layers as much as possible. To this end, the IP and ATM layers have been integrated by introducing multi-protocol label switching (MPLS) technology. The idea of MPLS is to forward IP packets at very high speed through label-switched connection-oriented paths. Later, the generalized version of MPLS (GMPLS) was proposed where the label entities could be anything suitable to identity the traffic, such as position in time, wavelength, or physical port. The particular case of GMPLS that is based on wavelength switching is MPλS. In this scheme, forwarding information is inherent in the signal wavelength, and is accomplished by optical cross connects (OXC) at intermediate nodes for switching. Although the electronic bottleneck is resolved by MPλS, it still faces many challenges. The main issue is switching granularity, which cannot be finer than a full channel wavelength, resulting in very inefficient use of available capacity in the network. Recently label entities other than wavelength have been used to overcome the granularity problem. The common objective of all these approaches is to optically switch the data to realize a network independent of the original format and bit rate of traffic. The label is processed either electronically or optically to extract the forwarding information. However, electronic label-processing is time- and power-consuming and will impose a bottleneck that limits the high transmission capacity of optical fibers. On the other hand, optical components capable of advanced operations such as logic gates and buffers are still in the

research stage. Therefore, a wise design considers the capabilities and limitations of photonic technology as well as the challenges of label switched networks.

Extensive results on the design and investigation of optical packet switched networks are reported in the literature. For instance, in [2,3] an explicit optical label that is encoded using on-off keying (OOK) modulation precedes the payload, and optical label recognition is performed using correlation-based structures. A subcarrier modulated label scheme is described in [4,5], where label processing is performed in the electrical domain. Employing optical code division multiple access (OCDMA) codes as label entities is another approach [6-8], where correlators are used to detect OCDMA labels optically. The main obstacle of correlation-based label recognition schemes is high splitting loss and component count that limits the network scalability. The optical label can also be encoded using several wavelength channels [9-15]. We will discuss these options for GMPLS in chapter 2. This thesis focuses on multi-wavelength labels due to many potential advantages, including label extraction with passive optical filtering, low bit rate label transmission, high-speed processing, scalability, and cost efficiency.

1.2. Objectives

Our main objective is to propose a multi-hop optical network architecture in which high-speed routing is performed based on optical labels. The network is connection-oriented, as in GMPLS, and the topology is mesh. The forwarding and control components in a node are separate optical and electrical modules, similar to GMPLS. Sophisticated control and routing procedures run over an electrical overlay. A virtual connection is established between the source and destination through the optical backbone. Accordingly, a label is associated to users' traffic. The forwarding mechanism, which is the focus of this thesis, is performed optically using the label. In contrast to MPλS where the data is carried over lambda paths, here, the network is capable of packet-by-packet label switching, providing greater functionality and finer switching granularity. The main features that we seek for this network are:

1) provision of a rich set of labels

2) compatibility with hierarchical addressing

3) high-speed label recognition and label swapping

4) scalability (increases in label cardinality requiring only moderate increases in hardware)

5) efficiency in implementation and reasonable cost.

1.3. Our Contribution

We propose an optical network architecture where the electrical packets are converted to optical packets at the edge nodes of the optical domain. The entire electrical packet is converted to the payload of the optical packet, and an optical label containing routing information is added.

The main contribution of this thesis is two novel optical label formats and the corresponding forwarding node architectures. In both approaches the label is encoded using a group of wavelength channels. With the available wavelengths, we either create labels that have the typical auto-correlation and cross-correlation properties of OCDMA codes, or we create labels that map each label bit to a distinct wavelength. In the latter case, no particular correlation properties are assumed. In either case, the labels are transmitted simultaneously with the payload resulting in zero time overhead.

For the first solution, we use spectral amplitude codes (SAC) [16-19] as optical labels due to their low-cost implementation and ultra-fast correlation-based label recognition. We mitigate the splitting loss associated with correlation-based processing by multiplexing codes (optical labels) to realize label-stacking and take advantage of hierarchical addressing [13]. In hierarchical addressing the size of the look-up tables in each switching node is reduced resulting in lower splitting loss in correlation-based nodes [20]. As the splitting loss remains the major impairment in this solution, our second approach avoids splitting loss entirely.

Secondly, we propose binary multi-wavelength (M-λ) labels where a multi-stage tree switch is exploited at the forwarding node [12]. We encode label bits with distinct wavelength channels and transmit them in parallel with the payload. At the forwarding node, the label bits are separated using an arrayed waveguide grating (AWG); each channel

controls a stage of a multi-stage switch. Suppose we have an N port switch and GMPLS signaling with N labels, each specifying an output port. This second approach gives maximum spectral efficiency, *i.e.*, L available label wavelengths supports $N = 2^L$ ports. Correlation-based techniques can only support much fewer labels as good auto-correlation and cross-correlation properties must be respected so that not all possible 2^L labels are suitable. Furthermore, splitting loss of labels for correlation-based processes is linear with N. In contrast, this second approach can exploit an AWG, so that losses are fixed and N can grow without increasing losses. This method is practical, high-speed, simple, and scalable. The packet forwarding over the node is on-the-fly.

Label swapping in both approaches can be performed by wavelength conversion [21-23]. However, determining the new label requires OEO conversion. We propose an alternative solution to label swapping, that is, time-stacking all optical labels required to reach the destination edge node, and transmitting the entire stack with the packet. In this case, all the sophisticated processing is performed in the electrical edge nodes, ensuring a simple, ultra-fast, and flexible optical backbone. We have experimentally examined all our proposed approaches and verified their functionality, as well as the bit error rate performance [13,12].

In the previously mentioned network structures, we did not address the packet contention problem. We assumed simply that in the case of contention only one packet among the contending packets would be delivered to the output ports and all others would be lost. Contention resolution remains an open research problem under investigation for mesh topologies. In the penultimate chapter of this thesis, we examine contention resolution in a much simpler (not mesh) network topology. We examine a randomized scheduling algorithm to mitigate contention. The algorithm reduces the probability of conflict and therefore improves the packet loss performance. We proposed a low complexity bipartite-graph matching algorithm and analyze the performance of the star network under a realistic traffic model [24].

1.4. Structure of the Thesis

The motivation for the proposed research, the overall objectives and our contributions were outlined in this chapter. A review of the principles of photonic multi-protocol label

switching networks is addressed in chapter 2. In chapter 3, we survey different optical networks employing labels composed of: 1) time domain signaling, 2) subcarrier frequency modulation, 3) optical CDMA codes, and 4) wavelength domain signaling. In chapter 4, we present our first contribution on label-switched networks: label stacking with spectral amplitude codes and correlation-based node structure. We discuss our second proposal about binary M-λ labels with multi-stage tree switches, followed by time-stacked labels in chapter 5. In chapter 6, we introduce a simple practical optical network topology and analyze a scheduling algorithm for contention resolution. The last chapter concludes the thesis.

Chapter 2

Overview of Optical Label Switched Networks

2.1. Introduction

Multi-protocol label switching (MPLS) was introduced to add new functionalities to IP networks. By performing label switching, MPLS is capable of forwarding packets at high speeds employing hardware lookup tables. In label switched networks, packets are transmitted through pre-determined paths that are decided at the edge nodes. The intermediate nodes are responsible for forwarding packets according to the packet label, as well as being responsible for label swapping. The concept of MPLS can be well adapted to

optical packet switched networks. The separation of routing and forwarding procedures in MPLS nodes is a key enabler for optical networks. The sophisticated routing and control processes can be performed in an electrical overlay to establish label switching paths. The relatively simple forwarding procedure, including label recognition and swapping, is performed by ultra-fast optical modules.

In this chapter we will briefly review MPLS network principles. We will focus on optical networks that are based on the label switching concept. The special characteristics of such optical networks such as label processing, packet format and contention resolution are discussed in this chapter.

2.2. Multi-Protocol Label Switching (MPLS)

MPLS has been designed to combine advantages of IP and ATM networks [20,25]. As in ATM, MPLS technology provides fast packet forwarding using label-switching. Unlike IP networks in which destination-based table lookup is performed at each node, MPLS is based on short label forwarding. In IP routers the header of the received packet is recorded in a register and then is compared to the entries of a lookup table using a longest prefix matching algorithm. The destination-based table lookup that provides the outgoing path is implemented in software, therefore, time consuming (on the order of microseconds). In MPLS networks, however, the destination address is only processed at the edge node. For any given destination, a label switched path (LSP) is established through the network prior to packet transmission by exchanging control messages between switching nodes and configuring the tables in the intermediate nodes. As shown in Figure 2.1, at the ingress node a label is attached to the packet to be used in the forwarding process. When the packet reaches an intermediate node, the output port and the output label are determined by the combination of input port and the received label. The MPLS lookup table of each node contains information regarding the next hop label (to be appended to the outgoing payload) and the path. The MPLS look up table is extremely short as compared to an IP lookup table. The process of removing the old label and attaching a new label is called label swapping (see Figure 2.1). At the egress node, the destination address will be recovered from the final label, and the packet will be delivered in its original format.

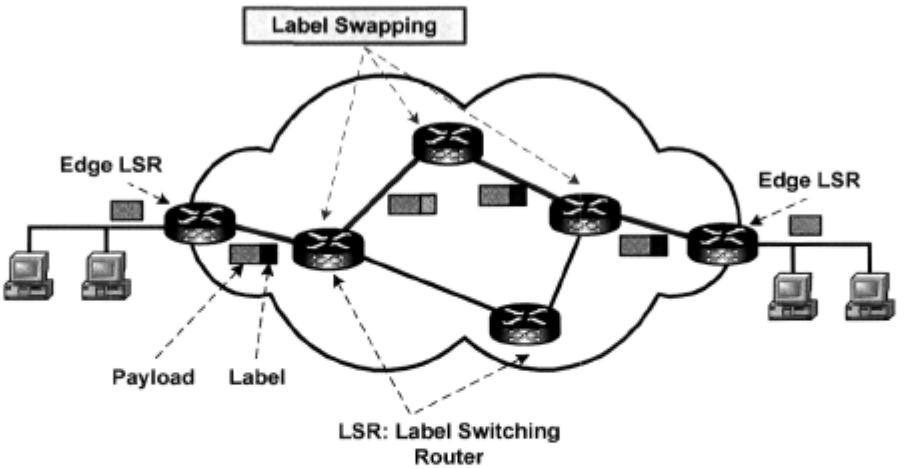

Figure 2.1. Principle of MPLS network.

The maximum number of label bits required for label switching networks is related to the maximum number of simultaneous paths that are established via an input port. Label switching requires much fewer label bits than those required for destination-based routing. As illustrated in Figure 2.2, the MPLS label consists of 32 bits: 20 bits for label, 3 bits for experimental functions, one bit for stack (explained later), and 8 bits for time-to-live (TTL). The short label provides the advantage of high-speed hardware implementation of the lookup table. At each intermediate node the incoming label is used to directly index the lookup table to obtain the output port and the label for the next hop.

In addition to fast service, MPLS provides the advantage of scalability by reducing the number of labels and consequently the size of the table. Therefore the network can be scaled to support the growing number of users. Another feature of MPLS is the possibility

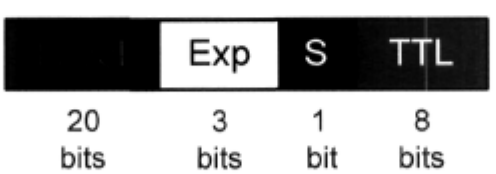

Figure 2.2. MPLS label format; Exp: experimental functions, S: stack, TTL: time-to-live.

of efficient traffic engineering, *i.e.*, the allocation of network traffic according to the available resources. In this context, the incoming packets are classified according to characteristics such as the same destination or the same traffic type (voice, data, etc). At the ingress node each packet is associated to a particular level, called a forwarding equivalence class (FEC), based on which the label is assigned to the packet. The intermediate node only uses the attached label to perform the forwarding process.

In MPLS networks, the two main responsibilities of a node, *i.e.*, routing and forwarding, are carried out separately. These procedures can be therefore handled independently by different technologies. For example, routing can employ the IP protocol, while forwarding can exploit ATM technology. This highlights the main advantage of MPLS over ATM: ATM needs its own protocols for routing and signaling; MPLS is better matched to IP networks. Another advantage of MPLS is that unlike ATM that uses fixed-length cells, MPLS carries variable length packets. Therefore, MPLS provides improved performance by avoiding segmentation and reassembly functions, and reducing overhead.

2.2.1. Hierarchical Addressing and Label Stacking in MPLS

Hierarchical addressing is supported by MPLS by assigning more than one label to the packet [20,25]. Large data networks make use of hierarchical addressing to reduce the size of lookup tables. In this context, the nodes (routers) that are near each other are grouped in a common region. Each node has detailed information of the nodes in its local region and is unaware of the details of other regions. By assigning a prefix address to each network region, routers only need to check a part of the address to decide whether the packet should be sent out or be routed inside the region, resulting in reduced table size. To see this more clearly, consider a network with 20 nodes. If the network is partitioned into 4 regions each containing 5 nodes, then the lookup table of each node will need 5 local plus 3 remote entries. Without hierarchical addressing 20 entities are required for the table (versus 8 entities when using hierarchical addressing).

Two-level hierarchical addressing is employed in IP networks by separating the IP address into two fields, *i.e.*, network ID and host ID. Likewise, the two-level hierarchy in ATM networks is specified by virtual path identifier (VPI) and virtual channel identifier (VCI)

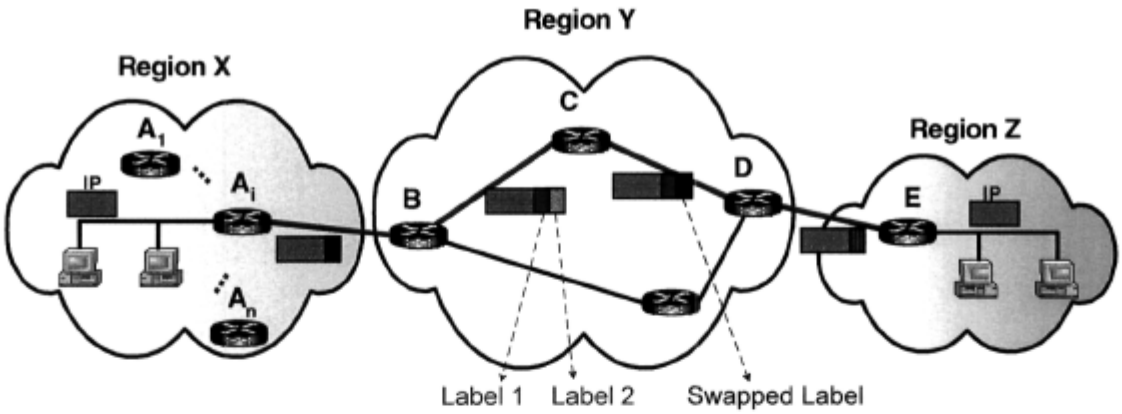

Figure 2.3. The idea of label stacking in MPLS networks.

fields of the ATM header. For a large network more levels would be required to provide improved performance. More precisely, the optimum number of levels for a network with R routers is $\ln(R)$ [26]. In MPLS an arbitrary number of levels can be used by stacking several labels related to different network domains. The label at the bottom of the stack is used for routing and is called the level 1 label. All the stacked labels have the S bit (see Figure 2.2) equal to zero except the label that is at the bottom of stack that has S equal to one. As an example, a stack of 2 labels is illustrated in Figure 2.3. Suppose that an IP packet arrives at node A_i (belonging to region X) and is destined to node E (belonging to region Z). At node A_i a level 1 label is assigned to the packet. When the packet reaches the edge node of region Y, node B pushes the label into the second level of stack and adds a new level 1 label. From node B to D the level 1 label is used for forwarding and at node D this label is removed and the remaining higher level label is considered for routing in domain Z. As router E is at the edge, the final destination address is processed and the IP packet is delivered to IP address of the PC. Note that when using hierarchical addressing, only one path is established between node B and D. Without this mechanism, n different paths are required to connect all the n nodes at region X to node E.

2.3. Optical MPLS

MPLS has been extended to introduce a new network structure called generalized MPLS (GMPLS). In GMPLS the label can be anything suitable to identity the traffic, such as

position in time, wavelength, or physical port. The routing and signaling protocols are generalized to support time-division multiplexed (TDM) circuits, light paths, and fiber connections. We use the term MPλS where forwarding information is indicated by the wavelength of traffic flow. The wavelength paths are established by appropriate setting of the optical cross-connects at the intermediate nodes. Although MPλS takes advantage of the high transmission capacity of fiber and resolves the bottleneck of electronic routers by photonic switching, it faces some challenges. Firstly, the number of users that can be supported is limited by the number of available wavelengths. Furthermore, the minimum granularity is an entire wavelength channel, resulting in poor spectral efficiency.

To overcome these issues, novel schemes have been demonstrated to achieve optical label switching on a packet-by-packet basis. In these schemes, at the edge node the incoming electrical packet is converted to an optical packet. Based on the information of the address field of the IP packet, an optical label containing the routing information is added to the IP packet (the entire electrical packet becomes the payload of the optical packet). The packet is switched all-optically, independent of the original format or bit rate of the electrical packet. The main responsibilities of the intermediate nodes, *i.e.*, label recognition and label swapping, are discussed in the following section.

2.3.1. Label Recognition and Swapping in Optical Networks

The processing required at the GMPLS intermediate nodes includes routing and forwarding. For optical networks, we assume that routing requiring high-level functionalities and complex processing is performed by an electronic overlay network. Forwarding, on the other hand, requires relatively simple label processing including label recognition, switch setting, and label swapping. Ultra-fast forwarding is desirable so that the packets can pass through nodes on-the-fly.

Label processing can be accomplished either electronically or optically. In the electronic approach, the received optical label is extracted and converted to an electrical signal. This signal is used to access the table lookup entities containing the output port and the new label. A control signal is generated to set the state of the switch and also to swap the optical label. The new label is combined with the payload to form the complete packet that will be

switched to the appropriate output path. The payload remains optical during this process; therefore, it is immune to the impairments caused by OEO conversion.

At high bit rates (gigabits per second), the packet duration is so short that electronic circuits have difficulty handling label processing before the next packet arrival. Among the forwarding procedures, table lookup is the most time-consuming function and all-optical processing is the only means to reduce this burden. The lack of optical components, such as logic circuits and buffers, necessitates special techniques and architectures capable of performing label processing by rudimentary but fast optical devices. Most of the reported work relies on optical correlation for label recognition [6,8,13], where several parallel correlators are used to check the incoming label (see chapter 4). The autocorrelation peak of the correlator matched to the incoming label will open a gate connected to the appropriate output port. This parallel processing approach is limited by high splitting loss and large number of required components. Another approach is to use a self-routing multistage switch, such as a tree switch [12], constructed of several switching elements (see chapter 5). Each bit of the label is used to control one stage of the switch to automatically direct the packet to the proper output port. The technology of switching elements (electronically controlled or optically controlled) determines if the optical label bits can be directly used as the control signal or if they must first be converted to an electrical signal [9,27-29]. The difference between the two approaches is in the time required for opto-electronic conversion and also the time required for the switch to change its state.

The other responsibility of the node, *i.e.*, label swapping, can be accomplished by either regenerating a new label or converting the old label to a new one. In both cases, the label information should be extracted by consulting the routing table to determine the new label. Label conversion may have the advantage of a reduced number of optical components to realize the label swapper module. For example, if the labels are comprised of several wavelength channels, wavelength conversion will alleviate the necessity of several lasers to generate the new label. Label swapping options are not treated in this thesis, although a very brief review of some optical label swapping techniques is provided in section 3.7 for context. We only review some optical label swapping techniques proposed in the literature in section 3.7. In appendix A, we report an experiment where a label swapper proposed by

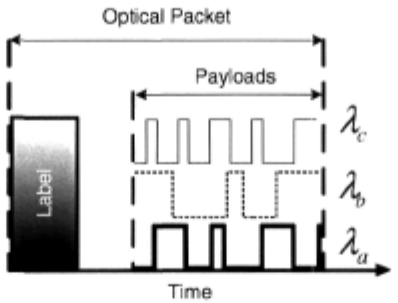

Figure 2.4. Transmission of payloads with the same destination address in a same optical packet in different wavelengths.

researchers from McGill University is integrated into our binary multi-wavelength node structure.

2.3.2. Optical Packet Format

The optical packet consists of two parts: label and payload. Several methods have been proposed in the literature for the label, including: on-off keying modulation, subcarrier multiplexed labels, optical CDMA labels, and multi-wavelength labels. A review of these methods is presented in chapter 3.

The optical packet may contain more than one payload. The IP packets having the same destination and the same characteristics (*e.g.* delay tolerance, quality of service, etc) are aggregated at the edge node and sent as an ensemble burst with a common label. In optical networks, it is also possible to send the packets with the same destination simultaneously on different wavelengths [9,15]. In the example of Figure 2.4, the three payloads are time-overlapped, occupy distinct wavelengths, and share a single label.

The optical packet can be of fixed- or variable-length. As the only currently available optical buffers are fiber delay lines, it is easier to implement networks with fixed-length packets. In this case, the length of the delay element is proportional to a multiple of packet length. However, a fixed-length optical packet is not well matched to variable-length IP traffic and may cause inefficiency in network performance. Variable-length optical packets are better adapted to the bursty nature of IP traffic. It has also been shown in [30] that with variable-length packets, contention is alleviated and the network packet-loss rate decreases. In [31], a combination of multiple stages of fiber delay lines is proposed to realize variable

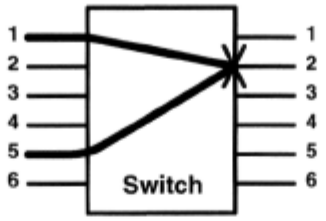

Figure 2.5. Contention occurs when the incoming packets from input 1 and 5 are destined to output 2 at the same time.

optical buffers, required for variable-length packets. Both fixed- and variable-length packets are currently considered for optical networks.

Optical packets can be classified as synchronous or asynchronous [30,31]. In asynchronous networks, the packets are switched on-the-fly; thus, the probability of contention is higher. However, the complexity and cost of asynchronous networks are lower; synchronizing in the optical domain is reported to require more sophistication than that of electronics [32].

2.3.3. Contention Resolution

Another functionality required for the GMPLS node is contention resolution. In a packet switching network, contention occurs when two or more incoming packets are destined to the same switch output at the same time. Figure 2.5 shows a conflict between packets from inputs 1 and 5 arriving at the same time and both destined for output 2. In electrical networks, contention is typically resolved by buffering, *i.e.*, one of the packets is delayed until the other one is delivered.

In optical networks, several approaches are proposed to resolve contention including: buffering, wavelength conversion, and deflection routing. Buffering using optical delay lines was demonstrated in [33], imitating random access memories (RAM) in electrical networks. Unlike RAMs, fiber delay lines are not variable memories and therefore this approach is only possible for fixed-length packets. Due to the sequential access to delay lines, long fibers are required to achieve a reasonable packet loss probability, working against system compactness. To reduce the required number of fiber delay-lines or even to resolve contention without any buffering, wavelength conversion was examined in [34]. In this scenario the packets that are destined to the same output at the same time are sent on

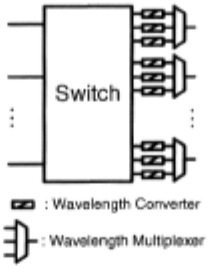

Figure 2.6. Wavelength conversion for contention resolution.

different wavelengths. As shown in Figure 2.6, when the contending packets are detected in the switch, their wavelength is first changed by the wavelength converters and then they are multiplexed to the desired output fiber. Another solution for contention is deflection routing in the space dimension as presented in [35]. When contention occurs, one of the contending packets is sent to the desired output; the remaining contending packets are sent to other outputs to be routed by less preferred paths.

The combination of the three contention resolution schemes, *i.e.*, using wavelength, time, and space, was investigated in [36]. The proposed router first tries to convert the wavelength of the contending packets to the free wavelengths. If that fails, the time-domain and space-domain solutions are examined. If all three methods fail, the packet will be discarded. It has been shown that good network performance in terms of throughput and packet loss can be achieved by the combined scheme.

Data grooming at edge routers was also examined to enhance the contention resolution process [30]. The self-similar nature of Internet traffic degrades the packet-loss performance of packet switched networks [37], due to the high number of contending packets at the switching nodes. If the packets could be transmitted at a more regular rate, congestion would be reduced. To this end, traffic shaping at the electrical ingress nodes has been suggested in [30]. Packets with common features, such as the same destination, are aggregated in a buffer to form a large optical packet. The optical packet is generated when the size of the assembled packets reaches a parameter called the maximum packet size. Simulations show that the combination of traffic shaping and contention resolution at core routers in wavelength, time, and space domain improves the packet-loss performance of network considerably.

A switch architecture, data vortex, is proposed in [38], where contention is resolved internally (in the switch) by deflection routing. The fabric is composed of 2×2 switching elements equipped with control logic. The interconnections topology and the signaling between switching elements avoids packet conflict. When contention occurs (for an internal or external port), one packet is routed to the correct path, while the other is routed along a different path and remains in the switch to be delivered to the desired output port later. The data vortex employs synchronous timing and fixed-length packets. It can approach throughput as high as ~100% at the cost of increased component count. The detailed discussion about the vortex architecture and its performance is provided in section 3.6.2.

In chapter 4 and 5 of this thesis, we present two novel strategies for packet forwarding. In these architectures, we did not consider contention resolution and assumed only one of the contending packets are delivered and the rest are lost. Contention resolution with wavelength conversion, deflection routing or limited optical buffer can be added to the proposed forwarding modules; contention resolution remains an open research problem. In chapter 6, we tackle the contention resolution problem for a simplified network topology [24]. We examine a randomized scheduling algorithm that reduces the probability of conflict and therefore improves the packet loss performance. The details of this algorithm and the simulation results are provided in chapter 6.

Chapter 3

Review of Optical Label Embodiments

3.1. Introduction

Various optical label encoding techniques other than MPλS have been proposed in the literature. The main objective of all these schemes is to perform high-speed optical packet forwarding. Examples of these methods include time domain encoding using on-off keying (OOK) modulation [2,3], subcarrier multiplexed labels [4,5], optical CDMA labels [6-8], and multi-wavelength (M-λ) labels [9-15,39-43]. Depending on the label format, some processing (mainly label recognition) can be accomplished in the optical domain. However, for approaches such as subcarrier multiplexed labels, opto-electronic conversion of the label is inevitable. In the following sections, we will review prior work on optical label

switching networks. Furthermore, we will address some label conversion techniques for label swapping.

3.2. Time Domain Label Encoding

In this scenario, the label is created via OOK modulation, a modulation distinct from that of the payload--both for modulation rate and format. The baseband label directly modulates an optical source to create a pattern of ones and zeros in the time domain. Given B label bits, 2^B labels are available. The payload follows the label in time, typically after a specified guard time to facilitate label detection. One way to recognize the label is to convert the optical label to an electrical signal and perform the label processing in the electrical domain. This approach is mainly limited by the slow speed of electronics, a significant bottleneck for ultra-high-speed fiber transmission.

Recently, some effort has been made toward the implementation of all-optical label recognition for time domain labels. For instance, a tunable, multi-wavelength optical correlator is demonstrated in [2], where the incoming label is split 2^B times to find a match in one of 2^B parallel correlators. Arrays of fiber Bragg gratings (FBG) are used for correlation. The gratings are positioned such that the round trip delay serves to align the label bits in time and provides an autocorrelation peak. As shown in Figure 3.1, the reflectivity of consecutive gratings increases along the array. Different reflectivities are achieved by detuning the gratings by stretching or by heating. This imposes a practical

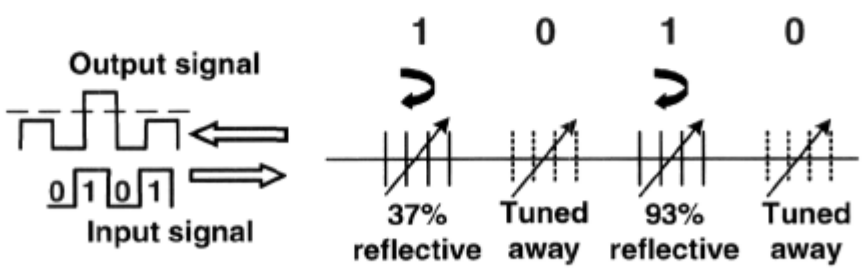

Figure 3.1. Optical correlator structure proposed in [2] to detect the time domain encoded label.

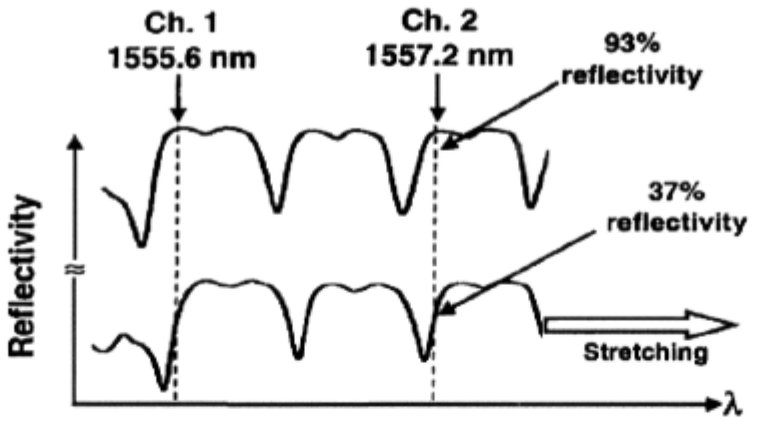

Figure 3.2. The periodic spectrum of the gratings used in the structure of correlators proposed in [2].

limitation on implementing large number of labels, as very fine resolution in the reflectivity of gratings is difficult to achieve.

The gratings are written to have periodic spectrum (Figure 3.2) so that they can operate at multiple wavelength channels simultaneously. Thereby, the correlation can be performed in a same device for packet headers on different wavelength channels, resulting in a reduced number of required components.

The experimental setup used in [2] is shown in Figure 3.3. This setup is designed to detect 10 Gb/s labels at two wavelengths. Ten percent of the incoming power is tapped off and sent to two arrays of FBGs to check for "1" and "0" bits in the header. In the "1" ("0") bits correlator, the FBGs that are related to "1" ("0") bit positions are tuned in band and the rest are tuned out of band. For example, for header "0101", the first and the third FBGs of the "1" bits correlator are tuned in band; the second and the fourth FBGs of the "0" bits correlator are also tuned in band. The reflected signal from the FBGs is wavelength demultiplexed to be fed to separate decision circuits related to different wavelengths to generate the switch control signal. A 2×2 lithium-niobate electro-optic switch is used as the switching element.

