

# Schedule Burst Proactively for Optical Burst Switching Networks

Jikai Li, Chunming Qiao

Department of Computer Science and Engineering  
State University of New York at Buffalo

**Abstract**— Optical Burst Switching (OBS) is a promising paradigm for the next-generation Internet infrastructure. In OBS, a key problem is to schedule bursts on wavelength channels with both fast and bandwidth efficient algorithms so as to reduce burst loss. To date, most scheduling algorithms avoid burst contention locally (or reactively). In this paper, we propose several novel algorithms for scheduling bursts in OBS networks with and without wavelength conversion capability. Our algorithms try to proactively avoid burst contention likely to occur at downstream nodes. The basic idea is to serialize the bursts on an outgoing link to reduce the number of bursts that may arrive at downstream nodes simultaneously (and thus reducing the burst contention and burst loss probability at downstream nodes). This can be accomplished by judiciously delaying locally assembled bursts beyond a pre-determined offset time at an ingress node using the electronic memory. Compared with the existing algorithms, our proposed algorithms can significantly reduce the loss rate while ensuring that maximum delay of a burst does not exceed its prescribed limit.

## I. INTRODUCTION

To meet the increasing bandwidth demands and reduce the costs associated with electronic switching/routing, several optical switching paradigms have been under intensive research. Of all these paradigms, optical circuit switching is relatively easy to implement but lacks flexibility to cope with the fluctuating traffic and the changing link state; Optical Packet Switching (OPS) is conceptually ideal, but the required optical technologies such as optical buffer and optical logic are too immature for it to happen anytime soon. An alternative approach called Optical Burst Switching (OBS) that combines the best of optical circuit switching and optical packet switching was proposed [1] [2] [3], and has received increasing amount of attention from both academia and industry worldwide [4], [5], [6], [7], [8].

In an OBS network, an ingress OBS node assembles client data units (e.g. IP packets) into bursts and sends out a corresponding control packet for each data burst. This control packet is delivered out-of-band and leads the data burst by an offset time. The control packet carries, among other information, the offset time at the next hop, and the burst length. At each intermediate node along the way from the ingress node to the egress node, the control packet reserves necessary resources (e.g., bandwidth on a desired output channel) for the following burst, which will be disassembled at the egress node.

In OBS, control packet processing and data burst transport are separated both in time and wavelength domains, which enables OBS to take advantage of high capacity of optical fibers and sophisticated control of electronics simultaneously. Com-

pared with optical circuit switching, OBS supports sub- $\lambda$  granularity and statistical multiplexing. Compared with OPS, OBS does not require stringent synchronization and Fiber Delay Lines (FDLs). Moreover, OBS networks support data transparency.

A popular bandwidth reservation protocol is Just-Enough-Time (JET). In JET, a control packet reserves a output wavelength channel for a period of time equal to the burst length, starting at the expected burst arrival time. If the reservation is successful, the control packet adjusts the offset time for the next hop, and is forwarded to the downstream node; otherwise, the burst is blocked and will be discarded if there is no Fiber Delay Lines (FDLs).

Given the fact that OBS uses one-way reservation protocols such as JET, and that a burst can not be buffered at any intermediate node due to the lack of optical RAM (a FDL, if available at all, can only provide a limited delay and contention resolution capability), burst loss performance is a major concern in OBS networks. Several methods can be applied to reduce loss rate, such as constraint-based routing, traffic shaping, deflection etc. In this work, our major interest is to design scheduling algorithms that can avoid possible burst contention.

The rest of the paper is organized as follows. Section II gives an overview of several existing algorithms, which includes Horizon, LAUC-VF, Min-SV and PWA. In Section III, our new algorithms that try to avoid burst contention pro-actively will be introduced. Section IV discusses several issues related to traffic engineering when our algorithms are deployed. Section V shows simulation results, followed by concluding remarks in Section VI.

## II. PREVIOUS WORK

In an OBS network, upon receiving the control packet, an intermediate node must quickly decide how to schedule the incoming burst. Which wavelength should be reserved for the burst is determined by a scheduling algorithm. Throughout this work, we use JET to reserve bandwidth, and hence, if burst  $b$  arrives at time  $r$  and leaves at time  $f$ , JET reserves a  $\lambda$  from time  $r$  to  $f$  only. If the incoming burst is successfully scheduled on one wavelength, the control packet and data burst are forwarded to the downstream node with an modified offset time between them; otherwise, the control packet and burst are dropped.

