Burst Rescheduling with Wavelength and Last-hop FDL Reassignment in WDM Optical Burst Switching Networks

S. K. Tan, G. Mohan, and K. C. Chua
Department of Electrical and Computer Engineering
National University of Singapore, Singapore
Email : elegm@nus.edu.sg

Abstract— **In this paper, we consider the problem of fast and efficient dynamic scheduling of bursts that belong to different classes of priority in *wavelength-division multiplexing* (WDM)-based *optical burst switching* (OBS) networks with limited optical buffers. In OBS networks, control and data components of a burst are sent separately with a time gap to ensure that resources such as wavelengths and *fiber delay lines* (FDLs) are reserved at various nodes before the data burst arrives. A *scheduling algorithm* with attractive features such as computational simplicity and efficient resource utilization is mandatory to quickly handle dynamic burst traffic and reduce burst dropping probability. While *void filling* algorithms achieve good burst dropping performance they are computationally complex. We propose *burst rescheduling* as an alternative to void filling which can do fast scheduling without requiring to examine and fill voids and at the same time can achieve good performance. Burst rescheduling uses two mechanisms known as *wavelength reassignment* and *last-hop FDL reassignment*. We develop a scheduling algorithm using the above rescheduling mechanisms called *Burst Rescheduling with Wavelength and Last-hop FDL Reassignment* (BR-WFR) which is computationally simpler than a void filling algorithm. We then discuss the signaling overhead and feasibility of implementing burst rescheduling. Through simulation experiments we demonstrate the effectiveness of the proposed burst rescheduling algorithm.**

I. INTRODUCTION

Wavelength division multiplexed (WDM) optical networks are a promising candidate for the next generation backbone transport networks. *Optical burst switching* (OBS), as described in [1], [2], [3], [4] is able to strike a balance between optical circuit and packet switching [2], [5], [6], [7].

A burst is formed at the ingress node by assembling a number of IP packets that are destined to the same egress node. A burst consists of a burst header (control packet) and a burst payload (data burst). The control packet and the data burst are initially separated by an offset time depending on the number of hops the burst has to traverse. The base offset time required is given by δH, where δ is the control packet processing time and H is the number of hops to be traversed. By extending *multi-protocol label switching* (MPLS) capabilities to OBS networks, explicit routing can be used at the ingress nodes [8]. The control packet traverses ahead of the data burst and reserve resources such as wavelengths and FDLs for the data burst before it actually arrives at a node. By using extra offset time, services belonging to different classes can be differentiated [2], [4]. A large number of long voids are likely to be created on wavelength channels due to the dynamic random arrival of bursts with different offset time and hop count, and the use of FDL buffers with varying length.

In the event that no wavelength or FDL is available for the entire duration of a burst at a node, the burst is dropped. In order to minimize this dropping, an efficient scheduling algorithm is required to choose the best wavelength on the outgoing link and an FDL (if needed) at the node. In the literature, a few burst scheduling algorithms have been proposed. They include *latest available unscheduled channel* (LAUC) and *latest available unused channel with void filling* (LAUC-VF) [1], [3].

LAUC [3] which is also known as *scheduling horizon* [1] uses only the last burst information on each wavelength. By doing so, it becomes computationally simple. The worst case time complexity is $O(W)$ where W is the number of wavelengths per link. Among all the eligible wavelengths it chooses the one whose latest available time is closest to the arrival time of a new burst. This will possibly improve the chances of scheduling the bursts that arrive later. In spite of this, it does not achieve good performance because it does not fill the voids between bursts. Also, FDLs may not be very useful because they may create longer voids. On the other hand, LAUC-VF performs significantly better than LAUC by allowing a new data burst to fill the voids between existing bursts. However, it is computationally complex. Its worst case time complexity is $O(NW)$ where N is the number of voids or bursts currently scheduled and W is the number of wavelengths per link. This complexity could be reduced to $O(NlogW)$, but it requires complex data structure such as binary tree which would result in increased processing time.