Another approach to optically recognize the time domain labels was presented in [3]. This scheme is also based on correlation detection, that is, the labels are split over N parallel

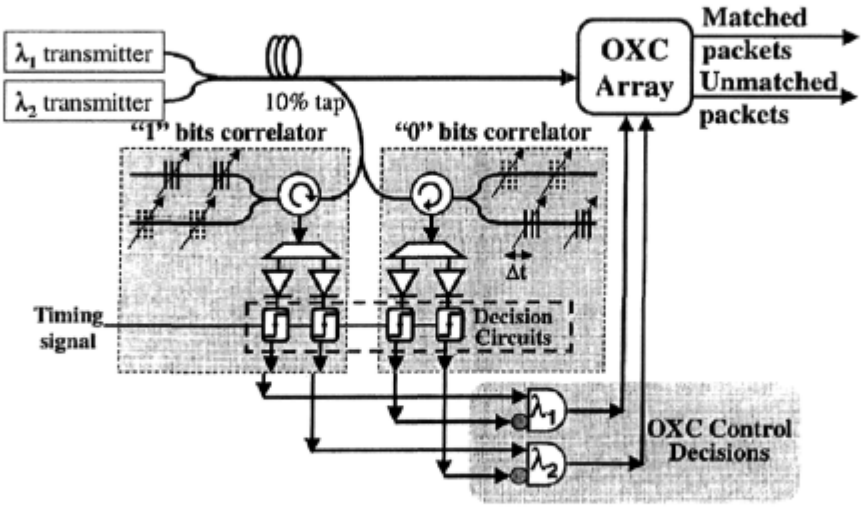

Figure 3.3. The experimental setup of multi-wavelength correlator used for time domain label recognition [2]. OXC: optical cross connect.

detectors to recognize one among N possible labels. Upon receiving the label at the label recognition module, the label bits are colored with distinct wavelengths; B wavelength are required for detecting B-bit labels. Each decoder consists of FBG arrays which are tuned/detuned according to the stored label. Tuning the label recognition module is possible by tuning the FBGs.

The schematic diagram of this method is illustrated in Figure 3.4. As an example, consider the receipt of a label with 3 bits as was achieved with the experimental setup of [3]. The label bits are converted to wavelengths λ_1, λ_2, and λ_3 respectively. The wavelength conversion is accomplished via cross gain modulation (XGM) in a semiconductor optical amplifier (SOA). The output of three lasers at λ_1, λ_2, and λ_3 are multiplexed and are time gated with the duration equal to the label bits using a modulator. Dispersion in a single mode fiber (SMF) will separate the wavelengths in time. The length of the SMF is chosen so that at the end of the fiber the wavelengths arrive at contiguous label bit intervals. This signal and the incoming label are sent to the SOA simultaneously. Because of XGM in the SOA, the output signal will be the logical compliment of the header bits with converted

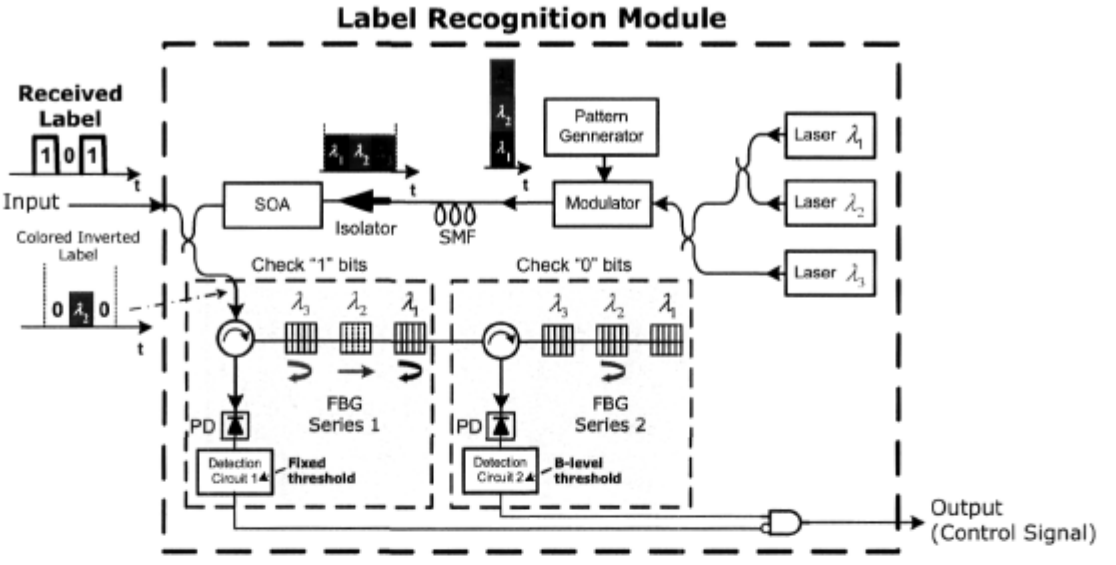

Figure 3.4. Schematic diagram of optical label recognition of time domain labels using time-to-wavelength mapping. *B*: number of label bits, FBG: fiber Bragg grating, PD: photodiode, SMF: single mode fiber, SOA: semiconductor optical amplifier.

wavelength. For instance when header "101" is received, "$0\,\lambda_2\,0$" will appear at the output of the SOA (see Figure 3.4).

The main drawback of the previously mentioned approaches is high splitting loss, since as many correlators as the number of labels must be used to check the incoming label to find a match to one of the stored patterns. Therefore, the incoming label should be spit 2^B times in the case of B label bits. Furthermore, the format of the control signal, *i.e.*, the autocorrelation peak, restricts the switching fabric to a simple gate. Therefore, to connect one input port to N outputs, N gates are required; consequently, to realize an $N \times N$ switch, N^2 gates are required. Furthermore, the label bit rate is fixed, determined by the round trip delay between FBGs. In the second approach, synchronization between the probe signal and the incoming label is difficult to achieve.

3.3. Subcarrier Multiplexed Labels

Subcarrier multiplexing (SCM) is another technique that is used to encode the label. In this approach the label and payload occupy the same waveband, but orthogonal frequencies.

The label bit stream is modulated onto a radio frequency subcarrier first, and then is added to the payload baseband signal. The total signal modulates the instantaneous power of an optical source. The spectrum of an optical packet reported in [4] is shown in Figure 3.5. The peak at the center corresponds to the payload and the two sidebands correspond to the label spectrum. In this experiment a 14 GHz subcarrier was used to encode the label. Therefore, the spectral distance between the baseband signal of Figure 3.5 and each sideband tone is 14 GHz.

As label and payload have separate frequencies, they can be transmitted simultaneously. The out-of-band label can be extracted using passive optical devices. In [4], a FBG with 10 GHz passband at the packet center frequency is used to extract the label. The label is then sent to a router controller to be processed by electronic circuits. As the whole packet is at one waveband, switching the wavelength switches packet and label. In [4] a combination of wavelength converters and arrayed waveguide grating (AWG) was used as the switching fabric.

The main drawback of subcarrier labels is electrical label processing, thus the speed is limited by the speed of the electronic circuits. Furthermore, active microwave components (such as mixers) are required for this method, increasing complexity and cost. In addition, the subcarrier frequency limits the bit rate of the payload, as the bit rate must not exceed the spectral distance between payload and label. The closely spaced payload and label can lead to crosstalk due to fiber dispersion and nonlinearities.

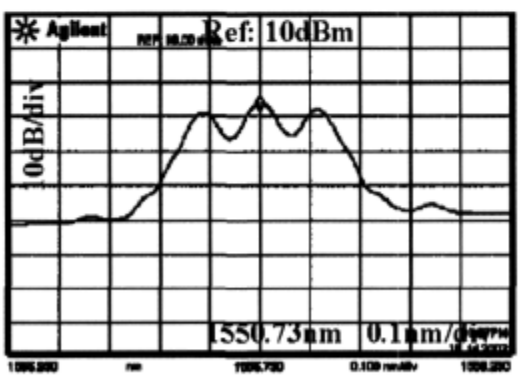

Figure 3.5. The spectrum of an optical packet with subcarrier label encoding [4].

3.4. Optical CDMA Labels

OCDMA was proposed for multiple access networks, in which multiple users share the transmission media. Such a network typicaly has a star topology such as that illustrated in Figure 3.6. Similar to any other multiple access network that is based on spread spectrum techniques, in OCDMA networks a code is assigned to each user as its identification; each data bit is encoded with this code before transmission. Therefore, all users can transmit simultaneously, yet only the receiver employing the correct code can decode the data.

OCDMA has been considered to carry the routing information of a hop-by-hop label switching network [6-8]. The motivation for OCDMA labels is to bypass the main bottleneck of electrical routers, *i.e.*, table lookup, instead employing all-optical label recognition using a set of parallel correlators. It should be noted that unlike conventional OCDMA networks, in a hop-by-hop network, packets with the same wavelength are never transmitted simultaneously; rather they are sent in non-overlapping time intervals. Therefore, interference from other users does not exist and the OCDMA codes are only used for their compatibility with optical label processing.

In this scheme, an OCDMA code is assigned to each packet as its label. As many correlators as the number of labels are used in parallel at the switch to identify the label. The incoming label is split and distributed to each correlator to find a match. If the label matches one of the correlators, the autocorrelation output will reach the threshold level and a control signal will be generated to open a gate to forward the packet to the appropriate

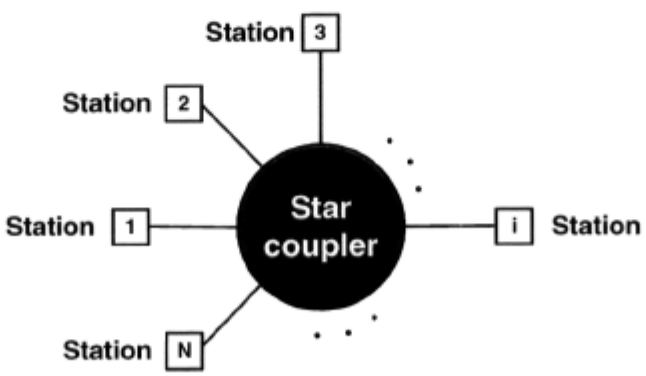

Figure 3.6. Star topology used for multiple access network.

output. Various OCDMA techniques have been examined for label switching networks [6-8]; we will briefly review some of them in the following sections.

3.4.1. Time-Wavelength Codes

OCDMA packet routing network using two-dimensional codes (time and wavelength) was suggested in [8]. The labels are T chips long where one of L wavelength bins is transmitted in each chip. The total number of available labels is $L!/(L-T)!$. The payload and label occupy different wavelength bands. The packet consists of a header (label), a payload, and a trailer, multiplexed in time. The header and the trailer are the same codes and indicate the beginning and the end of a packet. The label generator and decoder are constructed by FBGs. Label decoding is based on correlation, *i.e.*, parallel correlators are used to generate an autocorrelation peak to control switching gates. The header and the trailer are sent to the correlators; the autocorrelation signal from the header (trailer) opens (closes) a switching gate. Therefore, this system supports variable-length packets. Error-free three-hop routing of variable-length packets containing 40 Gbps data was reported in [8]. Similar to other label recognition schemes based on correlation, this method suffers from high splitting loss and also large number of components such as correlators and switching elements.

3.4.2. BPSK Codes

In [6] OCDMA based on binary phase shift keying (BPSK) is examined as labels. Several short pulses (chips) are sent in the label interval, the phase of which takes the value of 0 or π (see Figure 3.7). The optical encoder is constructed by tapped delay-line waveguides with thermo-controlled optical phase shifters; all integrated using planar lightwave circuit (PLC) technology. The encoder is tunable by adjusting the optical phase shifters. The encoder generates 8 chip optical labels with 5 ps chip interval. A mode locked laser (MLL) with full width at half maximum of 1.5 ps is used as the optical source. Error free routing of 64 bit payloads at 10 Gbps is reported. In this method, the chip rate is fixed and corresponds to the differential delay between waveguides. Moreover, costly components such as MLLs are required.

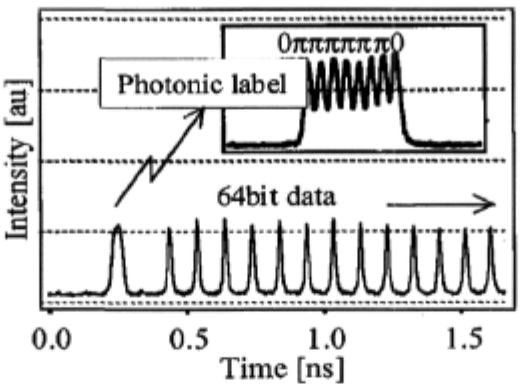

Figure 3.7. BPSK-OCDMA label used in [6].

In the BPSK scheme, the same codes can be used simultaneously in different wavelengths similar to a network with several independent layers in different wavelengths. As shown in Figure 3.8, at each node, the signal of the incoming fibers is first wavelength demultiplexed. The forwarding and switching operation of all the wavelengths is performed in a module called photonic label switching router (PLSR). A cascade of correlators is required for each of the inputs of PLSRs. Therefore, having L wavelengths and N labels requires $L \times N$ correlators in each PLSR. It is clear that the number of elements is quite large in this structure.

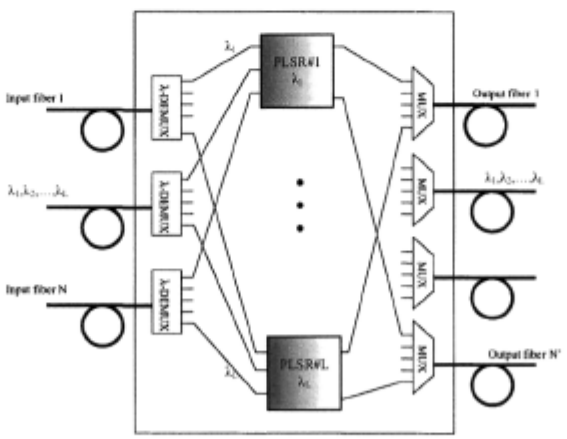

Figure 3.8. The node structure of the BPSK-OCDMA label switching network proposed in [6].

3.4.3. Compact Encoder/Decoder for PSK Codes

The main limitations of OCDMA-based packet switching techniques are splitting losses and the large number of encoders and decoders required for label generation and processing. A compact PLC device is designed in [7] to generate and process a set of phase-shift keying (PSK) optical codes. The device has N inputs and N outputs; if a short pulse is fed to one of the inputs, N PSK codes of length N symbols are generated simultaneously at the output ports. When an encoded pulse is fed to one of the inputs, the correlation of the input signal and the entire code set appears at the output; each port gives correlation with one code in the set. The output port where the autocorrelation peak occurs identifies the code. The autocorrelation peak is detected and drives an electronic circuit generating the control signal that sets the state of the switch. The switching gates at the packet routing node are used to block or forward the packet depending on the control signal received. Therefore, N^2 photodiodes and switching elements are required to realize an $N \times N$ node.

The schematic of the device based on arrayed waveguide gratings is shown in Figure 3.9 [44]. The device is composed of two slab couplers at the input and output and an AWG. The length between the adjacent waveguides varies by a constant ΔL. If a short pulse is fed into a device input, N copies of the pulse are generated by the input slab coupler, with phases given by Rowland circle configuration [7]. The pulses travel different lengths in the grating where the differential path delay is chosen larger than the input pulse width so that

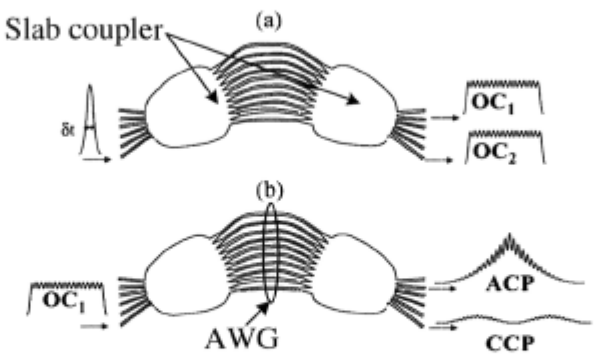

Figure 3.9. (a) Compact encoder and (b) decoder for PSK codes in AWG configuration; APC: autocorrelation peak, CCP: cross-correlation peak, OC: optical code [44].

the chips in the optical code do not interfere. The pulses are recombined by the output slab to build N PSK codes at the output ports. The design guidelines and the theoretical model of the device are presented in [7].

The encoder/decoder was fabricated and experimentally demonstrated to generate and process 16 PSK codes with 16 chips at 200 Gchip/s [44]. The pulses are 2.5 ps full-width at half-maximum (FWHM) width, generated by a mode locked laser diode at 10 GHz. In an optical packet switching experiment, optical packets containing 10 Gbps payload and three different labels were generated as shown in Figure 3.10. The packets were successfully routed to three outputs, using the experimental setup of Figure 3.11. Bit error rate (BER) less than 10^{-12} were measured in back to back and also 50 km transmission configuration.

The encoder/decoder can be used in "multi-dimensional" mode to increase the code cardinality without increasing the code length [45]. In this mode, two or more simultaneous laser pulses are driven to different device inputs to generate a unique set of N orthogonal codes. As different combinations of the inputs generate a different set of labels, n_l input pulses generate $\binom{N}{n_l}$ orthogonal codes. In this case, when the device is used as a decoder, autocorrelation peaks are generated at n_l output ports simultaneously. Label recognition is performed by detecting the set of autocorrelation peaks (*i.e.* output ports) corresponding to a particular code. The multidimensional operation of the device is examined in [45] where two short pulses at different wavelengths are sent to the device to generate codes. Two autocorrelation peaks appear at the output when a code is sent to the device, *i.e.*, when used as a decoder. Simulation shows that the packet-loss probability in the multi-dimensional

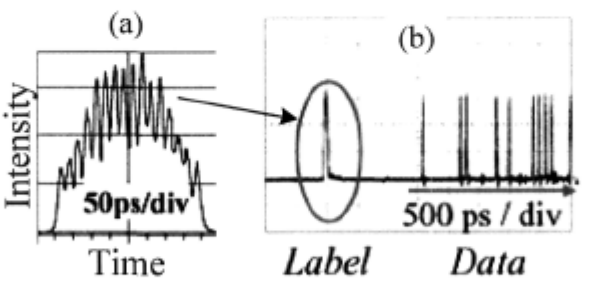

Figure 3.10. (a) PSK optical label at the encoder output, (b) optical packet with PSK optical label [44].

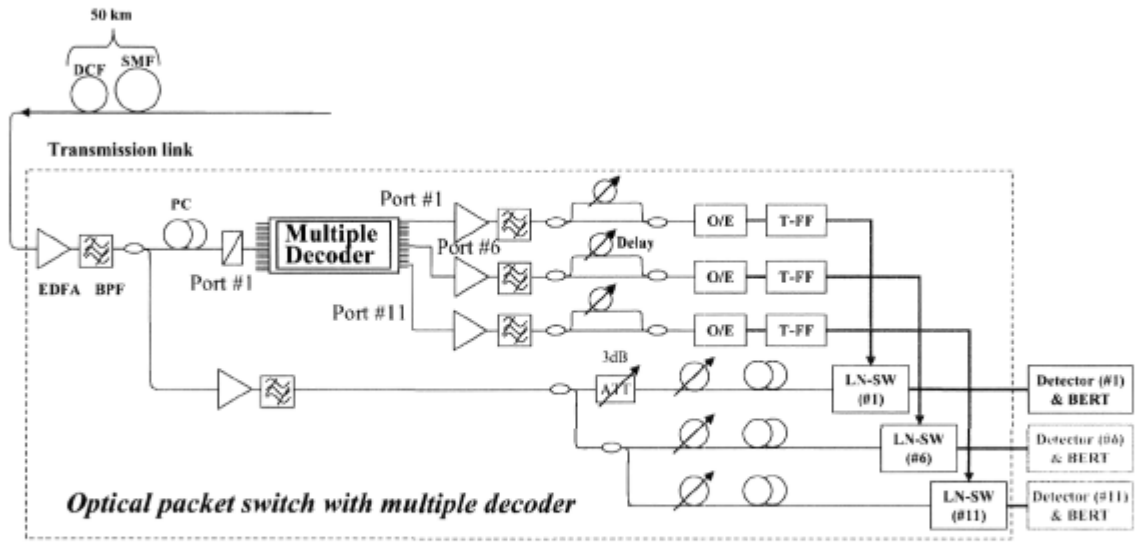

Figure 3.11. Experimental setup of PSK label switching using compact encoder/decoder; Att: attenuator, BERT: bit error rate tester, BPF: band pass filter, DCF: dispersion compensating fiber, LN-SW: $LiNbO_3$ optical gate switch, O/E: optical to electrical converter, SMF: single mode fiber, T-FF: T-flip-flop [44].

case is lower than the one dimensional.

Although the compact encoder/decoder eliminates the large number of parallel correlators and can generate an impressive number of labels, it is based on costly and complex ultra-short pulse technology. Another shortcoming of this scheme is that the differential length of the AWG waveguides effectively fixes the label chip rate.

3.5. Multi-Wavelength Labels

The label of an optical packet can be composed of several wavelength channels [9-15,39-43]. Compared with the time domain technique, the number of available labels can be increased without increasing the label duration or bit rate. At the cost of spectral efficiency, the low bit rate label requires only low speed electronics when label processing is performed electrically.

If the waveband of the label and payload are orthogonal (out-of-band-signaling), the label can be transmitted simultaneously with the payload (see Figure 3.12 (a)). This results in higher throughput, as the label does not extend the packet duration beyond that of the

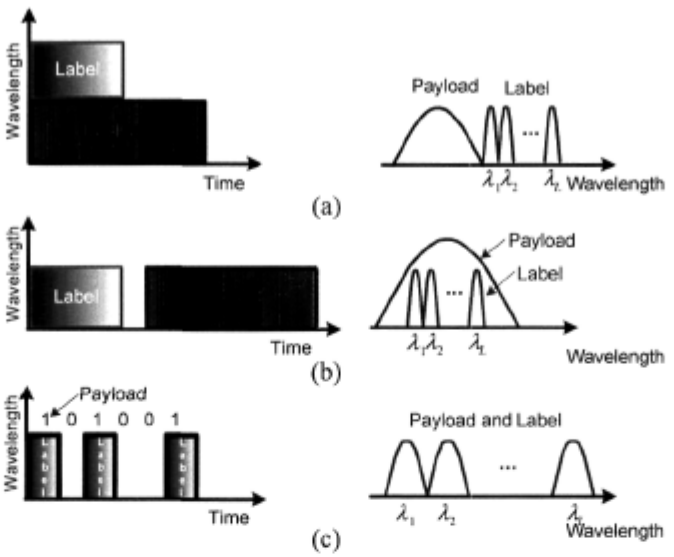

Figure 3.12. (a) Optical packet with out-of-band signaling. Optical packet with in-band-signaling, where (b) label and payload are time-multiplexed, or (c) the payload is encoded bit-by-bit and the label is implicit in the payload bits.

payload. Label extraction can be easily accomplished using passive optical devices. The processing and regeneration of the label is accomplished without modifying the payload. The payload is switched without being processed; the system is transparent to the bit rate and the format of payload.

When the label and payload occupy the same spectral band, it is called in-band-signaling. In this method, the label precedes the payload (see Figure 3.12 (b)) with a guard time between payload and label; synchronization between payload and label is required. As both parts of the optical packet are in the same waveband, the degradation that occurs during the transmission is the same for label and payload.

In in-band-signaling, label and payload can also time overlap as illustrated in Figure 3.12 (c) [13]. In this scheme, the payload bits are encoded, that is, they carry the label addressing information. The label bins encoded by the addressing information are modulated with high-rate payload data. No additional waveband or light source is required for the payload.

The wavelength bins reserved for the labels can be encoded in different ways to carry the addressing information. In the following sections we will discuss three encoding technique for multi-wavelength (M-λ) labels: spectral amplitude codes (SAC), binary M-λ labels, and weight-2 labels.

Firstly, codes with high autocorrelation and low cross-correlation properties, such as SAC, can be used as labels [13]. The number of available labels is equal to the code cardinality. Similar to other OCDMA techniques, high-speed correlation-based label recognition can be exploited. This approach requires parallel correlators, as the approaches in 3.4.1 and 3.4.2, however, SAC has advantages in terms of complexity and stability. Also, the semi-orthogonal property of the codes makes them compatible with hierarchical addressing that can be exploited to significantly reduce the size of lookup tables (*i.e.*, the number of parallel correlators) and therefore the splitting loss. For hierarchical addressing, two or more codes/labels are transmitted simultaneously, representing a stack of labels.

The second approach for encoding the multi-wavelength labels is binary addressing (binary M-λ). The label bits are mapped to distinct wavelength bins where the 0/1 bits turn off/on the corresponding bin. These labels have no orthogonality properties and maximum cardinality, *i.e.*, 2^L labels are available with L bins [9,10,12,15].

In [14,46] another encoding technique is proposed where only two wavelengths among the label band represent the label (weight-2 labels). Label recognition is achieved by generating the four wave mixing (FWM) product of the label wavelengths by using a nonlinear device, and then selective filtering. We will discuss the wavelength encoding techniques in the following sections.

3.5.1. Spectral Amplitude Codes

Codes with high autocorrelation and low cross-correlation properties, such as SAC, can be constructed with the wavelength bins dedicated for the labels [13]. SAC, typically used for OCDMA networks, is based on the encoding of incoherent broadband sources (BBS) in the frequency domain. The number of available codes/labels is equal to the cardinality of the code family. In the optical packet switching scheme, the labels occupy one band and the

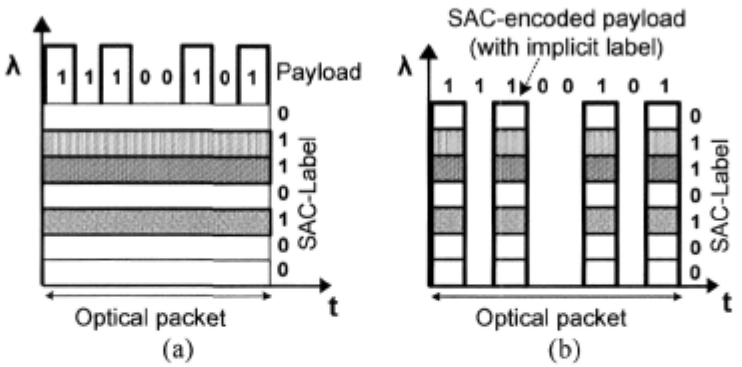

Figure 3.13. (a) Optical packet with separable SAC labels. (b) Optical packet with SAC-encoded payload.

payload is out-of-band, a scheme we refer to as separable SAC-labels. Labels are transmitted simultaneously with the payload and there is no added time overhead for the label (Figure 3.13 (a)). Label recognition can exploit ultra fast correlation-based processing.

We can also use the packet format consistent with Figure 3.12 (c), where the payload bits are SAC-encoded and the label is implicit in the payload (Figure 3.13 (b)); in this case the SAC labels are not separable. This second approach requires less components and is more spectral efficient, but the bit rate is limited due to intensity noise of the BBS. Packet routing can be performed either bit-by-bit requiring high-speed switches, or using an envelope detector generating a control signal with the same time window as the packet to set the switch (see section 5.2.2 for details).

In both methods, the label processing is achieved via parallel correlators, thus high splitting loss. We have mitigated this problem by hierarchical addressing to reduce the size of lookup tables (parallel correlators) and the splitting loss. We take advantage of the semi-orthogonal property of SAC to transmit two or more codes simultaneously, representing a stack of labels. The detailed explanation of this approach and the experimental proof of the concept is provided in chapter 4.

3.5.2. Binary Multi-Wavelength Labels

The second approach for encoding the M-λ labels is binary addressing. The label bits are mapped to distinct wavelength bins where the 0/1 bits turn off/on the corresponding bin.

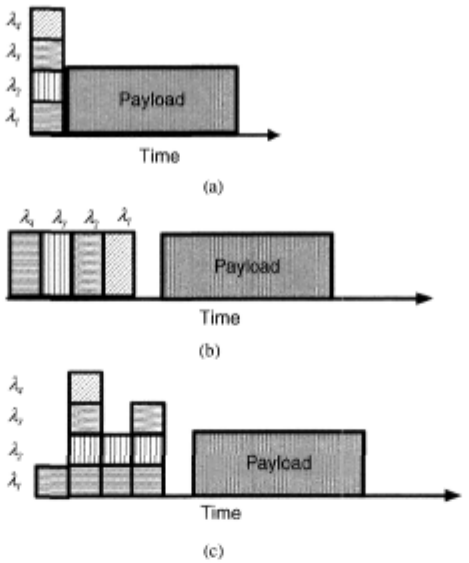

Figure 3.14. Binary M-λ label schemes (a) bit parallel (b) bit serial (c) parallel-serial.

These labels have no orthogonality properties and 2^L labels are available with L bins. Three different schemes are illustrated in Figure 3.14, namely bit parallel, bit serial, and parallel-serial. In the bit parallel scheme of Figure 3.14 (a), the wavelengths are sent simultaneously during the label interval. For instance, in the packet format used in [8,12,15] the label is transmitted during the entire packet duration as shown in Figure 3.15. Low speed electronics can be used to recognize and process the label because of the low transmission rate; the label information transfer rate is still high.

Parallel label bits also facilitate generation of the control signal for the switch. Multi-stage architectures can be used to create self-routed packet switched nodes [12] where each wavelength channel can control a stage of a multi-stage optical switch. A wavelength filter followed by a photodiode can generate an electronic control signal. Wavelength sensitive switching elements have been proposed in [9] to avoid the use of photodiodes and instead to control the switch all optically (see section 3.6.1). Even if higher-level label processing (such as a routing algorithm) is to be accomplished electronically at each node, it will be much simpler to have parallel bits. A detailed study of this method will be presented in chapter 5.

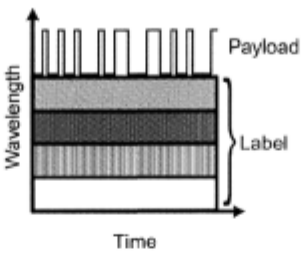

Figure 3.15. Packet format based on bit-parallel binary M-λ labels when the label is modulated at the packet rate [8,12,15].