There are two facts that make OBS scheduling problem challenging. One is that bursts may not arrive one after another (i.e. without any interval in between), and hence each channel

is likely to be broken into fragments, separated by idle intervals (also called voids). The voids could be wasted if an inefficient scheduling algorithm is used. Another fact is that different bursts may have different initial offset times, and the offset time of a given burst changes along the path from its ingress node to its egress node. The second fact implies that even if burst  $b_i$  arrives after burst  $b_j$ , burst  $b_i$ 's corresponding control packet  $h_i$  may arrive before  $b_j$ 's corresponding control packet  $h_j$ . These two facts make it very difficult for scheduling algorithms to determine which available wavelength is the "best" for the incoming burst. Several algorithms have previously been studied for solving the channel scheduling problem.

- Turner designed the Horizon Scheduling algorithm [9]. In this algorithm, a scheduler only keeps track of the so-called horizon for each channel - the time after which no reservation has been made on that channel. The scheduler assigns each arriving data burst to the channel with the latest horizon as long as it is still earlier than the arrival time of the data burst. The horizon algorithm is relatively simple and has a reasonably good performance in terms of its execution time. However, the horizon scheduling algorithm results in a low bandwidth utilization and a high loss rate. This is due to the fact that the horizon algorithm simply discards all the voids.
- Xiong *et al.* [4] proposed a channel scheduling algorithm, called LAUC-VF (Latest Available Unused Channel with Void Filling). LAUC-VF keeps track of all voids (closed intervals), as well as the horizons which can be considered as open intervals, and assigns the interval with the latest starting time that is still earlier than the arrival time of the arriving burst. This yields a better bandwidth utilization and loss rate than the Horizon Algorithm.
- Xu *et al.* [10] proposed several efficient algorithms for selecting channels for incoming data bursts using different criteria. A representative algorithm proposed in [10] is Minimum Starting Void (Min-SV). The data structure of Min-SV is constructed by augmenting a balanced binary search tree. By doing so, Min-SV enjoys loss rate as low as LAUC-VF and processing time as low as Horizon. For detailed information, please refer to [10].
- To avoid possible burst contention at downstream nodes, Wang *et al.* [11] proposed a Priority-based Wavelength Assignment (PWA) algorithm for ingress nodes. In PWA, each ingress node keeps a wavelength priority database for every destination node. When the ingress node schedules a burst, it searches the wavelength priority database. If the wavelength with highest priority is available, the burst is scheduled to this wavelength; otherwise, the algorithm checks the wavelength with the second highest priority. The priority of each wavelength is updated dynamically according to its burst loss profile. Simulation shows PWA can reduce loss rate in a OBS network. Unfortunately, PWA is only meaningful in a OBS network without wavelength conversion capability.

In the next section, we will describe our proposed algorithms which try to schedule bursts in a proactive way to avoid possible burst loss at downstream nodes. Compared with Horizon and LAUC-VF, Our proposed algorithms have a much lower loss

rate. Compared with PWA, our algorithms is applicable to OBS networks with or without wavelength conversion.

### III. PROPOSED WORK

Our idea is based on the observation that if the total number of simultaneously arriving bursts exceeds the number of wavelength, burst loss will be inevitable (assume there is no FDLs). To characterize such a situation, we use *overlapping degree* to describe the number of bursts that arrive at one link simultaneously. The proposed algorithms try to reduce the overlapping degree at edge nodes by using electronic buffer.

The overlapping degree is defined as follows. Given a link  $l$  and time  $t$ , if there are multiple bursts arriving at link  $l$  at time  $t$ , we say these bursts are *overlapped*. The total number of data bursts that arrive at link  $l$  at time  $t$  is the overlapping degree.

Clearly, the value of overlapping degree is directly related to the burst loss. The larger the overlapping degree is, the more likely an incoming burst will be dropped.

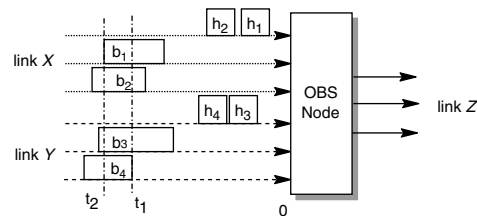


Fig. 1. Multiple bursts arrived

Fig. 1 shows an example where LAUC-VF fails to schedule all bursts successfully. In this example, the intermediate OBS node has two incoming links and one outgoing link, each link has two data wavelengths. Assume that four data bursts,  $b_1, b_2, b_3$  and  $b_4$ , arrive from the incoming links and these four data bursts are overlapped from time  $t_1$  to time  $t_2$  (in other word, the overlapping degree of period  $(t_1, t_2)$  is 4). Then, two out of the four bursts will be dropped if the OBS node does not have FDLs or all FDLs have already been used by other bursts.