Our work is motivated by the above observation that existing scheduling algorithms are either computationally simple or achieve good burst dropping performance, but not both simultaneously. We propose *burst rescheduling* as an alternative to void filling to effectively utilize the resources by packing the bursts tightly without needing a computationally expensive operation to examine the voids created on wavelength channels. Whenever a new burst is successfully scheduled, burst

0-7803-7802-4/03/$17.00 (C) 2003 IEEE

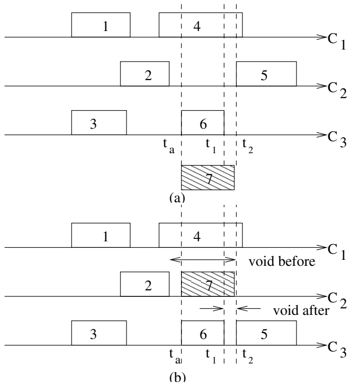

Fig. 1. Illustration of the benefit of wavelength reassignment. (a) LAUC fails to schedule burst 7. (b) Burst 7 can be scheduled by using wavelength reassignment.

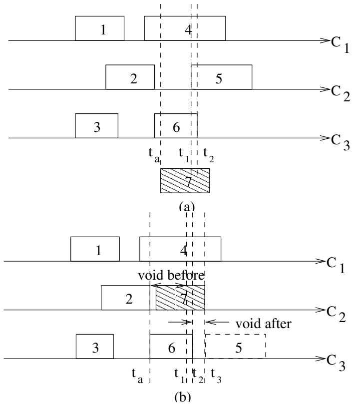

Fig. 2. Illustration of the benefit of burst rescheduling with FDL reassignment. (a) LAUC fails to schedule the new burst, wavelength reassignment does not help. (b) The new burst is scheduled by allowing FDL reassignment.

rescheduling uses one or both of the mechanisms – *wavelength reassignment* and *last-hop FDL reassignment* – on a possible existing burst to help increase the chances of accepting bursts that arrive later. We develop a scheduling algorithm using the above rescheduling mechanisms, namely, *Burst Rescheduling with Wavelength and Last-hop FDL Reassignment* (BR-WFR) which is computationally simpler than a void filling algorithm. We explain the proposed algorithm, signaling issue and the feasibility of implementing burst rescheduling in Section II. The performance of the proposed rescheduling algorithm is studied through simulations in Section III. Finally, concluding remarks are made in section IV.

II. PROPOSED BR-WFR RESCHEDULING ALGORITHM

We consider a network where every node is equipped with a limited FDL buffer of size F with FDL_i $(i = 1, 2, \ldots, F)$ capable of optically delaying data by i time units. Each link is assumed to carry W wavelengths.

A. Wavelength Reassignment

Wavelength reassignment considers reassigning the last burst from a wavelength to the wavelength assigned to the newly scheduled burst. Figure 1 illustrates the benefit of wavelength reassignment. Figure 1(a) shows that without wavelength reassignment, burst 7 cannot be scheduled. As burst 1 through burst 6 have been scheduled one by one to wavelength channels C_1, C_2 or C_3 based on the latest available wavelength similar to LAUC, burst 7 arriving at time t_a cannot be scheduled to any wavelength. Figure 1(b) shows that with wavelength reassignment, burst 7 can be scheduled successfully. Here, upon successful scheduling of burst 6, burst 5 at t_2 is reassigned from C_2 to C_3 as a shorter void $(t_2 - t_1)$ is formed by burst 5 with burst 6 at C_3 than with burst 2 at C_2. When burst 7 arrives, it can be scheduled to C_2.

B. Last-hop FDL Reassignment

If wavelength reassignment is not possible last-hop FDL reassignment can be used. The idea is to reassign an FDL to a burst which traverses its last hop. We do not consider the FDL reassignment at hops other than the last-hop because (i) FDL reassignment changes the burst arrival time at the downstream nodes and there is no guarantee that the required resources will be available, (ii) processing and signaling overhead could become high as every downstream node need to be notified. However, last-hop reassignment does not pose these problems. The benefit of FDL reassignment is illustrated in Figure 2. In Figure 2(a) burst 1 through burst 6 are scheduled to the latest available wavelength in the given order. With this, burst 7 arriving at time t_a cannot be scheduled. Figure 2(b) shows that burst 7 can be scheduled if FDL reassignment is used. When burst 6 is successfully scheduled neither burst 4 nor burst 5 can be rescheduled to the current wavelength C_3, i.e., no wavelength reassignment can take place. However, if burst 5 can be delayed by using a free FDL, it can be rescheduled to wavelength C_3 (the void formed at C_3 is smaller than void at C_2) and burst 7 can be scheduled to wavelength C_2.