The second possibility for composing the binary M-λ labels is bit-serial as illustrated in Figure 3.14 (b). In bit serial signaling a combination of time and wavelength forms the label [10,40-43]. One wavelength is sent in each timeslot, therefore, having T timeslots and L wavelengths L^T different labels are available. Compared with a bit parallel scheme, this method needs a lower number of wavelengths to produce equal number of labels, however, it needs higher speed processing because of the higher bit rate of the label. Furthermore, as the information encoded in the time domain must also be extracted, a more complex receiver structure (like serial to parallel converters at the input of the label processor) is required. All optical label processing has been demonstrated in [43] for the recognition of bit serial M-λ labels. Arrays of FBGs are used to form correlators to find a match between the incoming label and the stored patterns. The label must split over all the correlators, causing high splitting loss.

It is also possible to send more than one wavelength in each timeslot of the header as illustrated in Figure 3.14 (c) [43]. In this case the number of available labels increases to 2^{LT}, given L wavelengths and T timeslots. Each timeslot can carry the information that is related to a network domain, similar to the IP address. Therefore, at each network domain only the corresponding timeslot is detected to determine the routing decision. This method takes advantage of parallel bits, and yet it is not as restricted by the number of available wavelengths when a large number of labels is required.

In the rest of this thesis, we will use the term "binary M-λ" to refer to the bit-parallel case.

3.5.3. Weight-2 Codes and FWM Processing

In [14,46] another encoding technique for multi-wavelength labels is proposed where only two wavelengths in the label band are active in a given label. Label recognition is achieved by generating the FWM product of these two wavelengths. One of the FWM products is selected by optical filtering and is used as the label identifier. Appropriate spectral spacing between the two bins ensures a unique FWM product for each label. A bank of optical filters or an AWG separates out the label identifiers; the active output is detected and used as the control signal. In this case, the total number of labels is $\binom{L}{2}$ for L wavelength bins.

An algorithm is developed in [46] to search for optimum codes compatible with FWM label identification. For example, 36 codes can be constructed with 9 bins. Note that the spacing between the occupied bins is not equal. In this example, nine bins out of 36 equally spaced spectrum slices are dedicated for labels (the widest space between wavelengths in a code is 35 bins.) The remaining 27 bins are free and can be used for other applications. At the label recognition module, a bank of 36 optical filters or an AWG with 88 channels is required for selecting the label identifiers. Note that in the case of an AWG, only 36 output channels are used, as only these correspond to unique label identifiers. An AWG with large number of output channels is a disadvantage of this system. Furthermore, high cardinality codes might be less efficient in generating the FWM product; FWM of distant tones results in less power. Also spectral efficiency is poor unless unused bins can be exploited.

In the packet switching node, one label recognition module is required per input port. Therefore, N nonlinear devices (nonlinear SOA in [46]) and N AWG (or filter bank) is required for an $N \times N$ node. The switch fabric in this scheme is cross bar; N^2 optical switching gates (optically or electronically controlled) make an $N \times N$ switch. An optically-controlled switch based on SOA-MZI is used in [14] for packet switching using FWM processing. This scheme is more efficient than the correlation-based solution in terms of component count.

3.6. Switches Exploiting Binary Multi-Wavelength Labels

In this section, we will present two switch structures based on binary M-λ labels. In both cases the label is outside the payload band and is transmitted as a constant signal over the entire packet. The first switch, a wavelength recognizing switch (WRS), is a wavelength sensitive device performing all-optical switching [9]. There is no need for photodiodes and the label is directly fed to the switch as the control signal. In the second scenario, the interconnection of 2×2 electrically controlled switching elements creates a switch, also known as a data vortex, where contention is resolved internally [15].

3.6.1. Wavelength Recognizing Switch

An optical packet switching network using wavelength recognizing switch (WRS) device has been proposed in [8] for routing binary M-λ labels. The WRS is a switching element controlled by applying an optical signal. No electrical-to-optical conversion is required for routing a packet. The device has two inputs (I: input and C: control) and one output (T), as seen in Figure 3.16. A control wavelength is applied at the input C. If the signal at input I contains a wavelength that matches the wavelength of input C, an index grating is generated by the beat frequency between the two signals. The index grating deflects the signal from input I to the output (T).

The device has been implemented with a broad-area SOA. The control and the packet beams are coupled to the SOA with 6° separation angle as illustrated in Figure 3.16. The

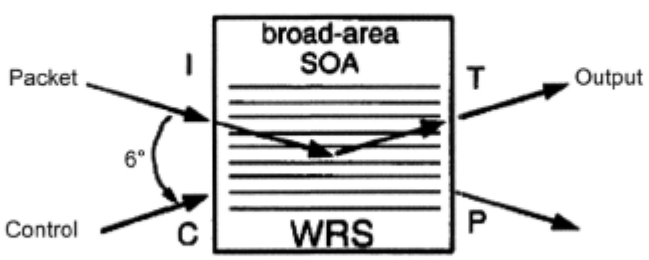

Figure 3.16. Schematic diagram of wavelength recognizing switch implemented by broad-area SOA [9].

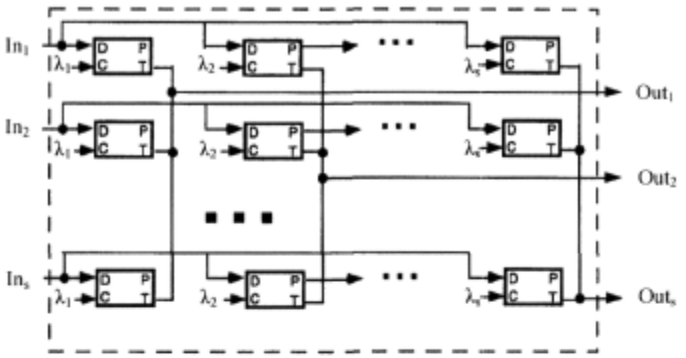

Figure 3.17. An s×s crossbar switch using WRS devices [9].

packet is deflected and coupled into the fiber at the output. The power of the address wavelength must be high, in the range of several milliwatts, to ensure good switching efficiency. The total power of the packet must be low enough not to saturate the gain of the SOA. The switching efficiency of the WRS, defined as the ratio of the output to the input signal power, is 8.2 dB. The extinction ratio, defined as the ratio of the signal output power when the control signal is on to that when the control if off, is 28.8 dB.

An optically controlled self-routing multistage switch such as Omega-Shuffle can be constructed by interconnecting several WRSs, as suggested in [9]. The Omega-Shuffle has s^g inputs and s^g outputs with g stages of $s \times s$ crossbar switches; an $s \times s$ crossbar is illustrated in Figure 3.17. When the label of an incoming packet contains wavelength k, it will be directed to the k^{th} output. The address wavelength space is partitioned into g wavebands corresponding to g stages. The wavebands are sub-partitioned into s bins where each slot controls a switching elements of a given stage. s lasers are required to produce the control signals for each stage. The response time of WRS is about 0.6 ns, therefore a guard time of approximately 1 ns is required between data and label as illustrated in Figure 3.18. The total guard time required for a switch with g stages would be g ns. The performance of the Omega-Shuffle switch has been studied in [9] via simulation. A switch with seven stages of 2×2 crossbars ($g = 7, s = 2$) can maintain 10^{-9} BER while routing packets containing four different wavelength payloads.

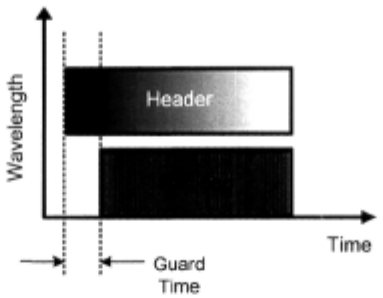

Figure 3.18. Illustration of the guard time required in WRS structure.

3.6.2. Data Vortex Switch

Binary M-λ labels have been used in an optical packet switch structure, called a data vortex [15,47,48]. Data vortex is a switch fabric constructed by interconnecting 2×2 switching elements (SE). The architecture couples deflection routing and a virtual buffering mechanism to resolve contention [38]. The packets are fixed-length and synchronous and each SE processes only one packet during a given time slot. The scheduling is based on distributed-control where each switching node is equipped with control logic making the routing decision. Control messages are exchanged between SEs before the packet arrives, to establish the switching configuration. The header bits are encoded using binary M-λ signaling and the payload is out-of-band (Figure 3.15). The header includes a frame signal to indicate the presence of a packet. At each switching element the corresponding header bit and the frame signal are filtered and used by the control logic for a routing mechanism.

a) Fabric architecture

The interconnection of SEs of the data vortex constructs concentric cylinders, as shown in Figure 3.19. The input (output) ports of the switch are connected to the SEs of the outermost (innermost) cylinder. As shown on the side view of the switch, the SEs over the same cylinder are connected in a fashion similar to the banyan switch interconnections [49,50]. The SEs of the same height in the adjacent cylinders are connected as shown by the dashed lines in Figure 3.19.

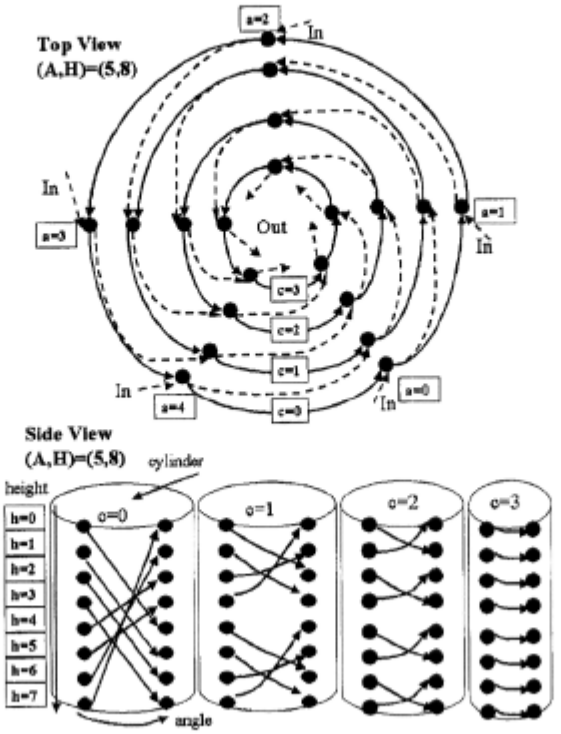

Figure 3.19. The top and side view of data vortex switch with $A=5, H=8, C=4$ [47].

The number of SEs along the height and circumference of a cylinder are H and A respectively, and the number of cylinders (C) is given by $C=\log_2(H)+1$. The maximum number of input/output ports is HA, therefore, $HA(\log_2(H)+1)$ SEs are required to realize an $HA\times HA$ switch fabric. The coordinates of the SEs are indicated by (a,c,h), where, $0\le a<A, 0\le c<C, 0\le h<H$.

The header of the packet contains the height and the angle information of the desired output port. Routing on each cylinder (except the innermost cylinder) is performed using one bit of the header related to the destination height. The packet is routed to its destination height in a bitwise banyan manner, *i.e.*, when a packet enters an SE (a,c,h), the c^{th} bit of the packet header is compared to c^{th} most significant bit in the height coordinate (h) of the SE. If the bits are equal (the packet is at the right height), the packet is forwarded to the inner cylinder; otherwise it is routed within the same cylinder. Once the packet reaches its

destination height, in the innermost cylinder, it is routed to the desired output according to the destination angle.

Contention is resolved in the data vortex by deflection routing. Adjacent SEs exchange electronic signals concerning deflection. The deflection control signal is generated by the SE when it sends a packet to a given output port. The signal prevents the adjacent SEs from forwarding packets to the same node. The deflection signaling gives priority to the packets traversing a cylinder over packets trying to enter that cylinder. When an SE receives a deflection signal it forwards the packet within its own cylinder. The packet stays in the same cylinder until it is again in a position to drop to an inner cylinder. This mechanism provides a virtual buffer for a blocked packet. The switch is kept synchronous by properly designing the link latencies.

b) Switching element

The interconnection of the 2×2 SEs forms the data votrex switch fabric. Two SOAs are used as gates in each SE as illustrated in Figure 3.20. The incoming packet is split over the SOAs; only one SOA receives enough current to forward the packet and the other SOA blocks the packet. The forwarding SOA also provides gain to compensate for the splitting loss. The switching rise and fall times of the SOAs are about 1 ns. The driving current is generated by control logic using the label information from the packet. Approximately 30% of the packet power is tapped off and filtered to extract two control wavelengths

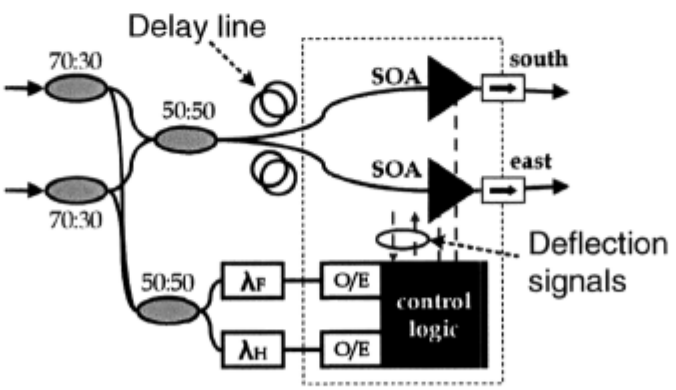

Figure 3.20. The schematic diagram of a switching element used in the data vortex [15].

corresponding to frame and header (λ_F, λ_H). Two low-speed photodiodes detect the control wavelengths; the operational frequency of the photodiodes matches the packet rate. The electronic control module decides which output port should be active, according to the signals received from the photodiodes, and also the electronic deflection signal received from the adjacent SEs. The electronic logic circuit is implemented with high-speed positive coupled emitter logic (PECL) gates in order to achieve minimal latency.

c) Performance of the data vortex

The performance of the data vortex has been studied by simulation under different traffic conditions [47,48]. The injection rate, defined as the ratio of successful packet injections into the switch over the number of injection attempts, is used to represent the throughput of the switch. The injection rate versus load for a switch with $A = 5$ and different heights is illustrated in Figure 3.21 [47]. The incoming packets have uniform distribution over the destination addresses and are independent of the packets coming into other input ports or time slots. According to the plot, for load=1, 26.1% of the injected packets are successfully routed for a switch with 640 (5×128) ports. The injection rate reduces to 23.7% for a switch with >10 k (5×2048) ports under the same traffic condition.

The switch latency, the mean number of hops propagated by the packets, is examined in [47,48]. Figure 3.22 shows the latency as a function of the switch height for uniform traffic [47]. As each hop requires about 10 ns for processing, a data vortex switch fabric with 2000 ports causes packet latency of about 200 ns.

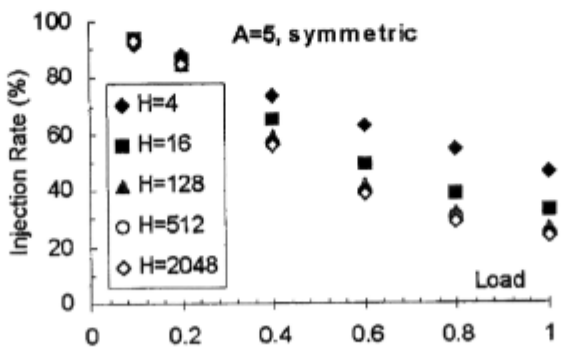

Figure 3.21. The injection rate versus load for a data vortex with $A = 5$ and different heights [47].

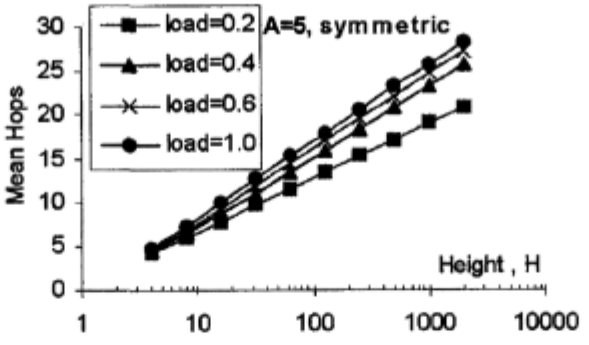

Figure 3.22. The mean number of hops for a data vortex with $A = 5$ versus switch height for different traffic load [47].

Simulation studies of the data vortex for nonlinear and bursty traffic are also reported in [48]. The switch maintains robust throughput and latency performance under modest nonlinear traffic, but an excessive degree of nonlinearity will severely limit the scalability.

An experimental demonstration of a 12×12 data vortex switch containing 36 2×2 switching elements is reported in [15]. Successful packet routing to 12 output ports is verified with latencies around 100 ns. BER performance of 10^{-12} for multiple-wavelength payload at 80 Gbps (eight-wavelength at 10 Gbps) is reported in [15].

Data vortex with high throughput was obtained when input ports at only one angle are used [51]. In this case, the rest of the angles are used exclusively for deflection to provide more virtual buffers. The higher acceptance rate is achieved at the cost of a higher number of components; component count increases by a factor proportional to the angle size. For instance, [51] shows that a vortex with 2048 ports and maximum traffic load maintains a nearly 100% acceptance rate if the angle size is six; that is six times more elements are required to implement the switch.

3.6.3. Conclusion on Switches

We presented two optical switch structures consistent with binary M-λ labels, *i.e.*, the packet format that we use in chapter 5. In the first scenario, wavelength-recognizing

switches construct an Omega-Shuffle. The switch is all-optical, in the sense that no optical to electrical conversion is required. The issues that WRSs face include crosstalk, polarization dependency, high power label, optical power dependency, and wavelength stability in the range 500 MHz. Furthermore, the Omega-Shuffle is internally blocking; an electrical overlay is required to resolve internal blocking and output contention.

The second scenario, data vortex, is constructed by electronically controlled switching elements using SOAs. The advantage of this switch structure is contention resolution by internal deflection routing. However, complexity and necessity of precise timing between the control modules are the issues of this system.

In chapter 5, we propose using a tree switch for binary M-λ label switching. The proposed structure can be implemented by either optically or electronically-controlled switching elements. For example, WRS is applicable in the proposed architecture if used as 2×2 switching element. The switch is internally non-blocking, but output packet conflict can occur; external contention resolution should be used. This structure is simple, practical, high-speed and cost-efficient [12]. A comparison between our proposal and data vortex is given in section 5.3.

3.7. Label Swapping Techniques

In addition to forwarding, another process required at intermediate GMPLS nodes is label swapping, *i.e.*, the incoming label must be replaced by a new outgoing label. The packet will continue on its way with this new label to the next hop. The new label may be either newly generated or converted from the old label. In some cases, converting the old label to the new label will result in more efficiency, less hardware, and/or higher speed.

In time domain and subcarrier label encoding techniques, optical label conversion is impossible. Techniques for converting BPSK and two dimensional OCDMA codes are proposed in the literature. An optical label swapper for BPSK OCDMA labels based on fiber nonlinearities was presented in [6]. A control signal is coupled to the fiber to propagate along the label. The high intensity of the control signal perturbs the refractive index of the fiber due to cross phase modulation. As a result, the phase of the label

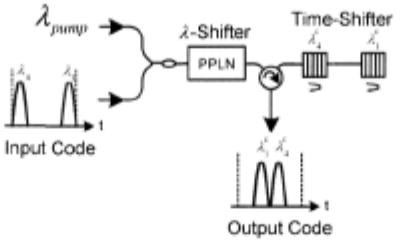

Figure 3.23. Code converter for time-wavelength OCDMA [52].
PPLN: periodically-poled lithium-niobate

elements are converted, as required, to achieve the new label. For example, the conversion of a 4-chip code [0000] to [00$\pi\pi$] was demonstrated experimentally in [6].

All-optical conversion for two dimensional (time-wavelength) OCDMA codes was demonstrated in [52]. In this scheme, the wavelength and the time position of the OCDMA chips are modified using the configuration of Figure 3.23. Wavelength conversion is achieved by difference frequency generation (DFG) in a periodically-poled lithium-niobate (PPLN) waveguide. The wavelength set of the incoming label is first mapped onto its mirror image with respect to the pump wavelength λ_{pump}. The spectrum of the input wavelengths as well as the converted wavelengths of an experiment reported in [52] is illustrated in Figure 3.24. DFG converts each wavelength to a new one at $\lambda^c \approx 2\lambda_{pump} - \lambda_{signal}$.

The time shifting required for generating the new label is enabled by FBGs that are tuned to the converted wavelengths. The FBGs are located in series to change the position of the PPLN output pulses. The experimental setup of [52] is designed to convert OCDMA codes

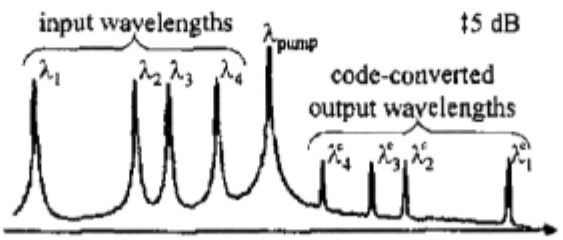

Figure 3.24. Wavelength shifting using a PPLN waveguide [52].

with four wavelengths and four time slots at 10 Gchip/s.

An optical label swapper for converting multi-wavelength labels is proposed in [21]. This structure is based on semiconductor fiber ring laser using SOAs. The SOAs provide sufficient gain for the loop to lase and the wavelength of the laser is determined by optical filters in the loop. The modulation of the ring laser at the label rate is achieved by cross gain modulation (XGM) in the SOA, that is, when the incoming label is injected the gain of the loop decreases due to XGM and the laser stops lasing. Non-inverting operation of the swapper is achieved by an primary SOA before the ring laser; XGM converts the incoming label to an intermediate wavelength. We have used the label swapper [53] in a packet switching experiment with binary M-λ labels and multi-stage switch (as reported in chapter 5), which is discussed in detail in appendix A.

Chapter 4

SAC Multi-Wavelength Labels with Label Stacking

4.1. Introduction

In chapter 3 we reviewed existing label switching methods and discussed their advantages and disadvantages. We concluded that correlation based label processing techniques are promising due to their ultra-high speed. However, these methods either inherently suffer from high splitting losses and thus cannot be scaled to large number of users [6,8], or employ complex technologies such as mode locked lasers (MLL) and phase modulation

[6,7,44]. In this chapter we present our first solution for high-speed label switching that mitigates these problems.

We propose for the first time, to our knowledge, combining OCDMA labels with the concept of label stacking. Our objective is to benefit from high-speed OCDMA label recognition, and meanwhile to mitigate the main obstacle of OCDMA based label switching networks, *i.e.*, high splitting loss and large component count. As explained in section 2.2, label stacking in MPLS/GMPLS networks has been developed to implement important networking applications such as hierarchical routing, resulting in smaller lookup tables at the nodes. In label stacking two or more labels are appended to a GMPLS packet. By combining hierarchical addressing with OCDMA label techniques, and reducing the number of labels that need to be checked at each hop, we can reduce the high splitting loss.

Among the multi-wavelength label techniques discussed in section 3.5, we use spectral amplitude codes (SAC) [16-19] as optical labels due to their compatibility with label stacking, low-cost implementation, and high switching speed. We stack labels using SAC codes transmitted simultaneously during the optical packet interval. The format of our optical packet is throughput efficient as there is no additional time overhead to the payload interval for the label.

Both packet formats addressed in section 3.5.1, *i.e.*, separable SAC-label [54,13] and SAC-encoded payload [13] are considered in this chapter. In the first case, the payload and the label have different wavelengths and independent modulation rates (see Figure 3.13 (a)). In the second case, the SAC is used to encode the payload and the label is implicit in payload bits (see Figure 3.13 (c)). This second approach requires less bandwidth and components compared to the first method. However, the payload bit rate is limited by intensity noise.

In section 4.2, we provide the architectural overview of label stacking with spectral amplitude codes. We discuss the proposed hardware for the forwarding node in section 4.3, which can be implemented with cheap and commercially available optical components. The experimental proof of the concept of separable SAC-labels and SAC-encoded payload are reported in sections 4.4 and 4.5 respectively. We demonstrated an optical switching path of two hops, using optical packets with two stacked labels. In both cases, we achieve label

recognition time in the range of a few nanoseconds corresponding to the traveling time of light through the label processing branch, plus the detection time of the photodiode. We experimentally achieve error free performance (BER<10^{-9}) after 80 km of fiber and 2-hop switching. The data rates for packets with separable SAC-labels and SAC-encoded payload are 10 Gbps and 2.5 Gbps respectively [54,13].

4.2. Label Stacking with Spectral Amplitude Codes

Recall from section 2.2.1 that label stacking is used in GMPLS systems by assigning more than one label to a single packet to support hierarchical addressing [20,25]. One of the advantages of hierarchical addressing is reducing the size of lookup tables. Label stacking is beneficial for label processing based on correlation techniques, as the number of labels that need to be checked at each node is reduced, resulting in reduced splitting loss.

Here, we use optical labels achieved via spectral amplitude codes (SAC) [16-19]. We use the orthogonal properties of SAC to send multiple time-overlapped SAC-labels in each optical packet, where each label corresponds to a network region [54,13]. For example, in Figure 4.1 (a) two time-overlapping SACs are transmitted over the payload interval. Interference from the labels in the same stack will be rejected at the receiver, as the codes have low cross-correlation and high autocorrelation peak. The number of stacked labels is limited by the capacity of decoders to successfully decode in the presence of multiple

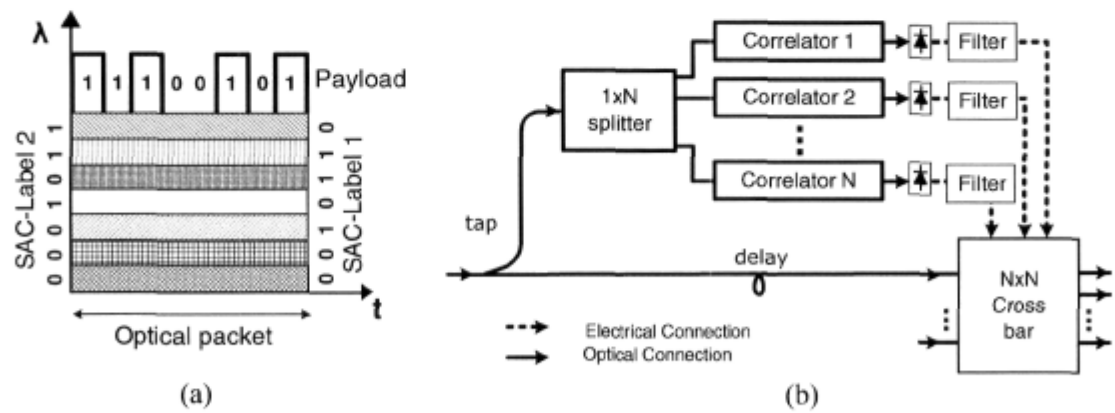

Figure 4.1. Optical packet switching with stacked SAC-labels; (a) the optical packet with separable SAC labels, (b) the structure of the forwarding node.

interfering labels [16-18].

We use SACs because of their low cost implementation, fast recognition, and compatibility with label stacking. SAC is based on the encoding of incoherent broadband sources (BBS) in the frequency domain [16-19]. In contrast to other OCDMA techniques, in SAC, the data rate and the chip rate are equal. Therefore, increasing the code cardinality does not lead to shorter chips or lower data rate. Moreover, in SAC-CDMA systems the receiver bandwidth is limited to the data rate, in contrast to time domain CDMA techniques for which the detection of short pulses requires high electrical bandwidth. SAC codes are designed such that interference can be cancelled using differential detection [16-19]. These systems are very low complexity and low cost compared to other OCDMA techniques.

The number of available labels is limited to the code cardinality. Various codes have been reported in the literature including Hadamard, *m*-sequence [16], and balanced incomplete block design (BIBD) [55,56] each with an associated cardinality. For example the cardinality of BIBD codes is equal to the code length.

4.3. Hardware

The structure of the forwarding module when using SAC as labels is illustrated in Figure 4.1 (b). At the forwarding module, as many correlators (composed of a decoder, a complementary decoder, and a balanced photodiode) as the number of labels in the corresponding label set are used in parallel. The label stack is split and distributed to all correlators to find a match. If any of the labels matches one of the correlators, an autocorrelation output signal with the duration equal to the packet interval will be generated. This control signal in turn establishes the desired connection in a cross-bar switch. In Figure 4.1 (b) only forwarding of packets is considered and other functions required at a node, such as packet contention resolution and label swapping, are excluded for simplicity.

The broadband source required for SAC generation can be either incoherent BBS or multi-wavelength source. The incoherent sources are more cost efficient. The SAC encoder and decoder can be implemented by fiber Bragg gratings (FBG). The decoder consists of two

FBGs, one corresponding to the code and the other to the complementary code, and a balanced photodiode; the combination of them eliminates the interference [16,17]. A total of N^2 switching elements, correlators, and balanced photodiodes are required to realize an $N \times N$ forwarding node.

As addressed in section 2.2, another process that is performed in GMPLS nodes other than forwarding is label swapping, which means replacing the old label with a new one. The new label will be used for routing at the next hop of the same network domain. Wavelength conversion techniques can be employed for label swapping to convert SAC labels. For example, the all-optical SAC label swapper introduced in section 3.7 and appendix A can be used [21,53]. However, a tunable label swapper and also the optical table lookup processing required for determining the new label are complex to attain optically.

An alternative to label swapping is to establish the optical path at the ingress node and assign one label for routing at each hop, then attach all the labels to the packet. The forwarding decision is taken according to the label assigned to the transit node, avoiding optical label swapping at the expense of having a larger number of stacked labels. Thus, label stacking can be used for hierarchical addressing, and also for transmitting the complete hop-by-hop routing information.

4.4. Optical Packet with Separable SAC-Labels

In this section we consider the case where the wavebands of label and payload are orthogonal. For example, a laser with a center frequency out of the spectral band of labels can be used for the payload (see Figure 3.13 (a)). As illustrated in Figure 4.1 (a), the labels are modulated at the packet rate, accompanying a high bit rate payload [13]. The independence of label and payload bit rates is achieved at the cost of higher bandwidth utilization; a laser is dedicated to the payload.

As the label and payload are transmitted simultaneously during the packet interval, the label does not extend the packet duration, resulting in higher throughput. Low-bit rate labels allow utilization of low-speed components for label recognition. Furthermore, the

intensity noise*, which is the main limitation for high-speed incoherent transmission, is not a restriction in this approach. Note that intensity noise is proportional to B_e / B_o, where B_e and B_o are electrical and optical bandwidth respectively [57,17]. A narrowband (small B_e) low-pass electrical filter is used for labels, leading to very low intensity noise over the label, thanks to its low bit rate. In the following subsections, we explain the experimental setup for this packet type and present measurement results. In section 4.5, we will introduce the second optical packet format, SAC-encoded payload, where the waveband used for the labels is also used for the payload.

4.4.1. Experimental Setup

The experimental setup for packets with separable SAC-labels is illustrated in Figure 4.2 [13]. The setup is composed of three major parts: a packet generator, and two forwarding nodes. The packet generator creates the optical packet containing a payload and two SAC-labels that are all transmitted simultaneously. Label 1 and 2 are used at the first and second forwarding nodes respectively. At the forwarding nodes, we use modulators (2.5 Gbps SDL Mach-Zehnder) as switching gates to emulate the cross bar switch shown in Figure 4.1 (b). If the packet contains the SAC-label matched to a forwarding node decoder, it is passed by the modulator, otherwise it is blocked. The packet is routed by the two forwarding nodes and the BER of the payload bits is measured at different points of the optical path. The setup is explained in more details in the following paragraphs.

A laser at 1550 nm and a BBS are used to generate the payload and the labels respectively. The laser is modulated by a 2^{10}-1 non-return to zero (NRZ) pseudo random bit sequence (PRBS) at 10 Gbps, representing the payload bits†. The 2^{10}-1 PRBS length is chosen to

---

* Intensity noise is the most important noise source when incoherent BBS is used. This noise is power dependent, thus increasing the signal power does not improve the signal to noise ratio (SNR). Intensity noise can be modeled for a given SNR by gamma distribution. The SNR can be obtained by $SNR = mB_e^2 / (2B_o - B_e)B_e$; $m = 1$ for polarized signal, and $m = 2$ for unpolarized signal.

† We generate a continuous pattern and chop it with a modulator (Mod 2) to form the payloads. This is because the error detector is used in *Burst gating* mode; this mode can only be used with a continuous PRBS pattern. In chapter 6, we use the Block BER option of the error detector, therefore, the pattern generator is programmed to generate the pattern of payloads and the silence period. In this case, the setup is simpler as the second modulator is eliminated.

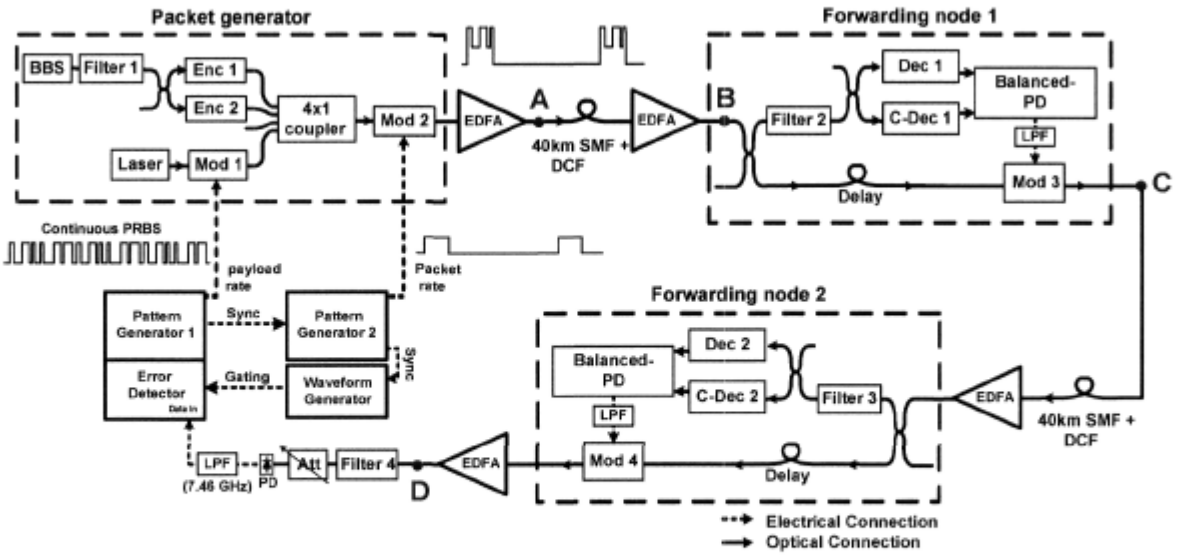

Figure 4.2. Experimental setup for packet switching with separable SAC-labels; BBS: broadband source, Mod: external modulator, Enc: encoder, Dec: decoder, C-Dec: complementary decoder, EDFA: Erbium doped fiber amplifier, LPF: low-pass filter, PD: photodiode, SMF: single mode fiber, and DCF: dispersion compensating fiber.

match the packet length, i.e., packets of 1023 bits. In order to generate the labels, the BBS is first filtered by a band-pass filter of 9.6 nm (Filter 1 in Figure 4.2) and is spectrally encoded by two parallel encoders. The encoders are chirped FBGs used in transmission, in which the apodization profile determines the transmitted spectral bands of the codes. Because the gratings are used in transmission there is no delay between the code wavelengths. The wavelength bins are 1.28 nm (160 GHz), the transmission loss is 1.1 dB, and the extinction ratio is 25 dB (Figure 4.3 (a)) [19]. Note that the 9.6 nm label bandwidth is only used for the proof of concept and it can be reduced by using multi-laser source where the laser bins are at short spectral distances [58].

The two SAC-labels are BIBD codes proposed in [55], with code length 7, weight 3 and cross correlation of one. The codes corresponding to encoder 1 and 2 are (0010110) and (0001011) respectively. The measured spectrums of the encoders are illustrated in Figure 4.3 (a). These encoders are designed to improve the performance compared to previous code families by reducing the intensity noise [17], which is already low here due to low packet rate.

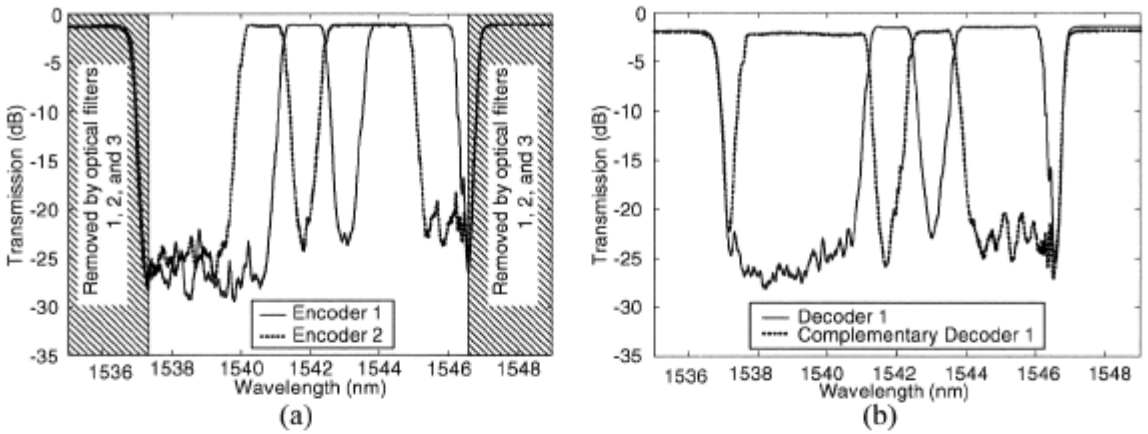

Figure 4.3. Transmission spectrum of (a) encoder 1 and 2, and (b) decoder 1 and complementary decoder 1.

The payload and labels are combined and sent to modulator 2 where the packets are created. Modulator 2 is driven by the second pattern generator synchronized with the first pattern generator. The second pattern generator produces rectangular pulses at the packet rate. The bits of the payload are packaged into packets 4092 bits long (equal to 4 periods of the PRBS sequence), with 0.41 μs duration and 3.7 μs separation (Figure 4.4). The packet length and separation are chosen for experimental convenience, (*e.g.* due to the AC-coupled modulator) and there is no conceptual limitation on this approach for implementation of variable packet length and short packet separation. The generated packets are amplified and sent through 40 km of single mode fiber (SMF) plus dispersion compensating fiber (DCF). The average optical power that we launch into the fiber link is -5dBm. The residual dispersion of the link is 0 ps/nm.

At the first forwarding node, a part of the signal is tapped off and sent to the label recognition unit. Filter 2 which is similar to Filter 1 (9.6 nm), reduces the out-of-band amplified spontaneous emission (ASE) of the amplifiers. The filtered signal is split over two branches; a decoder (Dec 1) identical to encoder 1, and a decoder that is the logical complement of encoder 1 (C-Dec 1). The measured spectrums of Dec 1 and C-Dec 1 are shown in Figure 4.3 (b). The outputs of the branches are sent to a 830-MHz balanced photodiode which provides an electrical output voltage proportional to the power difference of the two optical inputs. The subtraction removes the interference from the unmatched

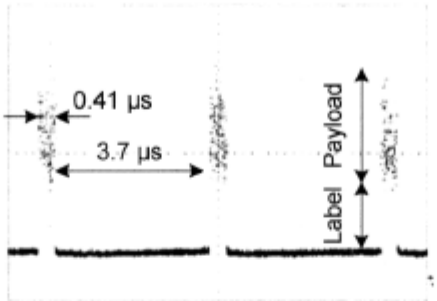

Figure 4.4. Oscilloscope trace of the generated packets with separable SAC-labels at point A of Figure 4.2.

code, *i.e.*, from other labels in the label stack. The electrical signal is amplified and low-pass filtered (with a 100 MHz filter) to remove the noise after photodetection*. This signal is then used as the control signal that remains for the entire packet interval and opens the gate (Mod 3). If the spectral code of the received packet is matched to the decoder pair, the balanced photodiode generates the control signal. However, if the packet does not contain code 1, no control signal is generated and the gate remains closed. Synchronization between the control signal and the packet at the modulator is assured by a fixed optical fiber delay line.

The packets that pass node 1 are forwarded to node 2 after traveling over 40 km of SMF followed by DCF. The residual dispersion of this link is -20 ps/nm. At forwarding node 2, the second code detection occurs. This node contains the decoder pair (Dec 2 and C-Dec 2) matched to the second code. As for the first node, if the packet contains SAC-label 2, the control signal opens the gate (Mod 4) and the packet is sent out.

If the initial packet contains both codes 1 and 2, it reaches the end of the link. When one of the two labels is missing, the packet is not received at the end of the path. In our setup only one correlator is used at each node. In a complete system, several parallel correlators would exist and the packet would be forwarded by the matched correlator.

---

* The rising edge of the control signal determines the guard time between the label and payload (see Figure 3.18). We do not focus on the rise time of the control signal here, nor in following chapters, as we used in-house electronics for amplification (in the case of Chapter 5 thresholders) that were not optimized for sharp edges.

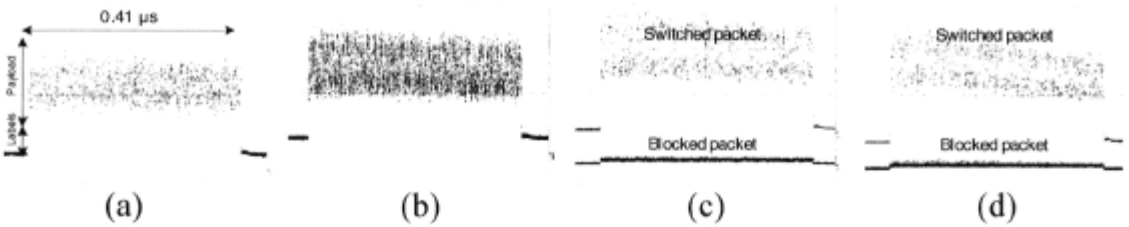

Figure 4.5. Oscilloscope traces of a single packet with separable SAC-labels; (a) before transmission (point A of Figure 4.2), (b) after traveling over 40 km of SMF plus DCF (point B of Figure 4.2). (c) Upper trace is the packet with appended label after successful switching at the first hop, lower trace is the blocked packet when code 1 is not present in the label (point C of Figure 4.2). (d) Upper trace is the packet after successful switching at the second hop, lower trace is the blocked packet when code 2 is not present in the label (point D of Figure 4.2).

We measure the BER of the payload at points A, B, C, and D of Figure 4.2. At each of these points, we filter out the label (with filter 4) and measure the BER of payload bits using an error detector in *burst gating mode*. In this mode, the BER is measured only at a window determined by the *gating input*. A waveform generator synchronized to the packet rate provides a gating signal to the error detector that measures the BER over 80% of the bits in the middle of packet intervals*. (In BER measurements presented in chapter 5, we use an error detector equipped with *Block BER* option; in this mode we are able to specify the location of bits in the payload to be used for the BER measurement). We measure the BER versus the average received power of the payload by using a variable optical attenuator. An electrical low-pass filter with a pass band equal to three quarters of the bit rate (7.46 GHz) is used after the photodiode to remove the out-of-band noise.

4.4.2. Results and Discussion

The oscilloscope traces of the packet as it travels through the system are shown in Figure 4.5 [13]. The packet containing both the payload and its labels after the transmitter (point A in Figure 4.2) and after the 40 km fiber link (point B) is illustrated in Figure 4.5 (a) and (b) respectively. The label is identifiable as an offset added to 10 Gbps payload bits.

---

* This restriction is due to experimental expediency and could be reduced with better electronics generating the control signal.

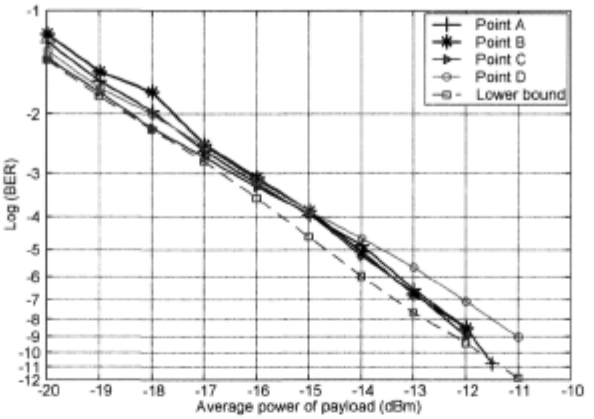

Figure 4.6. Bit error rate versus average received power of payload over packet duration for separable SAC-labels after transmitter (point A of Figure 4.2), after 40 km of SMF and DCF (point B of Figure 4.2), after forwarding node 1 (point C of Figure 4.2), and after forwarding node 2 (point D of Figure 4.2). The dashed reference line corresponds to the laser modulated at 10 Gbps by a continuous PRBS signal (*i.e.*, after Mod1 inFigure 4.2).

On the upper trace of Figure 4.5 (c), we show the forwarded packet at point C of Figure 4.2. The lower trace, having the same scale but different offset, shows the blocked packet when code 1 is not in the label stack. The upper trace of Figure 4.5 (d) shows the packet after traveling over the second 40 km fiber link and passing through the second node (point D in Figure 4.2). The lower trace shows the packet that is well extinguished when code 2 is not in the label stack. The gradual amplitude drop of the packet is due to the 830 MHz AC-coupled balanced photodiodes and electrical amplifiers that generate the control signal, not being well adapted to the detection of low rate packets. It would not be present with electronics designed for lower modulation rate.

In order to measure the BER, we used a narrowband optical filter of 0.2 nm (Filter 4 in Figure 4.2) to isolate the laser that represents the payload. We measured the BER of payload bits at points A, B, C, and D of Figure 4.2. The BER curves versus the average received power of payload over packet duration are presented in Figure 4.6. Neither transmission nor switching of the packets causes any performance degradation. In all cases we could obtain error-free transmission with average power of the payload around -12 dBm. The dashed line in Figure 4.6 corresponds to the laser modulated at 10 Gbps by a continuous PRBS signal (*i.e.*, after Mod1 in Figure 4.2). We draw this BER curve as a

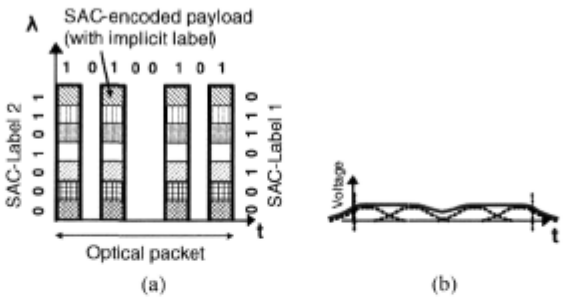

Figure 4.7. Optical packet with SAC-encoded payload. (b) The control signal to be used for the switch in the SAC-encoded payload case.

reference for comparison, as it shows the lower bound of BER measurement, with our error detector and photodiode. As seen in Figure 4.6, there is only a small difference (about 1 dB power penalty) between the BER of the switched packets and the ideal case (dashed line).

4.5. Optical Packet with SAC-Encoded Payload

In section 3.5 we introduced the packet format where the payload is encoded bit-by-bit (Figure 3.13 (b)). In this section we consider this type of optical packet with SAC-labels in which the same broadband source is used for both the labels and the payload. Labels and payload occupy the same spectral band, resulting in lower optical bandwidth utilization compared to separable SAC-labels [13]. In this approach, the wavelength bins of the label stack are selected by the corresponding encoders and are then modulated by the payload bits. Indeed, the packet consists of a payload with SAC-encoded bits and an implicit label. Here again, the packet interval is used efficiently, as the label does not extend the packet duration. We exploit label stacking by encoding the payload bits with more than one SAC.

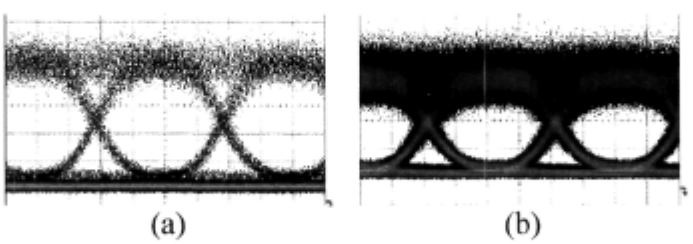

Figure 4.8. Eye diagrams of SAC-encoded payload at (a) 2.5 Gbps, and (b) 5 Gbps. Higher bit rate suffers from more intensity noise.

For example, Figure 4.7 (a) illustrates the case where two SACs are present. The structure of the forwarding node is the same as the system shown in Figure 4.1 (b). Again, part of the packet is tapped off and is split over the parallel correlators. The low-pass filters (or electronic circuits) after the photodiodes extract the envelope of the packet which open the gate during the entire packet interval (Figure 4.7 (b)). The transmission rate of this approach is lower (2.5 Gbps) than the packet with separable SAC-labels (10 Gbps). This is due to the intensity noise that limits the transmission rate of incoherent broadband sources [17,57], as explained in section 4.4. This limitation could be removed using a multi-laser source [59]. Figure 4.8 shows the noise power on the eye diagrams of SAC-encoded data at 2.5 Gbps and 5 Gbps. The two SACs demonstrated on Figure 4.3 are used where each code contains three wavelength bins of width 1.28 nm. We used low-pass electrical filters of bandwidth 1.87 GHz and 3.7 GHz for 2.5 Gbps and 5 Gbps data, respectively. Because of the poor eye diagram at 5 Gbps, BER measurements were only made at 2.5 Gbps.

4.5.1. Experimental Setup

The setup for the packet switching experiment with SAC-encoded payload is shown in Figure 4.9 [13]. We emphasize again that only simple and inexpensive components have been used. The setup consists of a packet generator and two forwarding nodes. In the packet generator (the same as Figure 4.3 (a)), the BBS is filtered and 9.6 nm of its bandwidth is sent to two parallel encoders through a 1×2 coupler. The encoder outputs are coupled to modulator 1 which is driven by a 2^7-1 NRZ PRBS at 2.5 Gbps, representing the payload bits. The 2^7-1 PRBS length is chosen to match the packet length. The continuous PRBS is chopped by the second modulator (Mod 2) to create optical packets 1016 bits long (corresponding to 8 periods of the PRBS sequence), 0.40 μs duration, and 3.65 μs separation. The sequence of generated packets is shown in Figure 4.10. The packets are amplified and transmitted toward node 1 by 40 km of SMF followed by DCF.

At the first forwarding node, part of the packet power is tapped off, filtered by filter 2 to remove out-of-band ASE, and sent to the decoder pair matched to code 1. If the following balanced photodetector bandwidth is matched to the bit rate, the payload bits will appear at

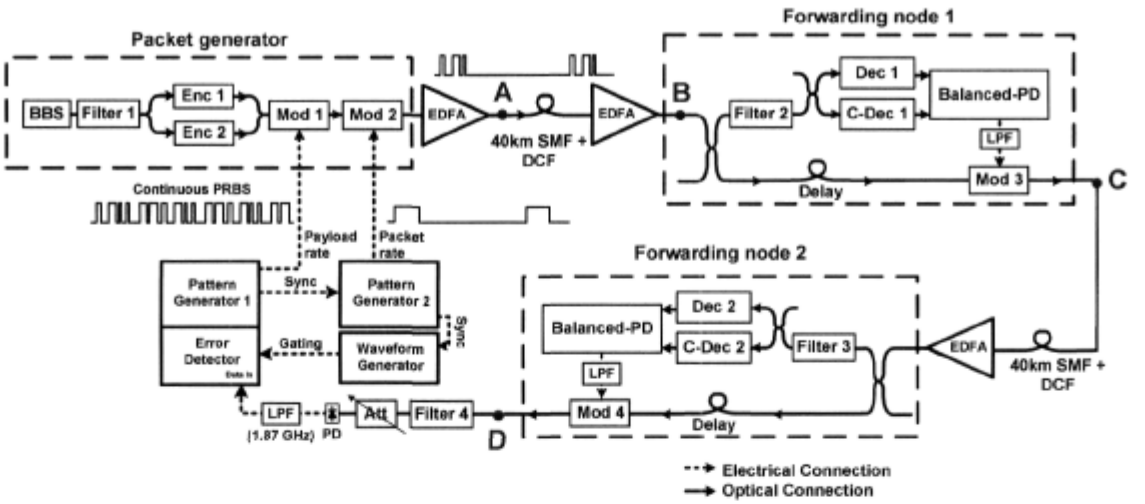

Figure 4.9. Experimental setup of packet switching with SAC-encoded payload; BBS: broadband source, Mod: modulator, Enc: encoder, Dec: decdcoder, C-Dec: complementary decoder, EDFA: Erbium doped fiber amplifier, LPF: low-pass filter, and PD: photodiode, SMF: single mode fiber, and DCF: dispersion compensating fiber.

its output. In our setup however, the 830-MHz photodiode followed by 10-MHz low-pass filter extracts the packet envelope of the 2.5 Gbps payload bits[*]. This signal is amplified and used as the control signal for the gating element (Mod 3). If the label is recognized properly, the packet is forwarded towards the next node through the second roll of 40 km of SMF plus DCF. The second forwarding node contains the decoders matched to code 2 and performs the label recognition process in the same manner as the first node. Only if the packet containing both labels 1 and 2, will reach the end of the optical path.

We measure the BER versus average optical received power of the packet at points A, B, C, and D of Figure 4.9. At each point, filter 4 (9.6 nm) is used to remove the out-of-band ASE of optical amplifiers, and a variable attenuator is used to change the received power. The low-pass electrical filter of bandwidth 1.87 GHz eliminates the out-of-band noise after detection.

---

[*] In a real system, an envelope detector should be used instead of the low-pass filter to provide shorter rise/fall times.

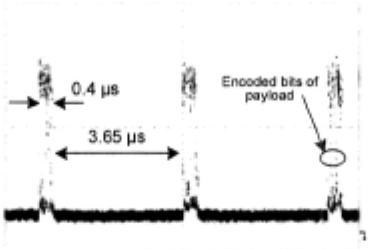

Figure 4.10. Oscilloscope trace of the generated packets with SAC-encoded payload at point A of Figure 4.9.

4.5.2. Results and Discussion

Figure 4.11 shows the packet traces at points A, B, C, and D of Figure 4.9 [13]. In Figure 4.11 (a) and (b), the packet after the transmitter (point A) and after passing through the 40 km fiber link (point B) is illustrated. In contrast to the packet traces of Figure 4.5, here no offset is added to the payload, as the labels are implicit in the payload bits. We show the forwarded packet after node 1 and 2 in Figure 4.11 (c) and (d), where the packet has traveled over another 40 km of fiber to reach node 2. The lower traces of Figure 4.11 (c) and (d) show the blocked packet when code 1 and 2, respectively, are absent in the label stack. The amplitude drop of the packet is again due to the high bandwidth AC-coupled balanced photodiode that is detecting the envelope of the low rate packets.

The BER of payload at 2.5 Gbps versus the average received power of packet is illustrated

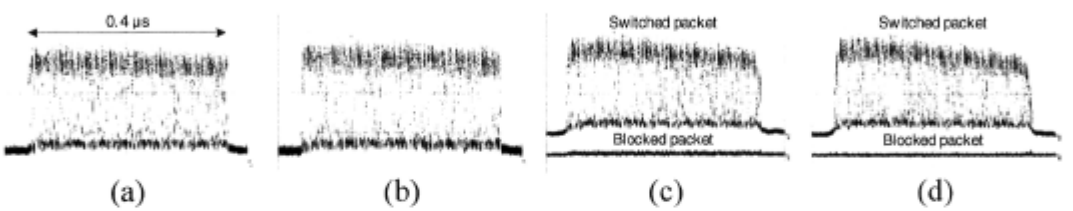

Figure 4.11. Oscilloscope traces of a single packet with SAC-encoded payload; (a) before transmission (point A of Figure 4.9), (b) after traveling over 40 km of SMF and DCF (point B of Figure 4.9). (c) Upper trace is the packet after successful switching at the first hop, lower trace is the blocked packet when code 1 is not present in the label (point C of Figure 4.9). (d) Upper trace is the packet after successful switching at the second hop, lower trace is the blocked packet when code 2 is not present in the label (point D of Figure 4.9).

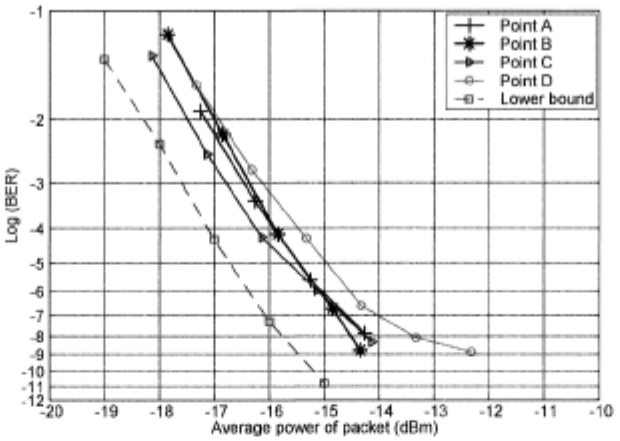

Figure 4.12. Bit error rate versus average received power of packet for SAC-encoded payload after transmitter (point A of Figure 4.9), after 40 km of SMF and DCF (point B of Figure 4.9), after forwarding node 1 (point C of Figure 4.9), and after forwarding node 2 (point D of Figure 4.9). The dashed reference line corresponds to the laser modulated at 2.5 Gbps by a continuous PRBS signal (*i.e.*, after Mod1 in Figure 4.9).

in Figure 4.12. The performance degradation after the transmission and switching is low, and the power penalty between back-to-back case (point A) and after the second node (point D) is less than 1 dB at 10^{-8} BER. The dashed reference line corresponds to the laser modulated at 2.5 Gbps by a continuous PRBS signal (*i.e.*, after Mod1 in Figure 4.9). The penalty between the switched packet BER and the dashed line, and the BER floor appeared in the figure is due to the intensity noise present when using a BBS. Using multiple lasers instead of the BBS will shift the curves toward the dashed line. Furthermore, we can achieve less bandwidth utilization by using lasers placed at short spectral distances.

4.6. Conclusion

In this chapter, we proposed for the first time an optical packet switching network using SAC-labels where the labels are stacked to realize hierarchical addressing. By reducing the size of lookup tables we mitigate the high splitting loss problem in correlation based label recognition techniques. This approach can be used for optical nodes to achieve ultra-fast packet forwarding. We experimentally demonstrated successful forwarding of packets for a network of two hops, connected by two 40 km fiber links. We examined transmission of

packets with separable SAC-labels and SAC-encoded payloads at 10 and 2.5 Gbps respectively; the first one having the advantage of high-bit-rate payload and the second one requiring less bandwidth and components. We showed that neither transmitting the packet over fiber nor the switching performed at each node degrades the BER performance. Therefore, this strategy is viable and has the potential to be applied for future all-optical packet switched networks.

Chapter 5

Binary Multi-Wavelength Labels with Multi-Stage Switch Structre

5.1. Introduction

In this chapter, we propose our second solution for high-speed label switching of optical packets: binary multi-wavelength (M-λ) labels with multi-stage switching. Similar to chapter 4, the optical labels consist of a collection of wavelengths, orthogonal to the payload wavelength [12]. However, the labels do not have any code properties such as OCDMA codes (section 3.5). Therefore, the number of available labels (2^L with L wavelength channels) is not limited to the cardinality of a code set, contrary to OCDMA-

based methods such as SAC-labels reported in chapter 4. Here, the label bits are transmitted in parallel, distinct wavelengths. At the forwarding node, an arrayed waveguide grating (AWG) separates the label bits and each bit (wavelength) controls a stage of a multi-stage switch. As the insertion loss of an AWG is fixed and does not scale with the number of branches, this structure overcomes the splitting loss problem, which is the main limitation in correlation-based label processing techniques.

In section 5.2, we explain the principle of operation of binary M-λ labels and the forwarding node architecture in more detail. We demonstrate that the proposed architecture achieves the objectives we discussed in section 1.2. Objectives 1, 2, and 4, *i.e.*, large number of labels, compatibility with hierarchical addressing, and scalability are covered in the architectural discussion. Objectives 3 and 5, *i.e.*, processing speed and cost efficiency, are addressed in the hardware discussion.