C. Algorithm Description

The BR-WFR algorithm has two phases called scheduling and rescheduling phases. Phase 1 examines only the last burst on every wavelength to find the best wavelength C_p (latest available) and an FDL (if needed). Phase 2 takes place if phase 1 is successful and an existing last burst from $C_j (\neq C_p)$ is considered to be rescheduled to C_p to make a shorter void in order to pack the bursts tightly to prevent future burst dropping. Phase 2 may also reassign an FDL used by a burst only if the outgoing link is the last hop for the burst. Burst rescheduling with wavelength reassignment (BR-WR) in which only wavelength reassignment is carried out, is a special case of BR-WFR. The pseudo-code of the algorithm is given below. Here, the new burst is assumed to arrive at time t. The latest available time of wavelength channel C_i is given by t_i.

- **Phase 1 [Burst Scheduling]**
- 1. [Wavelength search] For every wavelength C_i compute the gap, m_i as $|t - t_i|$. If $t_i \leq t$, set the FDL flag to false otherwise set the flag to true to indicate that an FDL is needed.

- 2. [Wavelength assignment] Choose wavelength C_p with false FDL flag (i.e., no FDL needed) such that m_p is minimum among all m_i. Assign wavelength C_p to the new burst and call phase 2; if no such C_p exists, go to step 3.
- 3. [FDL assignment] Choose C_p which has minimum m_p. Search for shortest FDL which is at least m_p long. Assign this FDL and wavelength C_p to the new burst and call phase 2. If no such C_p exists, drop the burst and exit.
- **Phase 2 [Burst Rescheduling]**
- 1. [Wavelength search] For every wavelength C_i other than C_p determine if the last burst can be rescheduled to C_p; compute the void V_i that would be created at C_p after rescheduling; set the FDL flag to true if the burst needs to be delayed.
- 2. [Wavelength reassignment] Choose wavelength C_j with false FDL flag such that V_j is the minimum among all the wavelengths. If no such C_j exists, go to Step 4.
- 3. [Notification] Reschedule the last burst from C_j to C_p. Send a special control packet "NOTIFY" to notify the next node about the change of wavelength of the rescheduled burst. Exit.
- 4. [FDL reassignment] Choose C_k whose last burst is at the last hop and V_k is the minimum among all the wavelengths. Search for the shortest FDL which is at least V_k in length. If no such C_k exists, rescheduling fails; exit.
- 5. [Notification] Reschedule the last burst from C_k to C_p. Send a special control packet "NOTIFY" to notify the next node about the change of wavelength and the change of arrival time.

D. Computational Complexity

- Phase 1 - Scheduling : BR-WFR examines the information of one burst on each wavelength. Also, it searches through the FDLs once. Therefore, the worst case complexity is $O(W + F)$.
- Phase 2 - Rescheduling : This involves searching through all the wavelengths once for the best last burst to be rescheduled and also the FDLs once, if delay is needed. Therefore it requires $O(W+F)$ time. BR-WFR therefore has the same computational complexity as LAUC which also runs in $O(W + F)$ time.

Thus, the overall computational complexity of burst rescheduling at a node is given by $O(W + F)$.

E. Signaling Overhead

Additional signaling is needed whenever rescheduling is successful. This is to notify the next node about the change of wavelength by sending a "NOTIFY" packet. However, we have verified through simulation experiments that rescheduling does not incur significant signaling overhead for BR-WFR. This is because rescheduling is not feasible all the time. Out of all the bursts that invoke the rescheduling process, only a fraction of these are successful in rescheduling. Further, the "NOTIFY" message is sent on only one link. It is worth noting that no complex algorithm is executed upon receiving the "NOTIFY" packet. Also, the "NOTIFY" packet is short as it carries only two fields with wavelengths C_j and C_p. Alternatively, without sending extra signaling packet, these information can be piggybacked to the control packet. It therefore does not incur significant processing time and does not consume significant control channel bandwidth when compared to the computational complexity gain achieved over existing algorithms such as LAUC-VF.

F. Feasibility of Implementation

The implementation of rescheduling is feasible as explained below.

- Rescheduling of a burst is done before the burst actually arrives at a node. It does not affect any ongoing traffic.
- Wavelength reassignment does not change the time schedule of a burst on the outgoing link and on other links along the path. Only the next node needs to be notified about the change in the wavelength.
- FDL reassignment changes the time schedule. However, we restrict such a reassignment to a burst for which the outgoing link is the last hop. This means that only the next node (which is also the destination node for that burst) needs to be informed.

III. PERFORMANCE STUDY

In this section, the performance of the proposed BR-WFR algorithm is studied through simulation. Its performance is compared with that of LAUC and LAUC-VF. We also study the performance of the rescheduling algorithm BR-WR which differs from BR-WFR in that it carries out only wavelength reassignment but not FDL reassignment. For a 95% confidence level, the confidence interval is below 5% of the reported values. Two metrics, *burst dropping probability* and *performance improvement* are used to evaluate the performance. We analyze the dropping probability for two classes of requests, i.e., class-1 and class-2 where class-2 has a higher priority over class-1. In order to differentiate class-2 from class-1 requests, class-2 requests are given extra offset time (three times the average burst length used by class-1 requests plus the maximum FDL length) so that they can reserve resources in advance than class-1 requests [4]. Let the dropping probability of LAUC, BR-WFR (or BR-WR), and LAUC-VF be D_l, D_b, and D_v, respectively. The performance improvements of BR-WFR over LAUC and LAUC-VF over LAUC are calculated as $(D_l - D_b)/(D_l)$ and $(D_l - D_v)/(D_l)$, respectively.

To calculate the signaling overhead incurred by BR-WFR, consider a network node where x bursts of the total bursts arrived have been successful in scheduling. Out of x bursts that have invoked rescheduling, only a fraction, say, k' has been successful in rescheduling. The signaling overhead is then calculated as $\frac{k'x}{x} = k'$.

We use a randomly generated network with 32 nodes and 82 bidirectional links. Every node can be an ingress or egress node and each link has 8 wavelengths. A burst is routed from an ingress node to an egress node along the shortest path. Burst arrival pattern is based on a Poisson process. A burst is equally likely to be a class-1 or class-2 burst. The burst durations are exponentially distributed with a mean (L) of 10 μs. The control packet processing time (Δ) is assumed to be 1 μs. The FDL length is measured in units of μs. These values are chosen so that the ratio $\frac{L}{\Delta}$ is greater than the ratio of the number of wavelengths used for data and control traffic. This will ensure that the control packet is processed before its data burst is transmitted on a link even at full link utilization [3]. All the nodes are equally probable to be a destination node for a burst.

A. Effect of Traffic Loading

The dropping probabilities for the overall, class-1, and class-2 requests are shown in Figures 3, 4, and 5, respectively for different arrival rates (per node per μs) and an FDL buffer size of 10. As the arrival rate increases, the dropping probability for all the algorithms increases. LAUC experiences the highest dropping probability. BR-WFR and BR-WR perform better than LAUC and close to LAUC-VF at low traffic load. The performance of all the algorithms are similar for the class-2 high-priority requests as shown in Figure 5. This is due to the fact that a class-2 high-priority request has a higher offset time, which makes it more likely to be assigned a wavelength channel. Moreover, it is not affected by any void because of the long offset time used. Therefore, not much improvement can be obtained by using LAUC-VF or our proposed algorithms. Also, we observe that as expected class-2 requests perform far better than class-1 requests.

The performance of BR-WFR and BR-WR are better than that of LAUC for traffic loads ranging from low to high and are closer to that of LAUC-VF at low traffic load. Since in practice, networks usually operate at low traffic loads (smaller than 5% dropping probability), our algorithms are useful . This is shown in Figure 6, 7, and 8 where the performance improvements for the overall, class-1, and class-2 requests for BR-WFR, BR-WR and LAUC-VF over LAUC are presented. Figure 8 shows that the dropping probability for all the algorithms over LAUC have similar performance for the class-2 high-priority traffic. This is due to the large initial offset time used for these high-priority traffic, as explained in the previous paragraph.

BR-WFR performs better than BR-WR at low traffic load as shown in Figures 6, 7, and 8. With both wavelength reassignment and last-hop FDL reassignment, the chances of rescheduling bursts are higher and bursts are packed tighter. However, as the burst arrival rate increases, difference in the performance between the two algorithms diminishes. This is because, the possibility of finding eligible bursts for rescheduling decreases with increasing traffic load. Through simulation, we observed that the signaling overhead incurred by BR-WFR and BR-WR to be about 27% and 23%, respectively. The signaling packet is small in size as it needs to carry only the wavelength-change information to the next node. Therefore, the extra signaling packets would not take up much bandwidth on the signaling channel.