In section 5.2.4, we propose a solution to avoid label swapping in the network: time-stacking the optical labels. In this approach all the complex routing processing for determining the end-to-end optical route and the corresponding labels is performed at the electronic edge nodes. The entire set of labels is transmitted with the packet in non-overlapping time intervals. This method avoids the sophisticated procedure required for label swapping, such as table look up and tuning the swapper, and benefits from a simple, ultra fast optical backbone. We compare our proposed architecture with other scenarios in section 5.3.

We experimentally demonstrate the proof of the concept in section 5.4 by routing optical packets with binary M-λ labels, and also time stacked labels. We first report an experiment where variable-length packets contain one label each, used for routing over a 1×4 tree switch. In the second experiment, the packets carry two time-stacked labels used for 2-hop routing. Successful packet routing and error-free transmission achieved for both experiments verifies the outstanding performance of the proposed systems.

5.2. Principle of Operation

5.2.1. Architectural Implications

We discussed using spectral amplitude codes as labels in chapter 4. In our second proposal presented here, we again compose the optical label with a collection of wavelength bins. As opposed to the first method, here we require no special code properties. We use bit-parallel multi-wavelength labels, where label bits are mapped to separate wavelengths and 0/1 state of a bit turns off/on the corresponding wavelength, *i.e.*, binary addressing (Figure 5.1) [8,12,15]. This approach provides a rich set of labels, as L wavelength bins yield 2^L different labels.

The label and payload wavebands are orthogonal, thus, they can be transmitted simultaneously, as illustrated in Figure 5.1. This results in higher throughput, as the label does not extend the packet duration. The label rate is low compared to payload bit rate; *e.g.*, the labels rate is equal to the packet rate in Figure 5.1 (a) (the label can also occupy a shorter interval as in Figure 5.4 and Figure 5.7). For instance, the optical packet duration for a 1000-bit payload at 10 Gb/s is 0.1 μs. For this size packet, the label rate is as low as 10 Mb/s. Note that the label bit rate is independent of the number of label bits because of parallel transmission of label bits. In other words, by label bit rate we mean the signaling speed of a single wavelength channel. Due to the low-bit-rate label, slow-speed

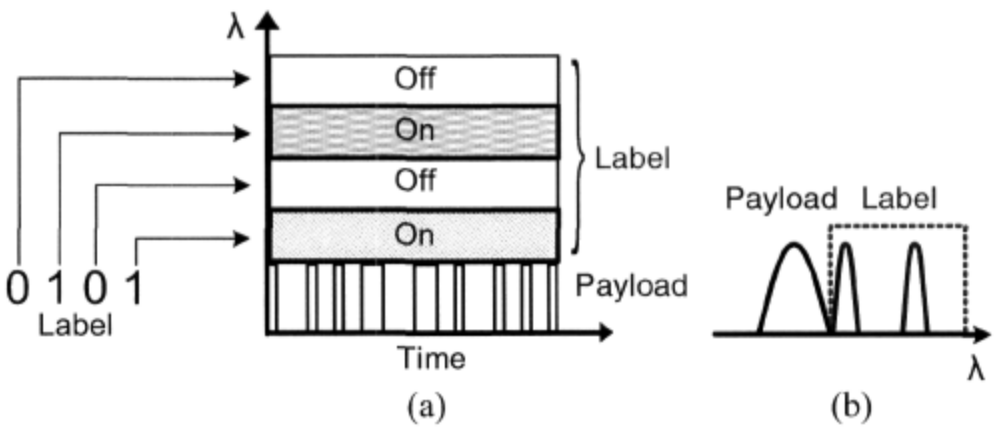

Figure 5.1. Optical packet with binary M-λ label, represented in (a) time domain and (b) wavelength domain.

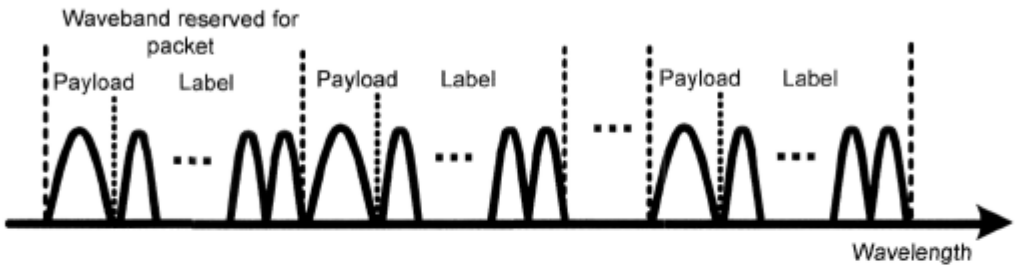

Figure 5.2. Wavelength allocation strategy for M-λ labels.

components can be employed for label generation and recognition.

In this scheme, scalability (*i.e.*, increasing the number of labels leading to more users with moderate increase in hardware complexity) of the network can be accomplished at the expense of spectral efficiency by extending the available bandwidth. As illustrated in Figure 5.2 the spectrum can be divided into orthogonal wavebands each subpartitioned into label and payload bands. Packets that are assigned to different wavebands can be transmitted simultaneously through the same fiber link. The network can be considered as several independent layers, each operating at a specific waveband. This scenario is also efficient in resolving contention by wavelength conversion.

Hierarchical addressing can be easily implemented with M-λ labels. Orthogonal wavebands can be assigned to different label sections corresponding to various network domains. For

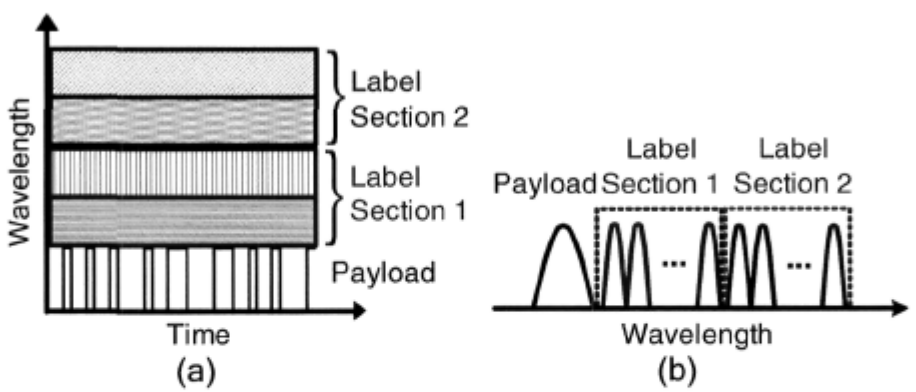

Figure 5.3. Optical packet with binary M-λ label when distinct wavebands are assigned to different network domains for hierarchical addressing. The packet is represented in (a) time domain, (b) wavelength domain.

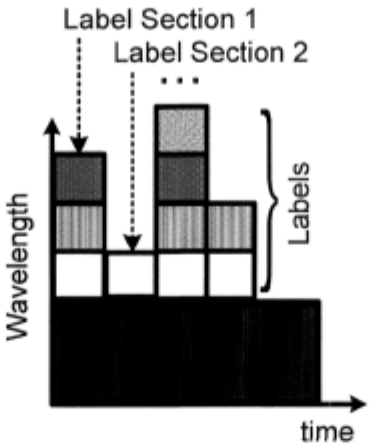

Figure 5.4. Optical packet with binary M-λ labels, where the labels are stacked in time.

example, a two section label is illustrated in Figure 5.3. In each network domain the related label section is easily extracted using optical filters, leaving the remaining label section unchanged. The extracted label section is used for forwarding and is subject to swapping when generating an outgoing label for the next hop. When a packet leaves a network domain, the egress router removes the corresponding label section.

Time domain stacking of binary M-λ labels is another approach for implementing hierarchical addressing, as illustrated in Figure 5.4. In this method, a time slot is assigned to each network domain. The label in the leading time slot is used for forwarding within one domain. This first label is swapped as needed within this domain and the packet forwarded until the packet exits the first domain. At the border of a network domain the leading label is removed and the rest of the label stack is moved forward by one time slot. This method trades-off improved spectral efficiency against increased hardware complexity (see section 5.2.2 (b)). Given the low label rate for a single label, increasing this rate even ten fold (for a stack of 10 labels) is a reasonable trade-off.

Hierarchical addressing provides several advantages. First, it reduces the number of required labels, meaning fewer label bits and thus better spectral efficiency. At each node, a shorter label is processed for the forwarding decision. Therefore the complexity and size of the required hardware will be reduced. For example, in the case of multi-stage switch at the router, the number of stages will be decreased. As each stage introduces delay, fewer stages

imply less overall delay. Finally, the main advantage of hierarchical addressing, *i.e.*, scalability, is achieved by supporting very large number of connections.

5.2.2. Hardware

Having discussed the architectural considerations, we focus on hardware implementations in this section. We will see that cost efficient sources such as broadband sources or directly controlled lasers can be used for the label generator. Furthermore, an ultra-fast scalable forwarding node can be constructed from simple and commercially available components.

a) Sources

The required sources for binary M-λ labels are low-cost compared to PSK labels [6,7] where mode locked lasers (MLL) are mandatory. We can use a broadband source, a multi-

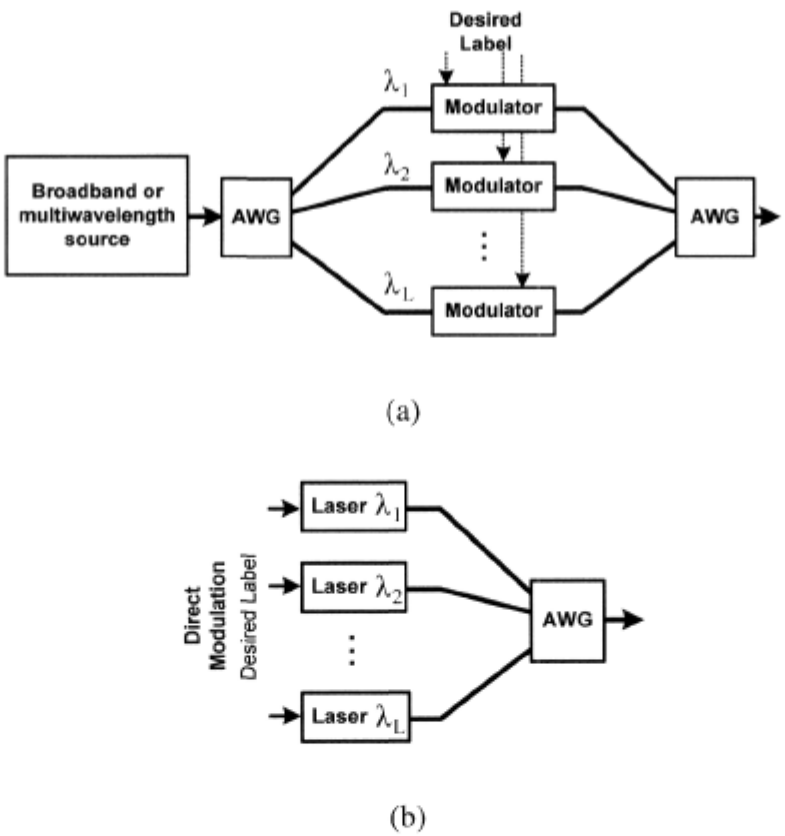

Figure 5.5. Binary M-λ label generator using (a) broadband source or multi-wavelength laser and (b) directly-controlled lasers.

wavelength laser, or directly controlled distributed feedback (DFB) lasers. Due to the low bit rate label, a broadband source can be used without being limited by intensity noise. Note that intensity noise is proportional to B_e / B_o, where B_e and B_o are electrical and optical bandwidth of the label respectively (see section 4.4) [17,57,59]. A narrowband (small B_e) low-pass electrical filter leads to very low intensity noise over the label. In Figure 5.5 (a), an AWG slices (separates) the spectrum of the broadband source (or equally well multi-wavelength laser). One modulator per output turns on/off the wavelength bins according to the desired label. The modulator outputs are combined onto the outgoing fiber using another AWG or a coupler.

Alternately, we can use a bank of directly-modulated single-wavelength DFB lasers, as illustrated in Figure 5.5 (b). Direct modulation of lasers is possible for bit rates lower than 2.5 Gbps; well above the prospective label rate of our system. Both implementations can easily generate any desired label pattern at variable bit rates.

b) Forwarding node

At the intermediate node of the optical network, label extraction is accomplished by passive optical filtering; here an AWG separates the label wavelengths. Each label wavelength controls one switch stage of a multi-stage switch, automatically directing the packet to the

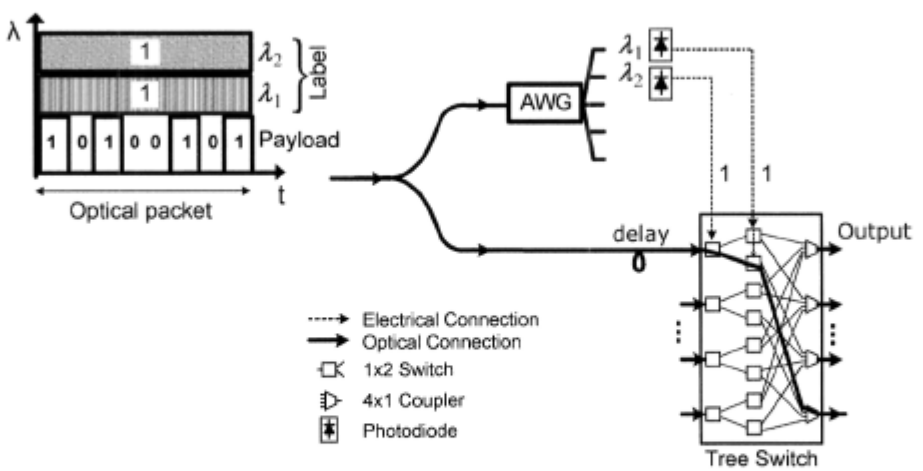

Figure 5.6. The schematic of the forwarding node with directly-controlled multi-stage tree switch for binary M-λ labels.

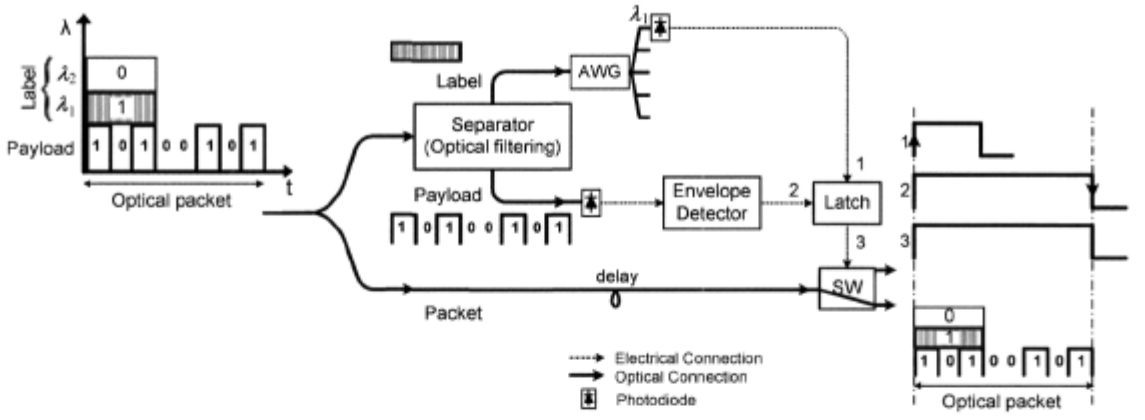

Figure 5.7. The schematic of the forwarding node for binary M-λ labels when the label occupies only a portion of the packet interval.

proper output, as shown in Figure 5.6. In contrast to the correlation based schemes where a label is split *N* times for label processing in an *N* port node, the label power loss in our approach does not vary with *N* but is fixed and equal to the loss of the AWG. The payload is delayed in a fiber delay line to be synchronized with the control signal at the switch. It is switched without being processed; therefore, the system is transparent to the bit rate and the format of the payload. The packet forwarding over the node is on-the-fly; the speed corresponds to the traveling time of light through optical components such as AWG, plus the detection time of photodiodes (PD), if an electronically controlled switch is used.

The label can be modulated either at the packet rate or higher. In the first case, the label occupies the entire packet interval (such as the packet in Figure 5.6). The control signal in this case is constant over the packet interval and can gate to the desired forwarding output; therefore, this format supports variable-length packets "natively"*. In the second case, the label occupies only a portion of the packet interval instead of the entire interval (*e.g.*, in

---

* The guard time between the label and payload is determined by the rise time of the control signal (see Figure 3.18). Although the packets pass through multiples stages, the guard time will not be multiplied by the number of stages. This is because the label wavelengths are detected in parallel as opposed to the WRS discussed in section 3.6.1.

time-stacked labels of Figure 5.4). To be compatible with variable-length packets, an envelope detector for the payload is required in the forwarding node.

The schematic of the forwarding node is shown in Figure 5.7. In this figure we show only one stage for clarity; a complete setup based on the same structure is shown in Figure 5.14. The payload and label are separated by optical filtering. The packet length is recognized by sending the payload to an envelope detector. The detected envelope is fed to an electronic circuit called *latch* in Figure 5.7. The latch has two inputs (port 1 and 2) and one output (port 3). By sending the detected label and the envelope to input 1 and 2 of the latch, we obtain a control signal with the same width as the packet. Output 3 is turned on by the rising edge of the signal of input 1 and is turned off by the falling edge of the signal of input 2; no signal is generated at the output when the label is off. Therefore, variable-length packets can be switched. The packet is time aligned with the control signal using fiber delay line and is forwarded to the proper output according to the control signal. This architecture is practical for time-stacked labels discussed in section 5.2.4. Note that whether exploiting a single label, or a time stacked label, the node can be implemented with low-cost and commercially available optical components, and simple electronic hardware.

c) Tree switch

The switch topology exploited in the forwarding node is a tree (Figure 5.6) topology. An $N \times N$ tree switch has $L = \log_2 N$ stages and is constructed from $N(N-1)$ 1×2 switching elements (SE). The tree switch has the advantage of scalability, *i.e.*, the switch dimension can be easily enlarged by increasing the number of stages.

The SEs may be controlled electronically or optically. For electronic control, one PD per label bit is required at the AWG outputs, *i.e.*, $N \log_2 N$ PDs for an $N \times N$ switch. When using optically controlled switches, the optical label bits directly control the switch [8,14,27]; no PD is required. When optically controlled SEs are used the label bit controlling the i^{th} $(1 \le i \le L)$ stage is split by 2^{i-1} (2^{i-1} SEs are present in the i^{th} stage, each requiring a copy of the control signal). The power of the optical control signals must regulated/amplified, despite the varying splitting levels, to attain the required power for the

switching element used. As optically controlled switches require relatively high power, it is likely optical amplifiers will be required. Some examples of optically-controlled switches are: one based on SOA-MZI used in [14] for packet switching with weight-2 labels and FWM processing (see section 3.5.3), and the wavelength recognizing switch discussed in section 3.6.1 [9].

The latency of a switch is defined as the number of SEs the packet should traverse before being delivered to the output. For the tree switch, this metric is fixed, equal to the number of stages of the switch which is $L = \log_2 N$. In a crossbar, the packet passes only one connection point, so the latency is one unit. In the case of the data vortex switch described in section 3.6.2, the latency is not fixed, as the packets go over various number of internal hops due to deflection routing within the switch, resulting in the undesirable loss of packet order. In the data vortex, the average latency increases with the traffic load, as illustrated in Figure 3.22 [47]. For instance, the expected number of hops for a fully loaded vortex with 2000 ports is about 20.

5.2.3. Contention Resolution

A tree switch is internally non-blocking, similar to a crossbar, as packets injected to different inputs never share a path throughout the switch. However, two packets are in conflict if they are destined to the same output port. For the structure proposed in this chapter we do not consider contention resolution. We assume that only one of the conflicting packets is delivered by the switch and the rest are discarded. Contention resolution techniques discussed in section 2.3.3, such as wavelength conversion or deflection routing are applicable for the proposed forwarding node.

Here, we evaluate the switch performance when no contention resolution is used, in terms of the ratio of successfully delivered packets. The bandwidth (BW) of a switch is defined as the average number of packets delivered per cycle. The bandwidth of an $N \times N$ crossbar is given in [60] as

$$BW = N - N\left(1 - \frac{p}{N}\right)^N,$$

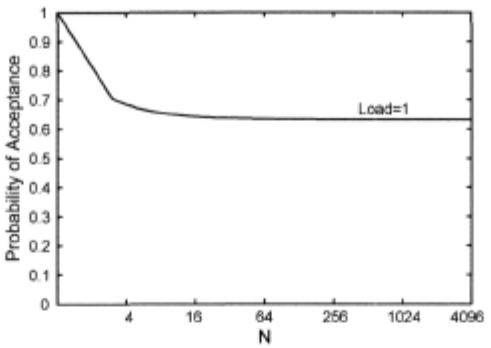

Figure 5.8. Probability of acceptance versus number of ports for a fully loaded tree switch.

where p is the traffic load, *i.e.*, the average number of packets per cycle. The same equation is valid for the tree switch as it has the same blocking characteristics as a crossbar. The bandwidth equation is obtained assuming that packets of different cycles and different inputs are independent and are uniformly distributed over the output ports. The probability of acceptance, defined as the ratio of BW to the expected number of requests (Np) can be obtained by

$$P_A = \frac{1}{p} - \frac{1}{p}\left(1 - \frac{p}{N}\right)^N .$$

The probability of acceptance as a function of the switch size is plotted in Figure 5.8 for maximum load ($p = 1$). As illustrated in Figure 5.8., more than 60 percent of the packets are delivered successfully even for large scale switches.

We discussed the data vortex in section 3.6.2 where contention is resolved by deflection routing inside the switch. The acceptance rate of data vortex was shown in Figure 3.21, which is 26.1 % for a switch with 640 ports [47]. It is shown in [51] that high throughput can be obtained at the cost of larger number of components. For instance a vortex with 2048 ports and maximum traffic load maintains nearly 100% acceptance rate if the angle size is six; that is six times more elements than those required to implement the switch (see section 3.6.2). Hence, exploiting a multistage tree architecture brings us a reasonable trade-off between cost/complexity and probability of acceptance without recourse to additional hardware for contention resolution.

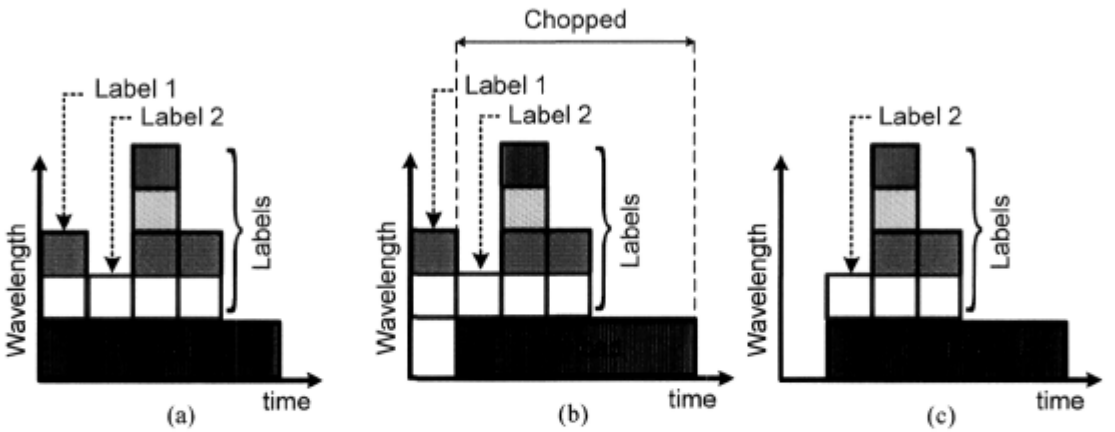

Figure 5.9. Optical packet (a) with time-stacked multi-wavelength labels, (b) in the first transit node where the payload is time aligned with label 2, and (c) at the output of the first transit node where label 1 is removed and label 2 leads the label stack.

5.2.4. Time-Stacking Labels versus Label Swapping

In section 5.2.1, we discussed time stacking labels for hierarchical addressing, where non-overlapping time slots are assigned to different network domains. Alternatively, a stack of labels could be used within a domain to indicate the hop-by-hop labels for routing within the domain. In this case each node need only "push" the stack to access the next label to be transmitted to the next node. Contrast this with the typical GMPLS routing procedure: the label corresponding to the network domain is extracted from the incoming packet and this label is subject to swapping before forwarding. That is, the label value is found in a lookup table and the corresponding outgoing label is identified. In an optical network, this operation requires optoelectronic conversion of the label, and an electronic, time-consuming table lookup (despite the reduced table size in GMPLS). We describe in this section the relative merits of hop-by-hop time stacked labels versus label swapping.

We propose to shift the burden of sophisticated processing to the smart electronic edge nodes and benefit from a simple, practical, economical, and ultra-fast optical backbone. In our scheme, the path from the source to the destination on the optical domain is determined at the edge nodes, and accordingly, all the labels required for the intermediate hops are attached to the packets in non-overlapping time slots (Figure 5.9 (a)). Note that in contrast to the scheme discussed in section 5.2.1, here we do not swap labels; instead, n time-

multiplexed labels are attached to the payload for an n-hop path*. At each hop, the leading label in the stack is used for forwarding and is removed when leaving the node; the rest of the labels are pushed ahead one label interval (Figure 5.9 (b), (c)) [63]. The label processing requires only simple and low-speed electronic hardware for forwarding multi-rate and variable-length payloads as explained in section 5.2.2 b.

The labels are transmitted in fixed-length time slots in chronological order; that is, the first label is used at the first hop and so on. The packet duration is greater than or equal to the number of labels in a stack times the slot interval. At the forwarding node, the label and payload are separated by optical filtering. The payload is delayed one label time slot by fiber delay line and is combined with the label again. Separating the leading label and the remaining packet is achieved by using an 1×2 optical switch. The signal used for controlling the switch is obtained by an envelope detector. The envelope is used in the same manner as Figure 5.7 for switching the variable-length packets.

We have examined the implementation of the typical GMPLS routing procedure with an optical label swapper developed by researchers at McGill University. In appendix A, we report a joint experiment [53] where the McGill label swapper was integrated into our binary M-λ labels and multi-stage switch. This optical label swapper for M-λ labels is based on a semiconductor fiber ring laser [21]. In the joint experiment we assumed that the new label is *fixed* and *predetermined* and we did not implement a lookup table. In a real system, once the new label is determined, the label swapper needs to be tuned to the new label; the tuning speed should be equal or greater than the packet rate (in the range of megahertz). This requires fast tuning the FBGs used in the ring laser structure; typical FBG tuning requires milliseconds, using stretching or thermal tuning [61]. Hence, the use of time stacked labels gives a moderate increase in complexity, but avoids serious technology challenges for label swappers.

---

* It is shown in [62] that the average number of hops in the Internet is ~15.

5.2.5. Conclusion

In this section, we proposed a practical, high-speed, and simple multi-hop optical network to accomplish the objectives 1-5 discussed in section 1.2. We proposed binary M-λ labels due to their potential for high-speed packet forwarding with low-complexity hardware. The label encoding method provides high cardinality; L wavelength bins gives 2^L labels. Increasing label cardinality only requires more lasers and an AWG with more output ports. It is compatible with hierarchical addressing by wavelength- or time-stacking labels. Label processing is on-the-fly, the speed corresponds to the traveling time of light through the AWG plus PD detection time. One advantage of this technique is fixed label power loss at each node, as opposed to the correlation based techniques where the loss scales with the number of labels. Furthermore, the network is scalable due to the rich set of labels and also the structure of the forwarding module using a multi-stage tree switch. Other advantages of the tree switch includes internally nonblocking, fixed latency, and simplicity. The label processing requires only simple and low-speed electronic hardware for forwarding multi-rate and variable-length payloads.

5.3. Comparison with Other Solutions

In this section we will compare multi-stage switching using binary M-λ labels with other solutions, in terms of spectral efficiency, power loss, and cost. We consider the methods discussed in chapter 3, namely, the compact decoder for PSK labels, parallel decoders for SAC-labels, FWM processing for weight-2 codes, and the data vortex.

a) Spectral efficiency

Binary M-λ labels (also used in the data vortex) are more spectral efficient than SAC-labels and also weight-2 codes, given the same waveband; binary M-λ labels also provides the highest cardinality (2^L for L bins). Recall chapter 3 that the number of labels for BIBD SAC-labels and weight-2 codes is L and $\binom{L}{2}$, respectively. Spectral spacing between label wavelengths determines the required spectrum, *e.g.*, channel separation of the AWG. For example, given $L=6$ (required for a 64×64 node) and 25 GHz AWG channel spacing,

1.2 nm bandwidth is required at 1550 nm for binary M-λ labels. This number reduces to 0.6 nm for an AWG with 12.5 GHz channel spacing [64]. The proposed system bandwidth requirements are much lower than previously proposed schemes based on SAC-labels (9.6 nm gives 7 codes) [54] or PSK codes discussed in section 3.4.3 [44]. In [44], 2.5 psec pulses occupy a 3.2 nm optical bandwidth at 1550 nm. In the case of PSK labels, the required waveband does not increase with the number of labels; it is determined by the spectrum of picosecond pulses generated by mode locked laser.

b) Power loss

Label

One of the main advantages of multi-stage switching with binary M-λ labels is fixed label power loss, equal to the AWG insertion loss (note that, when optically controlled SEs are used the label bit controlling the i^{th} $(1 \le i \le L)$ stage is split to 2^{i-1}). Whereas, the label is split to $N = 2^L$ in correlation-based label processing such as parallel decoders for SAC-labels, and also compact decoders for PSK labels. In FWM processing of weight-2 labels, the power of the label identifier varies with the spectral space between the two wavelength bins (in the case of cardinality 36, the spacing between the bins can vary from zero to 35 bins [14,46]). In data vortex, the SEs are constructed by SOAs, therefore, the power losses are compensated by SOA gain.