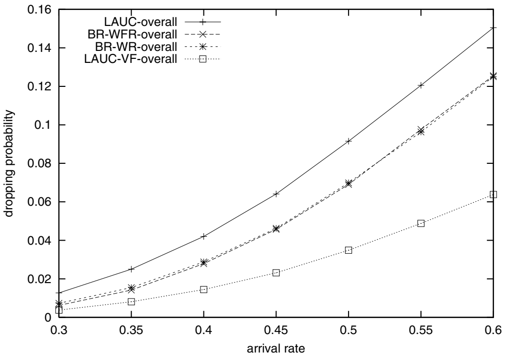

Fig. 3. Performance of overall (class-1 and class-2) bursts for varying traffic load.

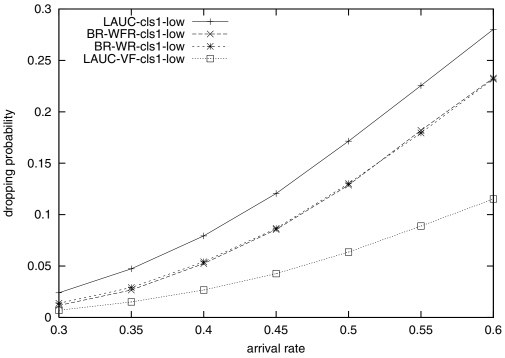

Fig. 4. Performance of class-1 bursts for varying traffic load.

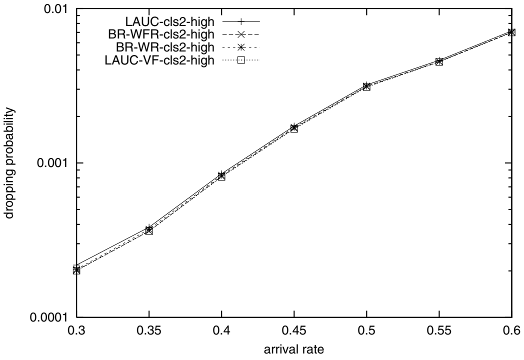

Fig. 5. Performance of class-2 bursts with varying traffic load.

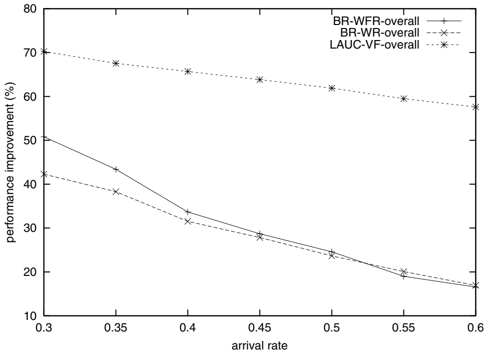

Fig. 6. Performance improvement achieved by BR-WFR, BR-WR, and LAUC-VF over LAUC for overall bursts for varying traffic load.

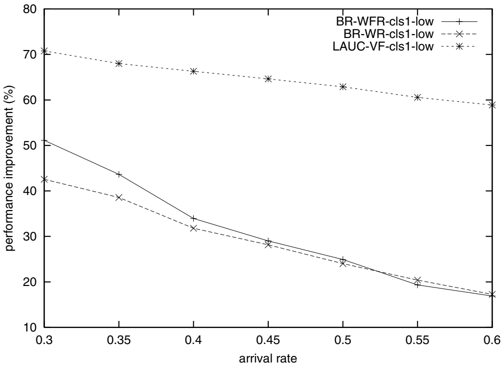

Fig. 7. Performance improvement achieved by BR-WFR, BR-WR, and LAUC-VF over LAUC for class-1 bursts with varying traffic load.

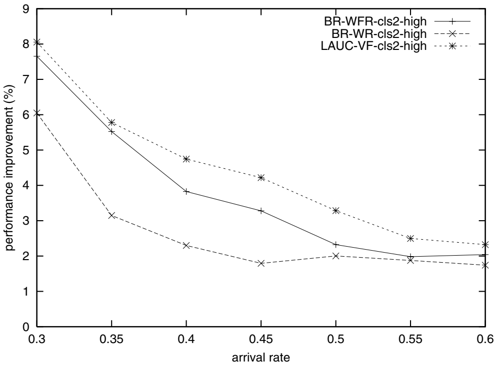

Fig. 8. Performance improvement achieved by BR-WFR, BR-WR, and LAUC-VF over LAUC for class-2 bursts with varying traffic load.