Payload

The number of SEs that the payload traverses in the tree switch is $L = \log_2 N$ resulting in power loss proportional to $L_s^{\log_2 N}$, where L_s is the insertion loss of an SE. At the output of the tree switch, the signal from N SEs are coupled to the same port (see Figure 5.6); therefore, the total power loss for the payload in the tree switch is proportional to $N(L_s^{\log_2 N})$. As for the crossbar switch used for PSK labels, SAC-labels, and weight-2 codes, the payload power loss is proportional to $N^2 L_s$. That is due to the payload passing only one switching element (loss= L_s) and two $1 \times N$ couplers (loss= N^2) at the input and output of the switch. For instance, the insertion loss of the Eospace 1×2 SEs available in

our laboratory is about 3 dB. For this L_s, the power loss of the tree topology is 3 dB less than that of the crossbar. In the case of smaller L_s, which is typical for optical switches, the tree would have less payload power loss than the crossbar. In the case of data vortex, the power loss is compensated by SOA gain.

c) Cost

The number of crossovers used for constructing the electrical switch typically impact the cost efficiency of switches. An $N \times N$ tree switch is constructed by $N(N-1)$ SEs. The number of SEs in an $N \times N$ crossbar is N^2. For a $HA \times HA$ data vortex (where $N = HA$), the number of SEs equals $HA(\log_2(H)+1)$ (see section 3.6.2).

Table 5.1. Component count for implementing forwarding nodes using different strategies. N-SOA: nonlinear-SOA, O-Filter: optical filter

Method	Number of Equipments
Binary M-λ labels using tree switch [12]	**PD (slow-speed):** $N \log_2 N$ **1×2 SE:** $N(N-1)$ **AWG with *N* channels:** N
PSK-labels using compact decoder [7,44]	**PD (high-speed):** N^2 **1×1 gate:** N^2 **Compact decoder:** N **T-flip-flop:** N^2
SAC-labels using parallel decoders [13]	**PD (slow-speed):** N^2 **1×1 gate:** N^2 **Decoders:** N^2
Weight-2 codes using FWM processing [14]	**PD (slow-speed):** N^2 **1×1 gate:** N^2 **N-SOA:** N **AWG with** $> N$ **channels:** N (Example: 88-channel AWG for 36 codes)
Data vortex [15]	**PD (slow-speed):** $2HA(\log_2 H+1)$ **O-Filter:** $2HA(\log_2 H+1)$ **SOA:** $2HA(\log_2 H+1)$ **Control logic:** $HA(\log_2 H+1)$

However, to make a fair comparison between the cost of the forwarding nodes, other building blocks should also be taken into consideration. We have summarized the component count for different architectures in Table 5.1. The binary M-λ labels using the tree switch is practical and cost-effective as it can be implemented with simple and low-cost equipments.

For example, the number of photodiodes required for the tree switch is $N(\log_2 N)$, *i.e.*, one per stage. If we use optically controlled SEs no photodiode is required. For crossbar, this number equals N^2 (one per SE), and for data vortex $2HA(\log_2(H)+1)$ (two per SE).

Another key building block of the forwarding node for binary M-λ labels with the tree switch is an AWG; N AWGs each with $\log_2 N$ ports are required (one AWG per input). As for the SAC-labels, each SE is controlled by a correlator, therefore, N^2 correlators are used totally. PSK labels and weight-2 codes are more efficient than SAC-labels, in terms of component count. One decoder for each input is needed for PSK codes, *i.e.*, N decoders overall (note that contrary to AWGs, the compact decoders are not commercially available and require special design and fabrication). In the case of weight-2 codes, N decoders implements the node and each decoder includes an nonlinear SOA (N-SOA) and an AWG. The data vortex employs two optical filters (o-filter) and two SOAs in each SE, which gives $2HA(\log_2(H)+1)$ o-filters and the same number of SOAs. It also uses $HA(\log_2(H)+1)$ logic circuits for deflection and control signaling (one per SE).

5.4. Experimental Demonstration

The experimental proof of the concept for binary M-λ labels with multi-stage switching structure is provided in this section. Here we report two sets of experiments; in the first case (section 5.4.1) we demonstrate generation and routing of variable-length packets where only one label (modulated at packet rate) is assigned to each packet. In the second case (section 5.4.2), we implement the forwarding node capable of routing packets with time-stacked labels. We examine routing and BER performance of the packets in both systems. The details of the experimental setups and the results are provided in the following sections.

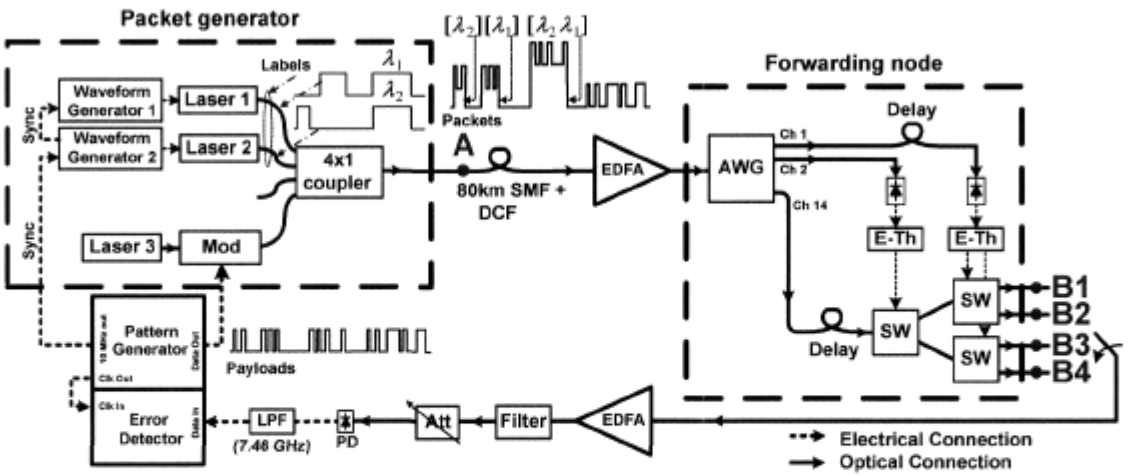

Figure 5.10. Experimental setup of binary M-λ label switching network; each packet includes a single label modulated at packet rate. Mod: modulator, SMF: single mode fiber, DCF: dispersion compensating fiber, EDFA: Erbium doped fiber amplifier, AWG: arrayed waveguide grating, E-Th: electrical thresholder, SW: 1×2 switch, Att: attenuator, PD: photodiode, and LPF: low pass filter.

5.4.1. Experimental Setup of Single Label per Packet

In the first experiment, we generate variable length packets with 2-bit labels (L=2 and N=4), transmit over 80 km of fiber, and optically route packets with a 1×4 tree switch. The power loss in label processing is fixed, 8 dB in this experiment, equal to the insertion loss of the AWG. The payload data rate is 10 Gbps, the available test equipment, and can be easily scaled up to 40 Gbps. The proposed label processor could be compact and integrated with either an electronically or optically controlled multi-stage switch.

The experimental setup is illustrated in Figure 5.10, which is composed of two major parts: a packet generator and a forwarding node. At the packet generator two directly-modulated DFB lasers (laser 1 and 2) are used for the label bits where the presence or absence of these lasers defines four different labels. The wavelength of the lasers, corresponding to the least and most significant label bits ($[\lambda_2, \lambda_1]$), are λ_1=1549.72 nm and λ_2=1550.12 nm respectively. The DFB lasers are modulated with a zero-bias waveform, and biased with a reverse voltage determining the laser output power. The center wavelength of the lasers varies by temperature; center wavelengths are adjusted via a temperature controller. The bias voltage and modulating signal amplitude also have a slight impact on laser

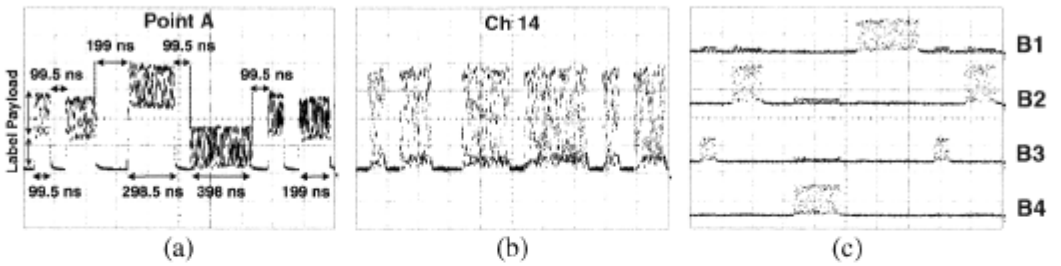

Figure 5.11. Oscilloscope traces of (a) generated packets, (b) payloads separated by channel 14 of AWG, and (c) switched packets of the setup of Figure 5.10.

wavelength. We use two programmable waveform generators to modulate the lasers. The waveform generators are synchronized with the payload pattern generator using the 10 MHz output.

The pattern generator is programmed with a pattern containing four packets of 2^{10}-1 PRBS data and a silence period between packets. This pattern drives a Mach-Zehnder modulator at 10 Gb/s, modulating a laser at 1554.94 nm. The packets are 99.5, 199, 298.5, and 398 ns long, containing 995, 1990, 2985, and 3980 payload bits, and [1,0], [0,1], [1,1], and [0,0] labels respectively. The label and payload signals are coupled to form the packets as illustrated in Figure 5.11 (a); the label is identifiable as an offset added to 10 Gbps payload. The time alignment of labels and payload is achieved by monitoring the output of the 4×1 coupler and adjusting the phase of the waveform generators. The packets are then transmitted through 80 km single mode fiber (SMF) plus dispersion compensating fiber (DCF) to emulate the impairment of transmission.

At the forwarding node, the label wavelengths are separated by an AWG, each to control one stage of a tree switch. The AWG has 16 channels with 50 GHz channel spacing, as illustrated in Figure 5.12. The center frequency of the channels can be tuned by a temperature controller to match the label wavelengths. Channel 1, 2, and 14 are used to separate the label bits and payload respectively. The labels are detected by slow-speed photodiodes with the operating frequency matched to the packet rate. The detected labels are fed to electrical thresholders providing the required voltage (+13 V) to control the switches. The combination of three $LiNbO_3$ 1×2 optical switches with switching time less than 100 ns constructs an 1×4 tree switch. The switches are manufactured by Eospace with

insertion loss of about 3 dB. They route the input signal to the upper (lower) output at the absence (presence) of the electrical control signal.

The payloads, separated from their labels by the AWG, (see Figure 5.11 (b)) are sent to the input port of the switch. The alignment between payloads and control signals at the switching elements is achieved by fixed fiber delay lines. Each packet is forwarded to one of the four outputs according to its label. The successfully routed packets at points B1, B2, B3, and B4 of the setup are shown in Figure 5.11 (c).

We measure the BER versus the average received power of the payload after being switched (point B1 in Figure 5.10) and also after the pattern generator (point A in Figure 5.10). We use an optical band-pass filter of 0.2 nm before detection to filter out the amplified spontaneous emission (ASE) of amplifiers (and labels at point A). An electrical low-pass filter with a pass-band equal to three quarters of the bit rate (7.46 GHz) is used after the photodiode to remove out-of-band noise.

In this experiment, we faced several technical challenges for BER measurement that are explained in the following. Firstly, the clock of the error detector should be aligned with the data input such that the sampling point is in the middle of the data bit. We do not use a clock recovery module in our setup and the clock output of the pattern generator feeds the clock input of the error detector. However, the packets are received by the error detector after traversing 80 km of fiber. The temperature variations of the environment change the

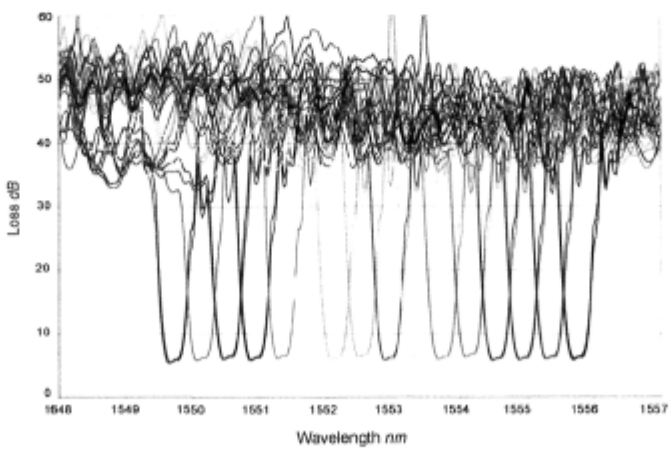

Figure 5.12. Spectrum of the 16-channel, 50 GHz AWG used in the setup of Figure 5.10.

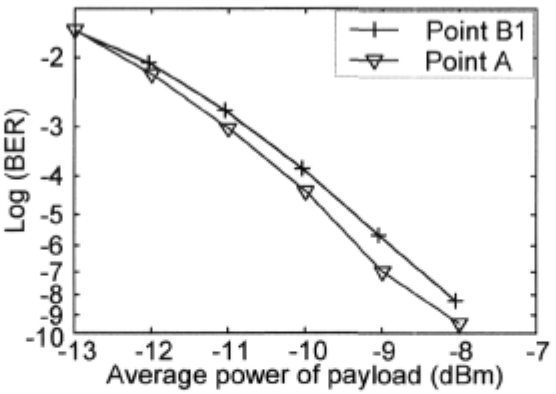

Figure 5.13. Bit error rate versus average received power of payload at point A and B1 of Figure 5.10.

refractive index of the fiber and, therefore also change the traveling time of the light and the optimum sampling time. For that reason, we manually adjust the optimum sampling time for each point of the BER curve instead of using the same sampling point for all the BER measurements.

Secondly, the bit error rate tester must be synchronized with the pattern generator; the synchronization is achieved by comparing the generated pattern and the received pattern. However, only one packet is received at point B1, whereas the pattern generator generates four. The remaining packets are routed to output ports B2-B4. We attain synchronization by disconnecting the control ports from the switches so that all the four packets are routed to B1; when the error detector is synchronized, we reconnect the control ports.

Note that the BER is plotted as a function of the average power of the payload in Figure 5.13. However, at point A, the power measured by the power meter is the average power of four packets, instead of only one. At point B1, the leakage of power from the other ports is added to the payload power (see Figure 5.11 (c)). Therefore, to measure only the average power of the payload, we monitor the signal of point A and B1 by an oscilloscope. We use the histogram of the oscilloscope and adjust the window over the entire period of four packets to measure the mean voltage (V_1). We next adjust the window to include only the desired payload (longest one here), and measure the mean voltage again (V_2). We use the ratio of these two numbers (V_2 / V_1) to normalize the power obtained by

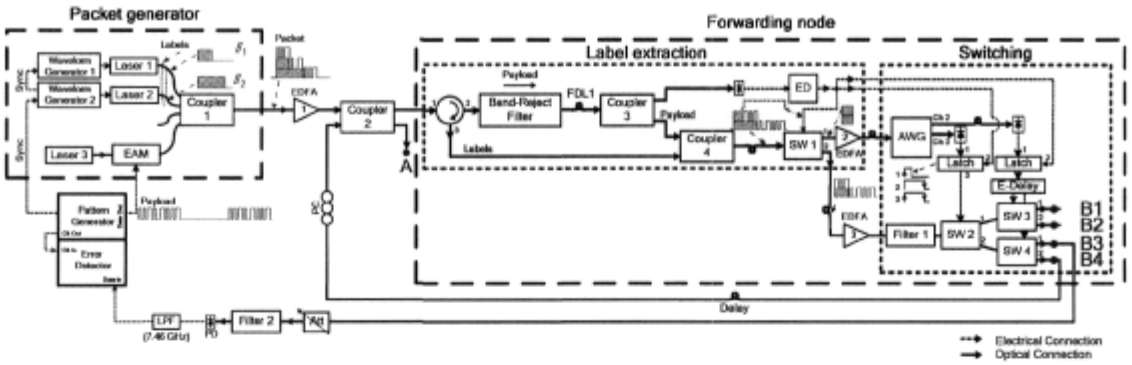

Figure 5.14. Experimental setup of the packet switching network with time-stacked binary M-λ labels; EAM: electro absorption modulator, EDFA: Erbium doped fiber amplifier, FDL: fiber delay line, ED: envelope detector, SW: 1×2 switch, AWG: arrayed waveguide grating, PC: polarization controller, Att: variable attenuator, PD: photodiode, LPF: low-pass filter.

the power meter (P_m) and calculate the actual average power (P_a) of the payload ($P_a = P_m \times V_2 / V_1$).

We measure the BER at points A and B1 of Figure 5.10 to evaluate the performance of the proposed system. We filter out the label and measure the BER of payload bits using an error detector. The error detector is equipped with block BER option. In this mode we are able to specify the location of the bits in the payload for BER measurement. The BER curves correspond to 1984 bits in the middle of the longest packet. Figure 5.13 presents BER at points A and B1 of Figure 5.10. The BER performance degradation after transmission and switching is low, as shown by the power penalty between back-to-back transmission (point A) and after switching (point B1) that is 0.5 dB at 10^{-9} BER.

5.4.2. Experimental Setup of Time-Stacked Labels

In this experiment, the optical packets contain two time-stacked labels, each label has two bits, *i.e.*, two wavelengths. The experimental setup consists of a packet generator and a forwarding node as illustrated in Figure 5.14. The first label contains routing information for the first hop. After successful routing, our setup recirculates the packet back to the input of the node. At this point, the second label will be processed to route the packet through its

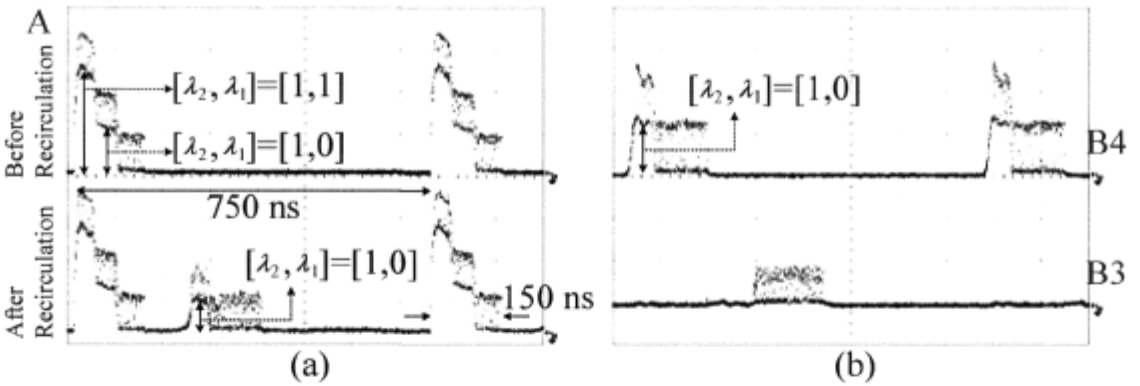

Figure 5.15. Oscilloscope traces of (a) point A before (upper trace) and after (lower trace) recirculation, and (b) point B3 and B4 of Figure 5.14.

second hop. The BER is measured after the packet generator, hop 1 and hop 2 to evaluate the performance of the switching structure.

In Figure 5.14, we generate packets of length 150 ns every 750 ns. Two directly modulated DFB lasers at λ_1=1549.43 nm and λ_2=1549.85 nm are used for the labels. The lasers are driven by programmable waveform generators to form time-stacked labels of length 50 ns. Label 1 and label 2 are [λ_2, λ_1]=[1,1] and [λ_2, λ_1]=[1,0] respectively, which correspond to output 4 and 3 of the switch (Figure 5.15 (a)). The payload contains 1500 bits of 2^{11}-1 PRBS data at 10 Gbps, generated by externally modulating a laser at 1552.25 nm. At the forwarding node, the labels and payload are separated by a band-reject filter which reflects the labels and transmits the payload (see Figure 5.16). The band-reject filter is implemented

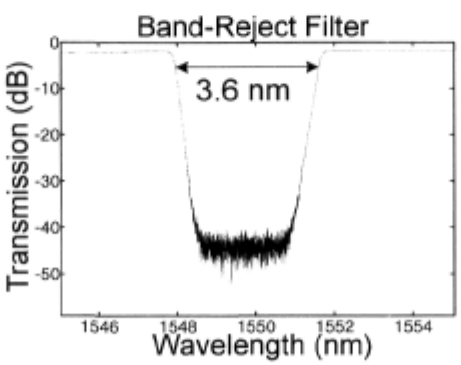

Figure 5.16. Transmission of the band-reject filter used in the setup of Figure 5.14.

by a fiber Bragg grating (FBG) with insertion loss of 2 dB, 3 dB bandwidth of 3.6 nm, and reflectivity of ~99.9%. The payload passes through a fiber delay line (FDL1) for a label interval (50 ns) and is split by coupler 3; the signal on the upper arm is sent to an envelope detector (ED) and the signal on the lower arm, now delayed by one label time slot, is re-coupled with the labels (see Figure 5.9 (b)). The envelope detector output is used as a control signal to drive a 1×2 optical switch (SW 1). The delay following coupler 4 is selected to ensure that the ED control signal cleanly slices the top-of-the-stack label and switches it to output 1 of SW 1, while the payload and newly shifted stack is switched to output 2. The label (output 1 of SW 1) is sent to an AWG to separate the label bits. The AWG has 16 channels with 50 GHz channel spacing and 8 dB insertion loss (Figure 5.12). The label channels trigger an electronic latch whose other input is connected to the ED control signal. The latch is triggered by the rising edge of the label and holds until the falling edge of the envelope (Figure 5.14). Therefore, the latch generates a control signal with the same width as the packet, whenever the label bit is present. This signal sets the state of the SEs. Three 1×2 SEs (SW 2, 3, and 4) form a 1×4 tree switch. The SEs direct the input to the upper (lower) output when the control signal is off (on). The time alignment between the incoming packet and the control signals is achieved by fiber delay lines and a fixed electronic delay (E-Delay). Filter 1 (1.2 nm) is used to remove the ASE of the optical amplifiers.

The packet is self-routed to output 4 by label 1 ([1,1]) and is recirculated to the input for the second hop. Next, label 2 ([1,0]) forwards the packet to output 3. We measure the BER of 1280 bits in the middle of the payload, after the packet generator, the first, and the second hop. Filter 2 (0.2 nm) is used to remove the ASE of the optical amplifiers. An electrical low-pass filter (7.46 GHz) after the photodiode removes out-of-band noise.

5.4.3. Results and Discussion

The oscilloscope traces of the packets at different points of the setup (Figure 5.14) are illustrated in Figure 5.15 (a) and (b). The upper trace in Figure 5.15 (a) shows the packets after the packet generator (point A of the setup), *i.e.*, when the lower arm of coupler 2 is disconnected; the packets contain two time-stacked labels. In the lower trace, the lower arm of coupler 2 recirculates the successfully routed packets for the second hop. Label 1 is

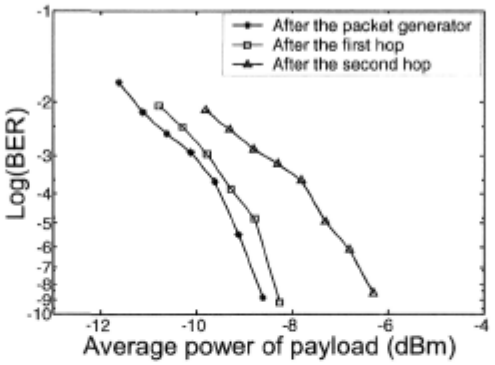

Figure 5.17. BER versus the average power of payload for points A, B3, and B4 of Figure 5.14.

removed from the routed packet, and label 2 is moved up in the stack. The upper (lower) trace of Figure 5.15 (b) demonstrates the packets after being switched for the first (second) time, that is, point B4 (B3) of the setup. The upper packet contains label 2, which is used for the second hop and is ultimately removed from the stack, as shown in the lower trace.

We measure the BER versus the average optical received power of the payload at points A, B3 and B4 of the setup. As illustrated in Figure 5.17, the performance degradation after switching is low, confirmed with the low power penalty between point A and B4. The 2 dB power penalty between hop 1 (B4) and hop 2 (B3) is due to the recirculating loop, which accumulates intensity noise between the first routed payload and the leakage from the electro-absorption modulator (EAM with 11 dB extinction ratio) during the silent period between packets. The polarization controller before coupler 2 is used to reduce the intensity

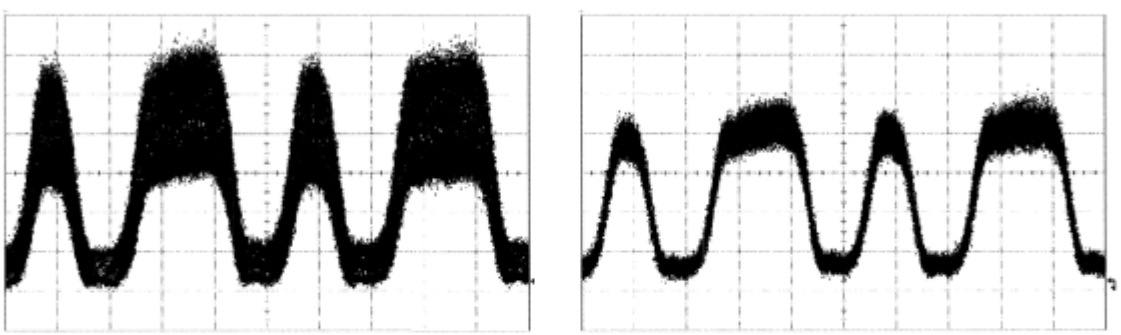

Figure 5.18. Oscilloscope traces of the payload bits after the first hop at point A of Figure 5.14. The intensity noise is reduced from case (a) to (b) by adjusting the polarization controller.

noise. The oscilloscope traces of Figure 5.18 show the payload bits at point A of Figure 5.14 after being switched once and recoupled to the input. The intensity noise is reduced from case (a) to (b) by adjusting the polarization of the two beating signals to be orthogonal.

5.5. Conclusion

In this chapter, we proposed binary multi-wavelength labels due to their potential for high-speed packet forwarding with low-complexity hardware. One advantages of this technique is fixed label power loss at each node, as opposed to the correlation based techniques where the loss scales with the number of labels. Furthermore, the network is scalable due to the rich set of labels and also the structure of the forwarding module using a multi-stage switch. The label processing requires only simple and low-speed electronic hardware for forwarding multi-rate and variable-length payloads. We also proposed a practical network with time-stacked labels to avoid the sophistication of label swapping. We reported two sets of experiments, all with low-complexity, off-the-shelf commercial components. Firstly, the major structure of the multi-wavelength label-switching examined when the packets carry a single label. Secondly, time-stacking of optical label was demonstrated. The successful packet routing and error-free transmission confirmed the outstanding performance of the proposed systems.

Chapter 6

Randomized Scheduling Algorithm for a Practical Optical Network

6.1. Introduction

In chapters 4 and 5 we discussed optical packet switching networks with mesh topology, and proposed solutions for packet forwarding in the intermediate nodes. However, our proposals did not offer contention resolution. In any packet switching network contention arises when more than one packet is destined to the same output port at the same time. The simplest way to resolve contention would be to buffer the contending packets either at the input or output of a switch, similar to electronic switches. Optical buffering using fiber

delay lines was demonstrated in [33]. Due to the sequential access in delay lines, long fibers are required to achieve a reasonable packet loss probability, working against system compactness. Synchronous networks with fixed-length packets require less buffering, but require sophisticated optical synchronizers [32]. Data vortex, discussed in section 3.6.2, [38,51] is a switch structure proposed for synchronous networks where contention is resolved by internal deflection routing in the switch. It requires high component count and precise timing for deflection signaling between the switching elements (see section 3.6.2).

In asynchronous networks with variable-length packets, there is no need for optical synchronizers. However, since the packets can arrive at any time, the probability of contention is higher. In this case a massive amount of fiber delay lines is required for contention resolution. To reduce the required number of fiber delay-lines, or even to resolve contention without any buffering, wavelength conversion was examined in [34]. In this scenario the packets that are destined to the same output at the same time are sent on different wavelengths (section 2.3.3). The problem with this method is that the state of the art optical wavelength converters are still complex and expensive to be used in commercial systems.

Based on the above statements, we conclude that mesh networks are complex to be implemented with the current optical technology, and the main issues are contention resolution and optical synchronization. In this chapter we present a practical, economical, and simple architecture that can be implemented without optical buffers and optical synchronizers [24].

The topology of our network is based on a single-hop structure, having star-like connectivity. The network consists of several edge nodes that are connected through a core node. The core nodes of several stars are interconnected via channel switches to construct a high capacity network. It has been shown in [65] that the capacity of such a network can be as high as petabits per second, much higher than the entire capacity of the Internet today.

Each edge node receives data from several electronic traffic sources and converts the data to the optical format. The edge nodes are connected to the core node by fiber links that can carry traffic on multiple wavelength channels. The core node contains a bufferless

nonblocking $N \times N$ switch, *e.g.*, a crossbar, in which each output can accept one packet and where the other packets contending for a given output are dropped. In this scheme, the required buffer in the network is moved to the edge nodes, where the packets are stored in electronic RAMs before being converted to optical packets. The packets are transmitted in fixed-length time slots. As the structure is single-hop and the fiber links are fixed-length, synchronization can be achieved easily by time locking the edge nodes to the core node, at the time of network establishment. Therefore, there is no need for optical synchronizers.

Contention in a star network can be resolved by assigning time slots to the incoming packets so that no packets appearing at the same time at the core node have the same destination. The edge nodes, in this example, request connection before transmission by informing the core about the state of the packet queues in their buffers. Based on this information, a centralized scheduler at the core uses a conflict-free scheduling algorithm to assign time slots to the packets [66]. This approach is suitable for networks with short fiber links where the propagation delay (d_{prop}) is negligible. In networks with large propagation delay, the centralized approach causes each packet to wait for at least d_{prop} at the edge buffers, resulting in unacceptable delay. Here, we focus on networks with large propagation delays.

A randomized scheduling algorithm is proposed in [67] to achieve packet delay that is small compared to d_{prop}. We use the randomized scheduling algorithm proposed in [67] to achieve contention resolution in the star part of the network. This approach obviates central control to coordinate the edge nodes, and instead minimizes the probability of conflict at the core node. This algorithm trades off reduced throughput for delay, so it is suitable for networks with long fiber links and large d_{prop} such as the proposed network architecture. The algorithm in [67] was conceived with two unrealistic assumptions: the packets are transmitted in frames with infinite length, and the incoming traffic has a property called (α, S)-smoothness. We evaluate the performance of the algorithm when these assumptions are violated. Performance is evaluated via simulation for finite-length frames and a realistic traffic type: Markov-Modulated Bernoulli (MMB) traffic. We further propose a low complexity bipartite graph matching algorithm to be used by the core ([67] assumes a

maximum cardinality matching). Our simulations show that randomized scheduling gives desirable performance even in realistic traffic scenarios. Based on these results we propose improvements to the scheduling algorithm to reduce packet delay, as well as simplify implementation, and reduce throughput penalty.