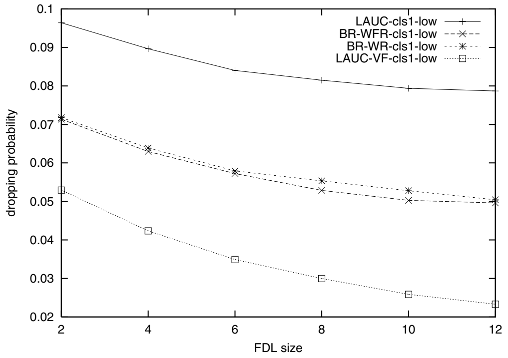

Fig. 9. Performance of class-1 bursts for varying FDL size.

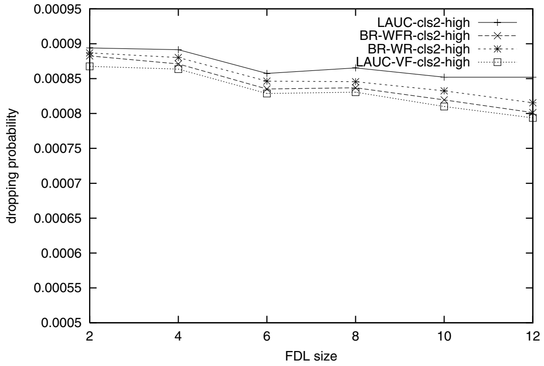

Fig. 10. Performance of class-2 bursts for varying FDL size.

B. Effect of FDL Buffer size

We study the performance of BR-WFR and BR-WR for varying lengths (or size) of FDL buffer. The dropping probabilities of the algorithms with increasing FDL size for the class-1, and class-2 requests are shown in Figures 9, and 10, respectively, for an arrival rate of 0.4. In general, the dropping probability decreases with the increasing FDL size because with a long delay the possibility of finding a free wavelength increases. For the range of FDL size from 2 to 12, the signaling overhead incurred by BR-WFR and BR-WR algorithms fall in the range of 23% to 28% and 20% to 23% respectively.

IV. CONCLUSIONS

We have proposed burst rescheduling for fast and improved scheduling in WDM optical burst switching networks. Burst rescheduling uses two mechanisms known as wavelength reassignment and last-hop FDL reassignment. We have addressed the implementation feasibility of burst rescheduling. Based on the above rescheduling mechanisms we have developed a computationally simple algorithm called BR-WFR. Through simulation experiments, we have demonstrated that the performance of the proposed algorithm is good for a range of traffic loads and is close to that of the complex LAUC-VF void-filling algorithm under light loads. Also, we note that the additional signals are very short and the bandwidth consumed and processing time due to them are small. The signaling overhead incurred by the proposed algorithm has been observed to be less significant when compared to the computational complexity gain achieved over LAUC-VF.

ACKNOWLEDGEMENT

This work was in part supported by the National University of Singapore Research Grant R-263-000-173-112.

REFERENCES

[1] J. S. Turner, “Terabit burst switching,” *J. High Speed Networks*, vol. 8, pp. 3-16, 1999.

[2] C. Qiao, M. Yoo, “Choices, features and issues in optical burst switching,” *Optical Network Magazine*, vol. 1, no. 2, pp. 37-44, April 2000.

[3] Y. Xiong, M. Vandenhoute, H. C. Cankaya, “Control architecture in optical burst-switched WDM networks,” *IEEE Journal on Selected Areas in Communications,* , vol. 18, no. 10, pp. 1838-1851, October 2000.

[4] M. Yoo and C. Qiao, “QoS performance in IP over WDM networks,” *IEEE Journal on Selected Areas in Communications*, vol. 18 no. 10, pp. 2062-2071, October 2000.

[5] L. Xu, H. G. Perros, and G. Rouskas, “Techniques for optical packet switching and optical burst switching,” *IEEE Communications Magazine*, pp. 136-142, January 2001.

[6] C. Guillemot *et al.*, “Transparent optical packet switching: The European ACTS KEOPS project approach,” *IEEE/OSA Journal of Lightwave Technology*, vol. 16, no. 12, pp. 2117-2134, December 1998.

[7] D. K. Hunter *et al.*, “WASPNET: A wavelength switched packet network,” *IEEE Communications magazine*, vol. 37, no. 3, pp. 120-129, March 1999.

[8] C. Qiao, “Labeled optical burst switching for IP-over-WDM integration,” *IEEE Communications Magazine*, pp. 104-114, September 2000.