In section 6.2 we define the (α,S)-smooth traffic model, and explain the randomized scheduling algorithm in section 6.3. Later, in section 6.4 the bipartite graph matching algorithm is described. Simulation results for different types of traffic are provided in section 6.5.

6.2. (α,S)-Smooth Traffic Model

The randomized scheduling algorithm (described in section 6.3) is motivated by the assumption that the traffic is (α,S)-smooth. Traffic that is (α,S)-smooth has fewer than αS packets arriving at any ingress node during any time interval of length S. In addition, no destination node is sent more than αS packets during any interval of length S.

To shape the traffic at the edge nodes, the incoming packets are buffered and no more than αS packets are transmitted in each frame of length S. To ensure that the total number of packets requesting the same output link during the same frame does not exceed αS, we must limit the maximum number of packets allowed to be transmitted from each source node to each destination node. This can be achieved by sending feedback information about the state of the queues of the edge nodes to a central controller that assigns an adaptive bandwidth provision matrix well-matched to the incoming traffic. This last requirement is inconsistent with the major goal of the scheduling algorithm, *i.e.*, achieving delay that is small compared to propagation delay. For this reason we also consider MMB arrivals as an alternative to (α,S)-smooth arrivals, as MMB arrivals are more realistic for a decentralized network.

6.3. Randomized Scheduling Algorithm

A randomized scheduling algorithm is proposed in [67] for star networks with large propagation delay, to minimize collision at the core node and consequently to minimize the

access delay of packets at the edge nodes. The access delay is defined as the number of slots that the packet must wait at the edge buffers before transmission. This algorithm has two phases; a first phase with multiple transmissions of the same packet in a large portion (for example three quarters) of a large frame, and the second phase of conflict-free scheduled transmission in the final portion of the frame. The minimized access delay of this algorithm is achieved at the cost of rather low throughput. In this section we will describe the algorithm in detail.

In the randomized scheduling algorithm, packets are aggregated at the edge nodes and are transmitted in frames of length M slots. Suppose that the traffic is (α, S)-smooth and $\alpha \leq 1/4$. In that case, no more than $M/4$ packets are received at each edge node during a frame. The transmission algorithm has two phases:

a) First Phase

The first phase uses $3M/4$ slots of the frame to transmit packets. In this phase, each packet is transmitted three times in timeslots chosen at random by the edge node. The edge node controllers will notify the core node of the packet locations prior to transmission of the frame. The frames received from different edge nodes may contain packets destined for the same output port at the same timeslot. In this case, only one of the contending packets can be switched to the output port, and the remaining contending packets will be discarded. As each packet is transmitted multiple times, the core node can attempt to recover a different transmission of a dropped packet.

The core node will use the information of the location of incoming packets from the edge nodes to switch packets to egress nodes at each time slot. The core will try to maximize the total number of successful packets. It constructs a bipartite graph $G(U, V, E)$ for each receiver, and finds a maximum match on G (explained in section 6.4). In the bipartite graph, U represents the packets from all the source nodes destined to a given destination node ($|U| \leq \alpha M$, where $|U|$ is the cardinality of the upper vertexes), V represents the time slots ($|V| = 3M/4$), and E (edges) represents the location of packets in a frame. An example of a bipartite graph is illustrated in Figure 6.1 (a). A match on G is shown in

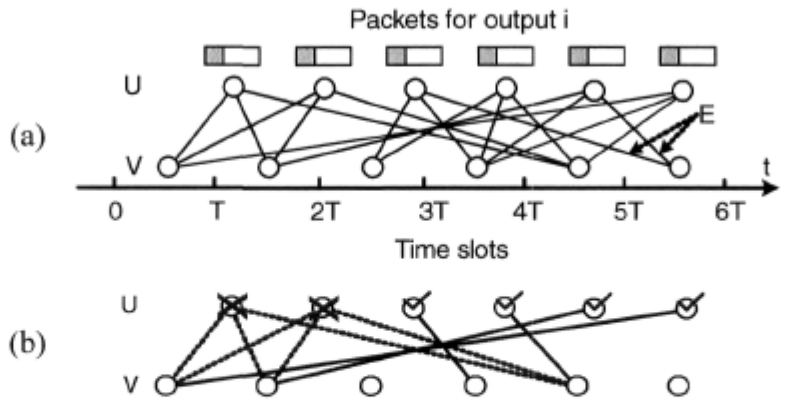

Figure 6.1. (a) An example of the bipartite graph, (b) an example of matching; U : packets destined to a given output port ($|U| \le \alpha M$), V : time slots ($|V| = 3M/4$), E : edges, T : slot duration.

Figure 6.1 (b), where packets 3, 4, 5, and 6 are matched and they will be switched in time slots 4, 5, 2, and 1 respectively. Packets 1 and 2 are not matched and are discarded.

b) Second Phase

The packets dropped in the first phase will be retransmitted in the second phase of transmission. The last quarter of each frame is reserved for the second phase. The core node will use a scheduling algorithm to assign conflict-free time slots to the leftover packets. The dropped packets will be retransmitted in a frame starting d_{prop} slots later. Many conflict-free scheduling algorithms have been proposed in the literature that can be used for the second phase of transmission [66,68]. In [68], $M/4$ time slots were shown to be sufficient to schedule $M/4$ packets. Therefore, it is guarantied that all the leftover packets are received successfully after the second phase. We will not explain the details of this phase here; our simulation results leads us to conclude in section 6.6 that this phase can be eliminated.

c) Previously Reported Bounds

When assuming (α, S)-smooth traffic, [67] shows that the average access delay of packets can be made small compared to d_{prop}, provided that the frame length is sufficiently large.

The average access delay is heavily weighted by successful packets in the first phase with small access delay, and minimally influenced by the small number of packets that must be retransmitted in the second phase and suffer access delay more than d_{prop}.

The bound for the probability of packet drop (P_e) of the first phase is given as follows [67]:

$$P_e \le C/M + (\alpha M - 1)\max\{(2C/M)^2, (C\alpha)^{\alpha M}\}, \quad (1)$$

where

$$C = 10\alpha e^2/9.$$

According to this equation, $P_e \to 0$ as $M \to \infty$. If we choose M large enough to make $P_e \le \varepsilon/3$ (ε a small number) and if $d_{prop} \ge 9M/\varepsilon$, then the access delay will be $d_a \le \varepsilon d_{prop}$.

As an example, according to this bound, M larger than 422063 (slots) is required to get the probability of packet drop $P_e \le 10^{-4}/3$. For slot duration equal to 10 μs, the frame length will be 4.22 s. Clearly this is an unrealistic scenario: very large frames are required to make the drop probability arbitrarily small.

We will demonstrate via simulation that, in contrast to the loose bound of (1), the drop probability decreases quickly for relatively small frames (on the order of 50 slots). Indeed, we will find that the drop rate is low enough to obviate the use of the second phase of retransmission and thus reduce the throughput penalty of the scheduling algorithm. While [67] assumed an optimal maximum cardinality matching algorithm at the core node, we examine a low complexity practical matching algorithm in our simulator, as described in the next section.

6.4. Bipartite Graph Matching Algorithm

As described in section 6.3, the core node attempts to maximize the number of packets that can be switched successfully in the first phase. This is performed by constructing a bipartite

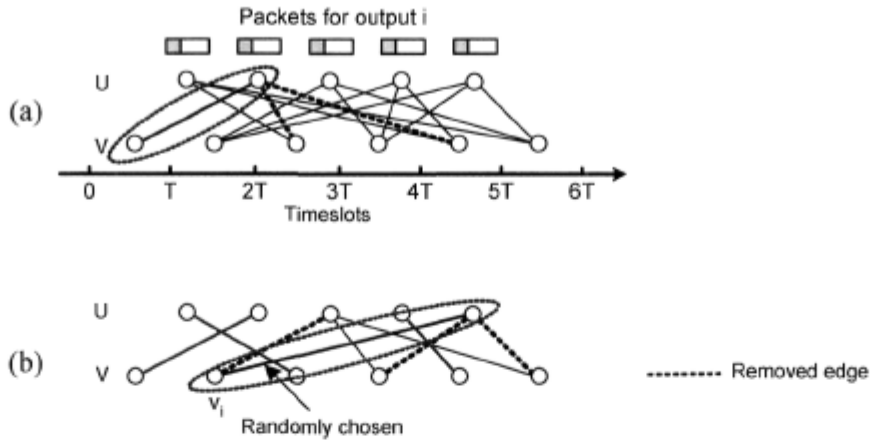

Figure 6.2. Bipartite graph matching algorithm, (a) first stage, (b) second stage.

graph for each output port and using a maximum cardinality matching algorithm to find a maximum match on the graph [67]. Maximum cardinality matching algorithms are too complex to be implemented in a practical system [69]; the most efficient algorithm currently known converges to a maximum-cardinality match in $O(n^5/2)$ iterations (for an n by n bipartite graph). We use a simple matching algorithm with a fixed, small number of iterations that pairs the vertexes of the bipartite graph but does not guarantee the maximum cardinality match. The algorithm adds the matched edges incrementally, as opposed to maximum cardinality matching algorithms, where the matched edges made earlier in the matching processes might be removed later during the process. The proposed algorithm consists of two stages; the first stage is operated for multiple iterations. In the first stage, the packets without contention are found and matched; the two remaining edges for each match are removed from the graph for these packets. For example in Figure 6.2 (a), the edge indicated by the solid line is selected and the two remaining edges indicated by dashed lines are removed from the graph. At the next iteration of the first stage, newly contention-free (due to pruning in the previous iteration) packets are identified and matched. After l iterations we enter the second stage of the algorithm, and randomly choose among the conflicting packets (Figure 6.2 (b)). The steps of the algorithm are as follows:

First stage:

Step 1: Identify all the contention-free packets and match them; $l = l - 1$

Step 2: Remove the two remaining edges originating from the matched packets.

Step 3: If unmatched packets remain go to step 1 for l iterations.

Second stage:

Step 4: Select randomly one of the edges in the first unmatched time slot, v_i.

Step 5: Remove all the edges originating from the packet selected in step 4.

Step 6: Remove all the edges ended to v_i.

Step 7: Go to step 4 until no unmatched time slots remain.

6.5. Simulation Results

6.5.1. (α,S)-Smooth Traffic

We simulated a star network with an $N \times N$ ($N = 4$, 8, and 16) switch at the core node and measured the packet drop rate of the first phase of the randomized scheduling algorithm. Initially, we consider (α, S)-smooth incoming traffic. We assume $\alpha = 1/4$ and the number of packets is equal to αM over all the frames. This assumption gives the worst case performance, as truly (α, S)-smooth traffic can have less than αM packets in a frame.

To simulate the (α, S)-smooth traffic, we use a traffic matrix for each frame whose $(i_1, i_2)^{\text{th}}$ entry is the number of packets originating from input i_1 and destined to output i_2. The traffic matrix should be a perfect matrix to represent (α, S)-smoothness. A perfect matrix is a square matrix with nonnegative integer entries, where the row- and column-sums are equal. In our worst case realization of (α, S)-smooth traffic with frame of length M, the row- and column-sums are equal to αM. An example of a perfect matrix with $\alpha M = 4$ is illustrated in Figure 6.3 (a). We use Monte-Carlo simulation and we change the traffic

$$\begin{pmatrix} 1 & 2 & 1 & 0 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 1 & 0 \\ 1 & 0 & 1 & 2 \end{pmatrix} \qquad \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

(a) (b)

Figure 6.3. (a) Perfect matrix, (b) uniform matrix; $\alpha M = 4$, $N = 4$.

matrix randomly for each trial but constrain the matrix to be perfect. We run the simulator for 10^6 frames, corresponding to $\alpha Mn \times 10^6$ total packets. The matching algorithm presented in section 6.4 with three iterations of the first stage is used to simulate the core node. We use three iterations, as more iterations did not improve the performance. Figure 6.4 gives the packet drop rate versus frame length, M slots. The curves marked with stars correspond to (α, S)-smooth traffic, where the traffic matrix in our simulator is randomly selected from the collection of all possible perfect matrices. We also simulate the network with uniform traffic matrices with entries equal to $\alpha M / n$. By uniform we mean a matrix where all the entries are the same. A uniform matrix with $\alpha M = 4$ and $N = 4$ is illustrated in Figure 6.3 (b). The uniform matrix gives the worst case performance for the system, since the packet conflict rate is maximized. The packet drop rate for the uniform traffic matrix is marked with circles in Figure 6.4.

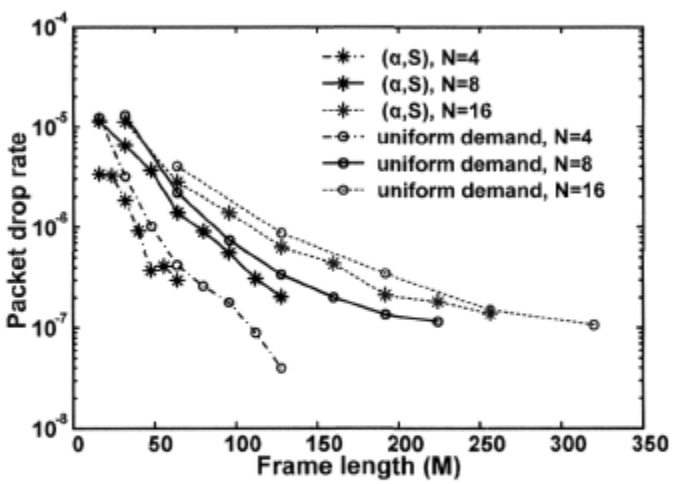

Figure 6.4. Packet drop rate versus frame length M slots with (α, S)-smooth traffic and uniform traffic matrix; N : number of ports.

In Figure 6.4 we see that, as predicted in [67], the packet drop rate decreases as the frame length increases. The simulations show that, in contrast to the bounds provided in [67], low packet drop rate (less than 10^{-5}) can be achieved even for frames as short as 50 slots. According to [67], the performance should be independent of the number of ports, however, in our results performance degrades for larger numbers of ports. This is because we use a matching algorithm that does not guarantee the maximum cardinality match assumed in [67].

6.5.2. Markov-Modulated Bernoulli Traffic

In the last section we assumed that the incoming traffic has the (α, S)-smoothness property, which is not realistic for a decentralized network. In a decentralized network we cannot ensure the total number of packets originating form all sources, and requesting the same destination is less than or equal to αM in all frames. In this section, we will simulate the packet drop rate for a more realistic arrival process, namely Markov-Modulated Bernoulli (MMB) which models the burstiness of traffic.

A MMB process is based on a two state Markov process, as illustrated by the transition diagram of Figure 6.5, where "1" denotes the arrival of a packet and "0" no packet arrival [70]. The transition probability from state j_1 to state j_2 ($j_1, j_2 \in \{0,1\}$) is indicated by $P(j_1, j_2)$. The MMB traffic generator is defined with two parameters: traffic rate α and burstiness ρ. The transition probabilities can be obtained form the following equations:

$$P(0,1) = \alpha(1-\rho), \text{ and}$$

$$P(1,1) = \rho + \alpha(1-\rho).$$

We assume that the incoming packets have mean arrival rate equal to the worst case (α, S)-smooth simulation, *i.e.*, $\alpha = 1/4$. We also assume that the packets are uniformly distributed among the outputs, that is, the probability of having a packet destined to a given output is α / N.

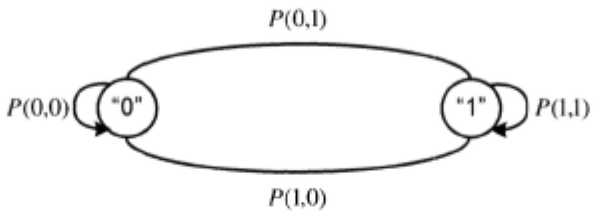

Figure 6.5. State transition diagram for the Markov-Modulated Bernoulli traffic model.

The incoming packets are generated by the MMB traffic simulator, they are buffered at the edge nodes, and no more than αM packets are transmitted at each frame. We do not put any restriction on the total number of packets destined for a given output port. Each packet is transmitted in three randomly chosen slots and the matching algorithm explained in section 6.4 is used to set up the connections of the core switch in each time slot. The burstiness parameters (ρ) of the MMB process is set to 0.7, 0.5, and 0. When ρ is equal to zero the MMB process is actually a simple Bernoulli process.

The packet drop rate versus the frame length is given in Figure 6.6. We plot the packet drop rafte of (α, S)-smooth traffic in the same figure for comparison. As expected, the performance is degraded for MMB compared to the (α, S)-smooth traffic. To understand why degradation occurs, consider the case where the packets from all the source nodes have the same output address and thus high probability of packet drop. Under the (α, S)-smooth assumption this is impossible, but under the Markov-Modulated Bernoulli assumption this can occur with non-zero probability. This probability increases with burstiness, resulting in a worsened packet drop rate. In Figure 6.6 we see that even with MMB arrivals, a packet drop rate lower than 10^{-4} can be achieved for short frames with only 300 slots in the worst case. This result shows the effectiveness of the randomized scheduling algorithm for more realistic traffic models.

6.6. Conclusions on the Utility of the Second Phase

As demonstrated in Figure 6.6, low packet drop rate (less than 10^{-4}) can be achieved even for short frames. Considering these results, we conclude that the second phase of transmission can be eliminated and the dropped packets can be retransmitted as is

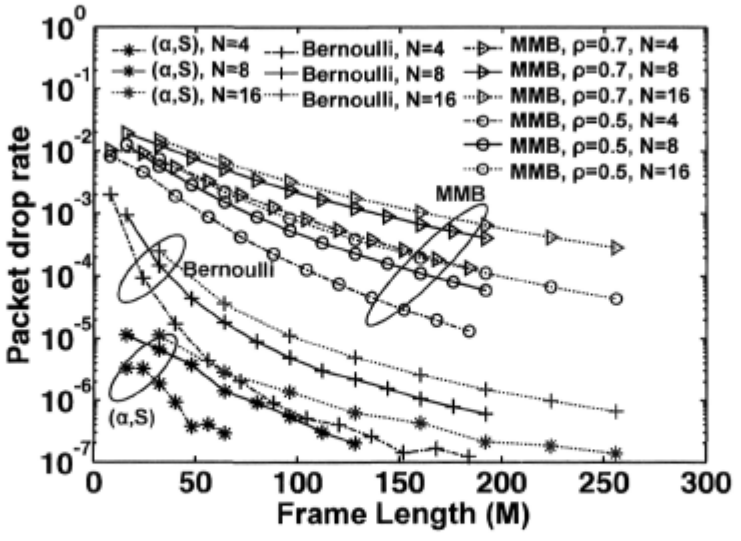

Figure 6.6. Packet drop rate versus frame length M slots; ρ : burstiness, N : number of ports, MMB: Markov-Modulated Bernoulli.

performed in the first phase of transmission. In this case, α may be increased from 1/4 to approach 1/3, resulting in higher throughput. Furthermore, eliminating the second phase leads to a less complex system as there is no need for a processing module for the conflict-free scheduling algorithm. It will also be beneficial in terms of packet framing delay at the edge nodes to have shorter frames (3/4 of the frame length required for the two-phase transmission) at the same performance.

6.7. Conclusion

In this chapter we presented a practical, economical, and simple optical network that can be implemented without optical buffer and optical synchronizers. This network is suitable for a single-hop star topology with large propagation delay, and trades off reduced throughput for delay which is much smaller than the propagation delay. We adapt an uncentralized randomized scheduling algorithm which is proposed taking the assumption of infinite length frames and (α, S)-smooth traffic. A matching algorithm is proposed to determine the connections of the core switch at each time slot. We show by simulation that the algorithm can be used for reasonable length frames and realistic traffic type like MMB. Our simulation results show an acceptable performance that makes this architecture desirable for a plausible packet-switched optical network.

Chapter 7

Conclusion

In this thesis, we investigated multi-hop optical packet switching networks in which routing is performed using optical labels. We reviewed the latest developments in enabling label switched optical networks and identified the main challenges. A wide survey showed that current solutions, mostly based on optical correlation techniques, inherently suffer from high splitting losses and thus cannot be scaled to a large number of users. We proposed optical networks based on multi-wavelength (M-λ) labels. We claim that with the proposed schemes, a scalable, high-speed, simple, and low-cost network can be implemented. Our proposal employs commercially available optical components such as incoherent optical sources, directly-controlled DFB lasers, AWGs, FBGs and low-speed electronic hardware.

We proposed two main structures: spectral amplitude codes (SAC) as labels with correlation-based label processing, and binary M-λ labels with multi-stage tree switch. Both approaches are capable of ultra-fast forwarding of variable-length packets.

In the first method, we use the auto- and cross-correlation properties of the SACs and code-multiplex the labels to implement a label stack. Hierarchical addressing is supported by the label stack reducing label splitting losses. The label processor uses correlators to identify the incoming labels, where the auto-correlation signal opens a switching gate to route the packet to the proper output port. The label processing is on-the-fly; the label detection speed corresponds to the traveling time of light through the correlator plus the photodetection delay. Optical packets can consist of a payload and separable SAC-labels or SAC-encoded payload. In the first case, a laser outside the band of the label is used to generate the high-rate payload; the labels are modulated at the packet rate. In the second case, the label is implicit in the payload bits; the label signal is the payload carrier. We examine the performance of the two schemes experimentally by implementing two nodes and routing optical packets with two stacked labels. The optical packets travel over 80 km of fiber end-to-end. We experimentally achieve error-free transmission at data rates of 10 Gbps and 2.5 Gbps for packets with separable SAC-labels and SAC-encoded payload respectively.

As the second label encoding approach, we use binary M-λ labels where the label bits correspond to separate wavelengths; the 0/1 state of a bit determine the omission/presence of the corresponding label wavelength. Routing of optical packets is performed over a multi-stage tree switch. An AWG separates the label channels; each channel controls one stage of a tree-like multi-stage switch. The advantages of this technique include high-speed packet forwarding, low-complexity hardware, fixed label power loss (equal to the AWG loss), rich set of labels, and scalability. The switch element could be controlled directly by the filtered label wavelength, or photodetected for an electronically controlled switch. We experimentally demonstrate the proof of the concept by error-free transmission of variable-length optical packets over 80 km of fiber and switching the 10 Gbps data with a 1×4 electronically controlled multistage optical switch.

We also suggest a label strategy to avoid label swapping to avoid label swapping. We time-stack all the required labels for the entire optical path and send them along the packet. This solution alleviates the sophisticated electrical table lookup to identify the next label, and also the requirement for a tunable swapper. We experimentally demonstrate this scheme by generating optical packets with two time-stacked labels and a 10 Gbps payload. The two labels are used for 2-hop routing over 1×4 electronically controlled multistage switch. We use low-complexity, off-the-shelf components for label processing.

Finally, in this thesis we investigate an optical network with a simplified topology (as apposed to the pervious networks with mesh topology) and address the packet contention issue. We have analyzed the performance of a practical, economical, and simple architecture that can be implemented without optical buffers and optical synchronizers. The network topology relieves the issue of synchronization. We use a randomized scheduling algorithm for contention resolution, alleviating the need for a centralized controller to coordinate the edge nodes. We propose a low complexity bipartite-graph matching algorithm to be used by the core. Performance is evaluated via simulation for a realistic traffic type, Markov-Modulated Bernoulli traffic. Our simulation results show acceptable performance (packet drop rate) that makes this architecture desirable for plausible, near-term packet-switched optical networks.

Future work

The maximum number of hops that the optical packets can tolerate should be studied. According to that number a TTL is assigned to the packets to limit its maximum number of hops before being discarded. A TTL decreasing mechanism consistent with the packet format should be designed. For example, in the case of time stacked labels, an additional wavelength can be assigned for this purpose. This wavelength is turned on only during the label slot that corresponds to the maximum number of hops. At each hop the TTL wavelength is checked and the packet is discarded when this wavelength is detected.

Contention resolution in a mesh topology is another unsolved issue. We discussed the contention resolution schemes proposed in the literature in section 2.3.3, including wavelength conversion, buffering and deflection routing. Due to our packet format based

on multi-wavelength labels, contention resolution with wavelength conversion might be complex and would require architectural network considerations. Employing limited buffering would well fit to our strategy. For example multi-wavelength switching with contention resolution mechanism based on feed-forward input buffers is proposed in [71].

Employing all-optical switches instead of electronically controlled switches needs further consideration. All-optical switches provide faster switching time and do not need photodetection [14,27,72]. However depending on the switch structure it might add cost and complexity to the system. An all-optical switch which is bit-rate and wavelength transparent requires a through study.

In chapter 6, we proposed using randomized scheduling algorithm for contention resolution in a star topology by transmitting packets in three randomly chosen time slots. Multiple wavelengths or orthogonal codes can be used instead of multiple time slots. This will reduce the length of the frame and increase the throughput. Further simulation study of the concept and possibly experimental demonstration would be another future research topic.

Appendix A

Binary Multi-Wavelength Label Switching with Label Swapper

A.1. Introduction

In chapter 5, we proposed binary multi-wavelength (M-λ) labels used with a multi-stage switch architecture for optical packet forwarding. In addition to forwarding, label swapping is performed in GMPLS label-switching routers, that is, the label is replaced by a new value to be used for forwarding at the next hop. All-optical swapping of M-λ labels using cross-gain modulation (XGM) and semiconductor fiber ring laser (SFRL) has been recently proposed in [21], by researchers from McGill University. The label swapper was brought to

our laboratory at Laval University where we incorporated it into our experimental packet switched network exploiting M-λ labels. In this joint experiment, we generated variable-length packets with four different labels, performed label swapping and switching through a 1×4 multistage switch, and showed the bit error rate performance [53]. In the following sections, we will first describe the principle of operation of label swapper and then explain the experimental setup and results.

A.2. All-Optical Label Swapper

All-optical swapper employs a SOA-based fiber ring laser [21] as illustrated in Figure A.1. Two SOAs are used in the SFRL; SOA_1 is slightly biased to provide a small gain, and SOA_2 is highly biased, capable of cross-gain modulation. The two fiber Bragg gratings in the loop determine the wavelengths of the output label (here $[\lambda_2, \lambda_3]$=[1555.3,1556.1 nm]). When the input signal to the SFRL is off, the gain of the SOAs is sufficient to establish a laser at λ_2 and λ_3 in the ring. On the other hand, when the signal is injected to SFRL, SOA_2 becomes saturated and does not provide enough gain to the ring to lase. Consequently, the output power turns off. The two wavelengths can be modulated in this way. However, if the labels are directly fed to the SFRL, label conversion

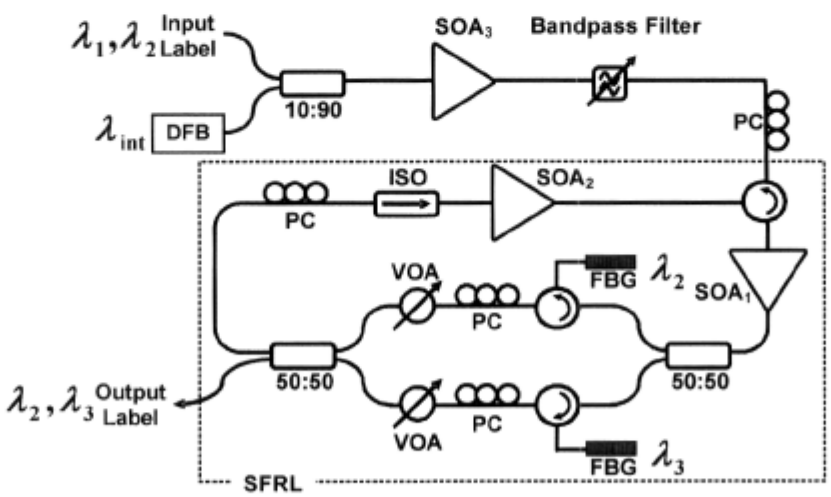

Figure A.1. All-optical swapping of M-λ labels using dual-stage XGM [53]; SOA: semiconductor optical amplifier, PC: polarization controller, ISO: isolator, FBG: fiber Bragg grating, VOA: variable optical attenuator.

will result in inverted output. Another SOA (SOA_3) has been employed before the SFRL to map the modulation pattern of the input label onto an intermediary wavelength λ_{int} by XGM. In addition to overcoming the problem of inversion, the first stage ensures that any performance degradation in XGM of SFRL due to overlapping wavelengths between the input and output labels can be avoided, since λ_{int} can simply be chosen out-of-band.

In the experimental setup, the two output label wavelengths are set by FBGs with an insertion loss of <1 dB, 3 dB bandwidth of ~5 GHz, and a reflectivity of 95%. Tunability of the output label wavelengths can be achieved by strain-tuning the FBGs. Variable optical attenuators (VOAs) are used to equalize the powers of the two lasing wavelengths and polarization controllers (PC) are used to optimize the lasing operation. The modulation speed of the SFRL (~100 KHz) corresponds to the cavity length. The cavity length is kept as short as possible so that high-speed operation is possible. The label swapper could be compact and integrated to further reduce the cavity length and increase the modulation speed.

A.3. Experimental Setup and Results

The simplified scheme of the experimental setup is illustrated in Figure A.2; it consists of a packet generator and two nodes. Variable-length packets with different labels are generated by the packet generator. Label swapping and forwarding is performed in the first and second node respectively. At the first node, label and payload being at orthogonal wavebands are separated by optical filtering. The labels are sent to a controller which decides if the labels should be swapped or remain unchanged. The controller drives a 1×2 switch the upper arm of which directs the labels to the label swapper and the lower arm bypasses the labels. The labels and payload are coupled again and sent to the second node, where forwarding takes place. Successful packet switching and BER measurement confirms the performance of the system.

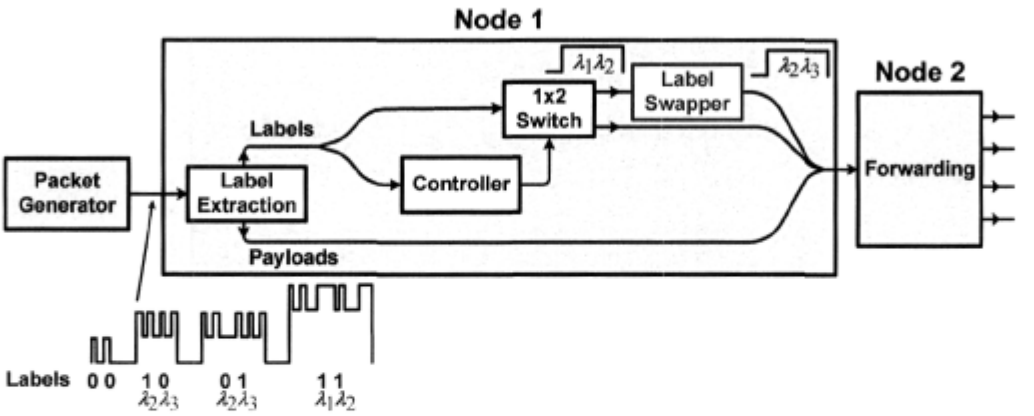

Figure A.2. Simplified scheme of the binary M-λ label switching experiment with label swapping.

Figure A.3 shows the experimental setup with more details. At the packet generator four packets are generated, the lengths of which are 2.496, 4.992, 7.488, and 9.984 μsec (packets 1, 2, 3, and 4 respectively). The packets contain 2^{15}-1 PRBS data at 10 Gbps generated by externally modulating a laser signal at 1553.9 nm. Three directly modulated DFB lasers driven by programmable waveform generators produce the labels. The laser wavelengths are $\lambda_1 = 1554.77$, $\lambda_2 = 1555.32$, and $\lambda_3 = 15556.12$ nm. Label and payload synchronization is achieved by connecting the 10 MHz output of the pattern generator to the

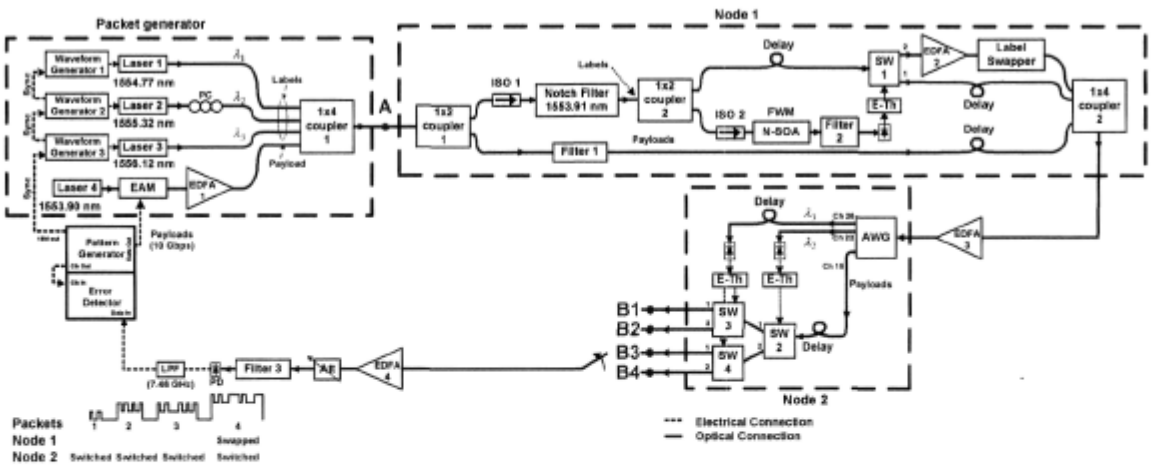

Figure A.3. Experimental setup of packet switching and label swapping with binary M-λ labels. EAM: electro-absorption modulator, PC: polarization controller, EDFA: Erbium doped fiber amplifier, ISO: isolator, N-SOA: nonlinear semiconductor optical amplifier, E-Th: electrical thresholder, SW: 1×2 switch, AWG: arrayed waveguide grating, , PD: photodiode, Att: attenuator, and LPF: low-pass filter.

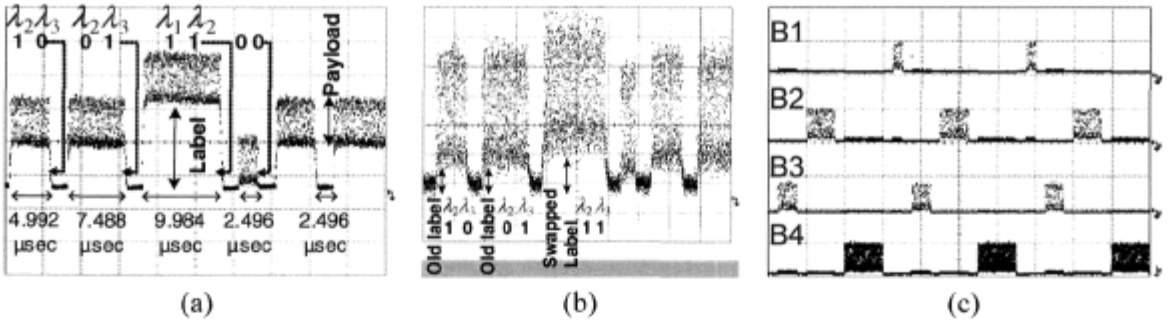

Figure A.4. Oscilloscope traces of (a) generated packets (point A of the setup), (b) packets after the label swapper, and (c) switched packets (point B of the setup).

synchronization input of the waveform generators. The label and payload signals are coupled to form the packets. We monitor the coupler output and adjust the phase of the waveform generators to align payloads and labels in time. The oscilloscope trace of the generated packets is illustrated in Figure A.4 (a).

At the first node, packets are split by 1×2 coupler 1 where the payloads are stripped off at the lower arm using a band-pass filter of 0.2 nm (Filter 1). The labels are extracted at the upper arm using a notch filter with the center wavelength matched to the payload wavelength. The notch filter is a fiber Bragg grating used in transmission with 24.8 GHz 3 dB bandwidth, insertion loss of 2 dB, and 98% reflectivity (see Figure A.5). The control mechanism which decides about swapping or bypassing the labels is based on four wave mixing (FWM) in a nonlinear SOA (N-SOA). In this setup, only the label of packet 4 [λ_1 ,λ_2] is swapped. This label generates FWM products at the N-SOA. An optical band-

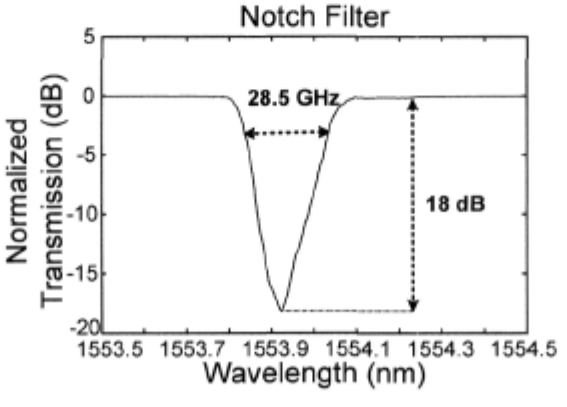

Figure A.5. Normalized transmission of the notch filter utilized in the setup of Figure A.3.

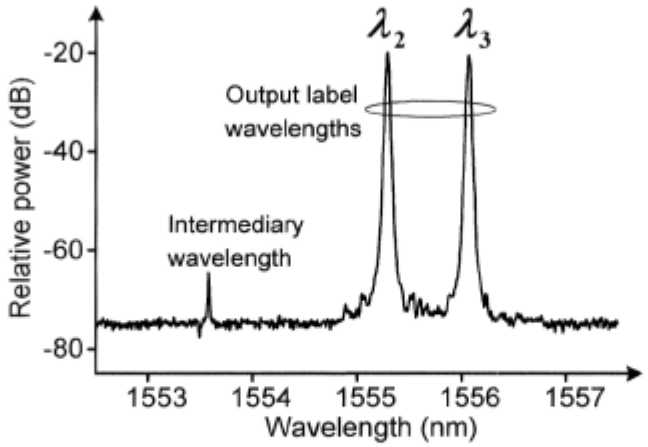

Figure A.6. Spectral output of the label swapper [53].

pass filter of 0.2 nm selects one of the FWM products which is then detected and fed to the 1×2 optical switch (SW1). Therefore, only label [λ_1 , λ_2] traverses the swapper; the rest of the labels are bypassed through port 1 of the switch and remain unchanged.

The result of all-optical label swapping is illustrated in Figure A.6, which shows the output label wavelengths with high signal-to-noise ratio and the absence of the original input label wavelengths. Note that λ_2 is common in both input and output labels and it is re-generated by the SFRL at the output. The oscilloscope traces of the signal after the first and second stage of the swapper are illustrated Figure A.7 (a) and (b). In Figure A.7 (c) an electrical low-pass filter with 10 MHz pass-band has been used after detection to remove the

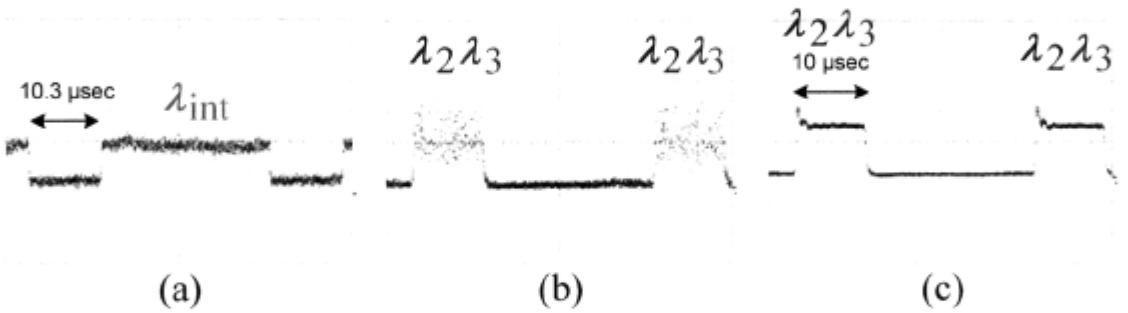

Figure A.7. Oscilloscope traces of (a) the output of the swapper first stage (intermediary wavelength), (b) swapper output, and (c) swapper output when a 10 MHz electrical low-pass filter is used to remove the intensity noise of the swapper output.

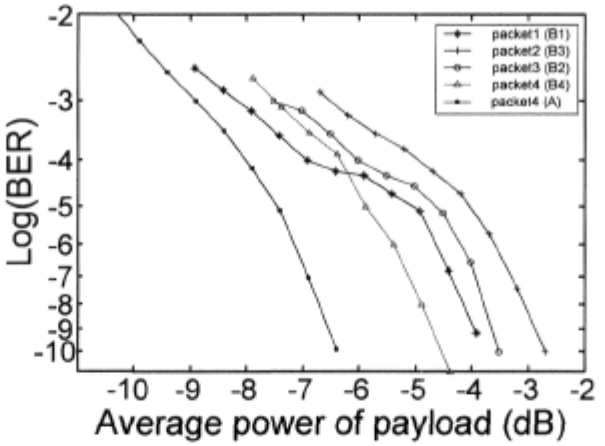

Figure A.8. Bit error rate versus average received power of payload for points A and B of Figure A.3.

intensity noise of the converted label. The labels and payload are coupled again; the time alignment is adjusted by fiber delay lines. Figure A.4 (b) shows the new label having been re-inserted with its corresponding payload.

At the second node, the label wavelengths are separated by two arms of an AWG, each to control one stage of a tree switch. The AWG has 32 channels with 25 GHz spacing and 5 dB insertion loss. The labels are detected by slow-speed photodiodes and fed to electrical thresholders providing the required voltage to set the state of the switches. The combination of three $LiNbO_3$ 1×2 optical switches with switching time less than 100 ns constructs a 1×4 switch. Each packet is forwarded to one of the four outputs according to its label. The alignment between payloads and control signals at the switching elements is achieved by fixed fiber delay lines. We measure the BER versus the average received power of the payload after being switched and also after the pattern generator (point A in the setup). We use an optical band-pass filter (pass-band=0.2 nm) before detection to filter out the ASE of amplifiers (and labels at point A). An electrical low-pass filter with a pass-band equal to three quarters of the bit rate (7.46 GHz) is used after the photodiode to remove the out-of-band noise. The technical tips mentioned in section 5.4.1 for BER measurements are also exploited in this experiment. The BER curves are illustrated in Figure A.8, where 80% of

the bits in the middle of the payload are considered in BER measurement*. We believe the variation between BER curves for the four outputs is due to different ASE for variable-length packets added by EDFA 4. Successful packet switching with the swapped label and also error free performance validates the feasibility of combining the swapper and the forwarding structure in a label switching network.

---

* This restriction is due to experimental expediency and could be reduced with better electronics generating the control signal.

References

1 A. Banerjee, J. Drake, J. P. Lang, B. Turner, K. Kompella, and Y. Rekhter, "Generalized multiprotocol label switching: an overview of routing and management enhancements," *IEEE Communications Magazine*, vol. 39, pp. 144-150, Jan. 2001.

2 J. E. McGeehan, M. C. Hauer, A. B. Sahin, and A. E. Willner, "Multiwavelength-channel header recognition for reconfigurable WDM networks using optical correlators based on sampled fiber Bragg gratings," *IEEE Photonics Technology Letters*, vol. 15, pp. 1464–1466, Oct. 2003.

3 M. C. Cardakli, S. Lee, A. E. Willner, V. Grubsky, D. Starodubov, and J. Feinberg, "Reconfigurable optical packet header recognition and routing using time-to-wavelength mapping and tunable fiber Bragg gratings for correlation decoding," *IEEE Photonics Technology Letters*, vol. 12, pp. 552-554, May 2000.

4 M. Y. Jeon, Z. Pan, J. Cao, Y. Bansal, J. Taylor, Z. Wang, V. Akella, K. Okamoto, S. Kamei, J. Pan, and S. J. B. Yoo, "Demonstration of all-optical packet switching routers with optical label swapping and 2R regeneration for scalable optical label switching network applications," *IEEE Journal of Lightwave Technology*, vol. 21, pp. 2723-2733, Nov. 2003.

5 K. Vlachos, J. Zhang, J. Cheyns, Chi, N. Sulur, E. Van Breusegem, I. T. Monroy, J. G. L. Jennen, P. V. Holm-Nielsen, C. Peucheret, R. O'Dowd, P. Demeester, and A. M. J. Koonen, "An optical IM/FSK coding technique for the implementation of a label-controlled arrayed waveguide packet router," *IEEE Journal of Lightwave Technology*, vol. 21, pp. 2617-2628, Nov. 2003.

6 K. Kitayama, N. Wada, and H. Sotobayashi, "Architectural considerations for photonic IP router based upon optical code correlation," *IEEE Journal of Lightwave Technology*, vol. 18, pp. 1834-1844, Dec. 2000.

7 G. Cincotti, N. Wada, and K. Kitayama, "Characterization of a full encoder/decoder in the AWG configuration for code-based photonic routers-part I: modeling and design," *IEEE Journal of Lightwave Technology*, vol. 24, pp. 103–112, Jan. 2006.

8 N. Wada, H. Harai, and W. Chujo, "Multi-hop, 40 Gbit/s variable length photonic packet routing based on multi-wavelength label switching, waveband routing and label swapping," *Proceedings of Optical Fiber Communications Conference (OFC'02)* Anaheim, CA, pp. 216-217, Mar. 2002.

9 B. Hoanca, S. Dubovitsky, Xiaoxing Zhu, A. A. Sawchuk, W. H. Steier, and P. D. Dapkus, "All-optical routing using wavelength recognizing switches," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 2243–2254, Dec. 1998.

10 L. A. Buckman, M. S. Wu, G. Giaretta, G. S. Li, P. K. Pepeljugoski, J. W. Goodman, A. Varma, K. Y. Lau, and C. J. Chang-Hasnain, "A novel all-optical self-routed wavelength-addressable network (SWANET)," *IEEE Photonics Technology Letters*, vol. 7, pp. 1066–1068, Sept. 1995.

11 D. J. Blumenthal, P. R. Prucnal, and J. R. Sauer, "Photonic packet switches: Architectures and experimental implementations," *Proceedings of the IEEE*, vol. 82, pp. 1650-1667, Nov. 1994.

12 P. Seddighian, J. B. Rosas–Fernández, S. Ayotte, L. A. Rusch, S. LaRochelle, and A. Leon-Garcia, "Low-cost, scalable optical packet switching networks with multi-wavelength labels," *Proceedings of Optical Fiber Communications Conference (OFC'07)*, Anaheim, CA, Mar. 2007.

13 P. Seddighian, S. Ayotte, J. B. Rosas–Fernández, J. Penon, L. A. Rusch, and S. LaRochelle, "Label stacking in photonic packet switched networks with spectral amplitude code labels," *IEEE Journal of Lightwave Technology*, vol. 25, pp. 463-471, Feb. 2007.

14 J. B. Rosas-Fernández, W. Mathlouthi, M. Presi, S. LaRochelle, L. A. Rusch, and I. White, "All optical recognition of 36 SAC-labels with 12.5 GHz minimum bin

separation using a single correlator for optical label switching," *Proceedings of 24th European Conference on Optical Communications (ECOC '07)*, Sep. 2007.

15 A. Shacham, B. A. Small, O. Liboiron-Ladouceur, and K. Bergman, "A fully implemented 12×12 data vortex optical packet switching interconnection network," *IEEE Journal of Lightwave Technology*, vol. 23, pp. 3066-3075, Oct. 2005.

16 M. Kavehrad and D. Zaccarin, "Optical code-division-multiplexed systems based on spectral encoding of noncoherent sources," *IEEE Journal of Lightwave Technology*, vol. 13, pp. 534-545, Mar. 1995.

17 S. Ayotte, M. Rochette, J. Magne, L. A. Rusch, and S. LaRochelle, "Experimental verification and capacity prediction of FE-OCDMA using superimposed FBG", *IEEE Journal of Lightwave Technology*, vol. 23, pp. 724-731, Feb. 2005.

18 S. Ayotte, M. Rochette, J. Magne, L. A. Rusch, anf S. LaRochelle, "Experimental demonstration and simulation results of frequency encoded optical CDMA," *Proceedings of IEEE International Conference on Communications (ICC '04)*, vol. 3, pp. 1704–1708, 20-24 Jun. 2004.

19 J. Penon, S. Ayotte, L. A. Rusch., and S. LaRochelle, "Incoherent SAC OCDMA system at 7×622Mbps," *Proceedings of Conference on Lasers and Electro-Optics (CLEO '06)*, Long Beach, CA, May 2006.

20 Alberto Leon-Garcia, *Communication Networks, Fundamental Concepts and Key Architectures*, McGraw-Hill, 2004.

21 R. E. Gordon and L. R. Chen, "Demonstration of all-photonic spectral label switching for optical MPLS networks," *IEEE Photonics Technology Letters*, vol. 18, 586-588, Feb. 2006.

22 J. M. H.Elmirghani and H. T. Mouftah, "All-optical wavelength conversion: Technologies and applications in DWDM networks," *IEEE Communications Magazine*, vol. 38, pp. 86-92, Mar. 2000.

23 M. S. Borella, J. P. Jue, D. Banerjee, B. Ramamurthy, and B. Mukherjee, "Optical components for WDM lightwave networks," *Proceedings of the IEEE*, vol. 85, no. 8, pp. 1274-1307, Aug. 1997.

24 P. Seddighian, A. Leon-Garcia, and L. A. Rusch, "Analysis of randomized scheduling algorithm for a practical bufferless optical network," *IEEE Communications Letters*, vol. 11, pp. 537–539, Jun. 2007.

25 U. Black, MPLS and Label Switching Networks, 2nd Edition, Prentice Hall, 2002.

26 Andrew S. Tanenbaum, *Computer Networks*, Prentice Hall, 1996.

27 J. Kurumida, Y. Tatara, H. Uenohara, K. Kobayashi, "All-optical header recognition sub-system based on SOA-MZI switches," *Proceedings of Pacific Rim Conference on Lasers and Electro-Optics (CLEO'05)*, pp. 1790–1791, Aug. 2005.

28 X. Ma and G. Kuo, "Optical switching technology comparison: optical MEMS vs. other technologies," *IEEE Communications Magazine*, vol. 41, pp. S16-S23, Nov. 2003.

29 V. Chan, K. L. Hall, E. Modiano, and K. A. Rauschenbach, "Architectures and technologies for high-speed optical data networks," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 2146-68, Dec. 1998.

30 F. Xue, Z. Pan, Y. Bansal, J. Cao, M. Jeon, K. Okamoto, S. Kamei, V. Akella, and S. J. B. Yoo, "End-to-end contention resolution schemes for an optical packet switching network with enhanced edge routers," *IEEE Journal of Lightwave Technology*, vol. 21, pp. 2595-2604, Nov. 2003.

31 F. Callegati, G. Corazza, and C. Raffaelli, "Exploitation of DWDM for optical packet switching with quality of service guarantees," *IEEE Journal on Selected Areas in Communications*, vol. 20, pp. 190-201, Jan. 2002.

32 P. B. Hansen, S. L. Danielsen, and K. E. Stubkjaer, "Optical packet switching without packet alignment," *Proceedings of 24th European Conference on Optical Communications (ECOC '98),* vol. 1, pp. 591-592, Sep. 1998.

33 D. K. Hunter, M. C. Chia, and I. Andonovic, "Buffering in optical packet switches," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 2081-2094, Dec. 1998.

34 S. L. Danielsen, P. B. Hansen, and K. E. Stubkjear, "Wavelength conversion in optical packet switching," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 2095-2108, Dec. 1998.

35 F. Forghierri, A. Bononi, and P. R. Prucnal, "Analysis and comparison of hot-potato and single-buffer deflection routing in very high bit rate optical mesh networks," *IEEE Transactions on Communications*, vol. 43, pp. 88-98, Jan. 1995.

36 S. Yao, B. Mukherjee, S. J. B. Yoo, and S. Dixit, "A unified study of contention-resolution schemes in optical packet-switched network," *IEEE Journal of Lightwave Technology*, vol. 21, pp. 672-683, Mar. 2003.

37 W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 1-15, Feb. 1994.

38 C. S. Reed, "Multiple level minimum logic network," U.S. Patent, 5 996 020, Nov. 30, 1999.

39 D. J. Blumenthal, R. J. Feuerstein, and J. R. Sauer, "First demonstration of multihop all-optical packet switching," *IEEE Photonics Technology Letters*, vol. 6, pp. 457-460, March. 1994.

40 N. Wada, H. Harai, W. Chujo, and F. Kubota, "80G to 10G bit/s variable rate photonic packet routing based on multi-wavelength label switch," *Proceedings of 27th European Conference on Optical Communications (ECOC '01)*, vol. 3, pp. 308–309, 2001.

41 C. Skoufis, S. Sygletos, N. Leligou, C. Matrakidis, I. Pountourakis, and A. Stavdas, "Data-centric networking using multiwavelength headers/labels in packet-over-WDM networks: A comparative study," *IEEE Journal of Lightwave Technology*, vol. 21, pp. 2110-2122, Oct. 2003.

42 Shilin Xiao, Qingji Zeng, Jianxin Wang, Jie Xu, and Yun Wang, "Realization of multiwavelength label optical packet switching," *IEEE Photonics Technology Letters*, vol. 15, pp. 605–607, Apr. 2003.

43 A. Stavdas, C. Skoufis, I. Angelopoulos, G. Stassinopoulos, and I. Pountourakis, "On multi-λ packet labeling for metropolitan and wide-area optical networks," *Photonic Network Communications*, vol. 3, no. 1/2, pp. 131–145, Jan./June 2001.

44 N. Wada, G. Cincotti, S. Yoshima, N. Kataoka, and K. Kitayama, "Characterization of a full encoder/decoder in the AWG configuration for code-based photonic Routers-part II: experiments and applications," *IEEE Journal of Lightwave Technology*, vol. 24, pp. 113-121, Jan. 2006.

45 G. Cincotti, M. S. Moreolo, G. Manzacca, X. Wang, N. Wada, and K.-I Kitayama, "Multi-dimensional optical code processing in MPLS photonic routers," *Proceedings of Optical Fiber Communications Conference (OFC '06),* Anaheim, CA, Mar. 2006.

46 J. B. Rosas-Fernández, S. Ayotte, L. A. Rusch, and S. LaRochelle, "A simple ultrafast forwarding architecture using a single optical processor for multiple SAC-label recognition based on FWM," Submitted to *Journal of Selected Topics in Quantum Electronics*, Nov. 2007.

47 Q. Yang, K. Bergman, F. G. Johnson, and G. D. Hughes, "WDM packet routing for high capacity data networks," *IEEE Journal of Lightwave Technology*, vol. 19, pp. 1420-1426, Oct. 2001.

48 Q. Yang and K. Bergman, "Performance of the data vortex switch architecture under nonuniform and bursty traffic," *IEEE Journal of Lightwave Technology*, vol. 20, pp. 1242-1247, Aug. 2002.

49 Xiaohong Jiang, Hong Shen, Md. M.-ur-R. Khandker, and S. Horiguchi, "Blocking behaviors of crosstalk-free optical Banyan networks on vertical stacking," *IEEE/ACM Transactions on Networking*, vol. 11, pp. 982-993, Dec. 2003.

50 C.-T. Lea, "Multi-$\log_2$N networks and their applications in high-speed electronic and photonic switching systems," *IEEE Transactions on Communications*, vol. 38, no. 10, pp. 1740-1749, Oct. 1990.

51 C. Hawkins, B. A. Small, D. S. Wills, and K. Bergman, "The data vortex, an all optical path multicomputer interconnection network," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, pp. 409-420, Mar. 2007.

52 D. Gurkan, S. Kumar, A. Sahin, A. Willner, K. Parameswaran, M. Fejer, D. Starodubov, J. Bannister, P. Kamath, and J. Touch, "All-optical wavelength and time 2-D code converter for dynamically-reconfigurable O-CDMA networks using a PPLN waveguide," *Proceedings of Optical Fiber Communications Conference (OFC '03)*, vol. 2, pp. 654–656, March 2003.

53 P. Seddighian, V. Baby, C. Habib, L. R. Chen, L. A. Rusch, and S. LaRochelle, "All-optical swapping of spectral amplitude code labels for Packet switching," *Proceedings of IEEE/LEOS Photonics in Switching Conference*, San Francisco, CA, Aug. 2007.

54 P. Seddighian, S. Ayotte, J. B. Rosas–Fernández, J. Penon, S. LaRochelle, and L. A. Rusch, "Label stacking using spectral amplitude code labels for optical packet switching," *Proceedings of Optical Fiber Communications Conference (OFC '06)*, Anaheim, CA, Mar. 2006.

55 C-S. Weng and J. Wu "Perfect difference codes for synchronous fiber-optic CDMA communication systems", *IEEE Journal of Lightwave Technology*, vol 19, pp. 186-194, Feb. 2001.

56 Z. Wei and H. Ghafouri-Shiraz, "Codes for spectral-amplitude-coding optical CDMA systems," *Journal of Lightwave Technology*, vol. 20, no. 8, pp. 1284– 1291, Aug. 2002.

57 J. W. Goodman, *Statistical optics*. Wiley, New-York, 2000.

58 G. Brochu, S. LaRochelle, and R. Slavik, "Modeling and experimental demonstration of ultracompact multiwavelength distributed Fabry-Pérot fiber lasers," *IEEE Journal of Lightwave Technology*, vol. 23, pp. 44-53, Jan. 2005.

59 S. Ayotte and L. A. Rusch, "Experimental comparison of coherent versus incoherent sources in a four-user λ-t OCDMA system at 1.25 Gb/s," *IEEE Photonics Technology Letters,* vol. 17, pp. 2493-2495, Nov. 2005.

60 J. H. Patel, "Performance of processor-memory interconnections for multiprocessors," *IEEE Transactions on Computers*, vol. C-30, pp. 771-780, Oct. 1981.

61 A. Iocco, H. G. Limberger, R. P.Salathé, L. A.Everall, K. E.Chisolm, J. A. R.Williams, and I. Bennion, "Bragg grating fast tunable filter for wavelength division multiplexing," *IEEE Journal of Lightwave Technology*, vol. 17, pp. 1217-1221, Jul. 1999.

62 R. Tucker, "Optical Packet-Switched WDM Networks—A Cost and Energy Perspective," *Optical Fiber Communications Conference (OFC'08)*, San Diego, Feb. 2008.

63 P. Seddighian, Y. BenM'Sallem, A. Leon-Garcia, and L. A. Rusch, "Time-stacked optical labels: An alternative to label-swapping," *Optical Fiber Communications Conference (OFC'08)*, San Diego, Feb. 2008.

64 Y. Hibino, "Recent advances in high density and large scale AWG multi-demultiplexers with higher index contrast silica based PLCs," *IEEE Journal of Selected Topics in Quantum Electronics,* vol. 8, pp. 1090-1101, Nov./Dec. 2002.

65 M. Beshai, J. Fitchett, and A. Graves, "Structures of high-capacity reliable telecommunication networks," *Proceedings of Fourth International Workshop on Design of Reliable Communication Networks*, pp. 272–281, Oct. 2003.

66 T. Weller and B. Hajek, "Scheduling nonuniform traffic in a packet switching system with small propagation delay," *IEEE/ACM Transaction on Networking*, vol. 5, pp. 813–823, Dec. 1997.

67 B. Hajek and T. Weller, "Scheduling nonuniform traffic in a packet switching system with large propagation delay," *IEEE Transaction on Information Theory*, vol. 41, pp. 358–365, Mar. 1995.

68 E. A. Varvarigos, "The "Packing" and the "Scheduling" packet switch architectures for almost all-optical lossless networks," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 1757-67, Oct. 1998.

69 N. McKeown, "The iSLIP scheduling algorithm for input-queued switches," *IEEE/ACM Transaction on Networking*, vol. 7, pp. 188-201, Apr. 1999.

70 I. Stavrakakis, "Efficient modeling of merging and splitting processes in large networking structures," *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 1336–1347, Oct. 1991.

71 K. Watabe, M. Takagi, K. Machida, T. Tanemura, H. Imaizumi, Y. Nakano, and H. Morikawa, "320Gb/s multi-wavelength optical packet switching with contention resolution mechanism using PLZT switches," *Optical Fiber Communications Conference (OFC'08)*, San Diego, Feb. 2008.

72 N. Calabretta, Y. Liu, F. M. Huijskens, M. T. Hill, H. de Waardt, G. D. Khoe, and H. J. S. Dorren, "Optical signal processing based on self-induced polarization rotation in a semiconductor optical amplifier," *IEEE Journal of Lightwave Technology*, vol. 22, pp. 372-381, Feb. 2004.