Note that if we can reduce such overlapping by delaying burst  $b_2$  at an upstream node so it arrives at link X after  $b_1$  and similarly delaying  $b_4$  so it arrives after  $b_3$ , the OBS node will not drop any burst as shown in Fig. 2.

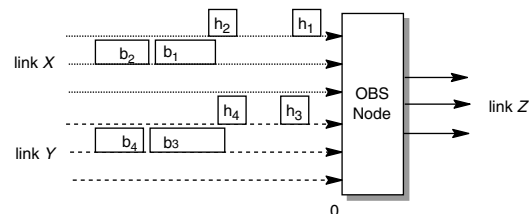


Fig. 2. Burst overlapping is reduced

Based on the above observation, we propose several scheduling algorithms that can reduce the overlapping degree, these algorithms are collectively called as Burst Overlap Reduction Algorithm (BORA). Without loss of generality, we assume that there are multiple Optical Burst Switching (OBS) [3] paths passing a given link  $l$  in a OBS network. Each OBS path has

one ingress node and one egress node. In order to reduce burst overlap on link  $l$ , each OBS path should try to reduce the burst overlap on itself. In an OBS network without FDL, the core nodes cannot delay some bursts to reduce the burst overlap, however, ingress nodes with electronic buffer can do this by using a well designed scheduling algorithm.

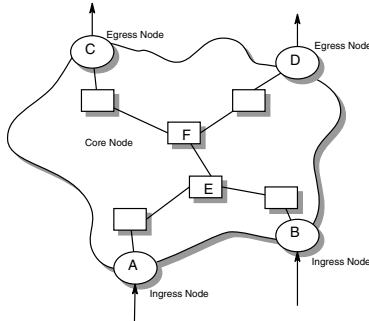


Fig. 3. OBS Network Architecture

Fig. 3 shows an OBS network example. In this example, all packets are assembled at nodes  $A$  and  $B$ , and dissembled at nodes  $C$  and  $D$ . Suppose that bursts  $b_1$  and  $b_2$  in Fig. 1 are from ingress node  $A$ ,  $b_3$  and  $b_4$  are from ingress node  $B$ . For node  $A$ , in order to reduce burst overlap, it delays the transmission of burst  $b_2$  until burst  $b_1$  is sent out. Similarly, when a new burst is assembled and ready to be sent at node  $B$ , it will be delayed. Fig. 2 shows the result of such delayed transmission. One limitation of this simple algorithm is that the delay time introduced by this algorithm could be too large. To limit the extra delaying time, we modify the algorithm to make it work with a bounded delay for each burst, say  $\alpha$ .

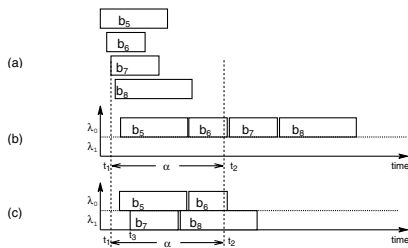


Fig. 4. Pack Bursts at an Ingress Node (Fixed-order searching)

Fig. 4 is an example illustrating how an ingress node schedules a set of locally generated bursts. Fig.4 (a) shows four bursts,  $b_5, b_6, b_7$  and  $b_8$ , arriving at the scheduler sequentially. Fig. 4 (b) shows the scheduling result when ingress node ignores the delay time of each burst. In Fig. 4 (b), the delay time of burst  $b_7$  and  $b_8$  exceeds maximum delay time  $\alpha$ . Fig. 4 (c) shows the scheduling result when ingress node takes the maximum delay time into consideration. More specifically, after bursts  $b_5$  and  $b_6$  have been assigned to  $\lambda_0$ , ingress node tries to schedule burst  $b_7$  to  $\lambda_0$ , but since the delay time of burst  $b_7$  will be larger than the maximum allowed delay time  $\alpha$ , ingress node assigns  $\lambda_1$  to  $b_7$ . Similarly, when ingress node schedules  $b_8$ , it checks  $\lambda_0$  first. Again, since the delay time would exceed  $\alpha$ , ingress node tries  $\lambda_1$  and reserves it for  $b_8$ .

### A. Fixed Order Search

In the above description, the ingress node always searches wavelength channels using a fixed order and the algorithm stops either when a suitable channel is found which satisfies the maximum delay requirement, or all channels have been checked and none of them satisfies the requirement.

The above algorithm can work with or without utilizing voids that may exist on each outgoing wavelength channel of each ingress node. More specifically, if the ingress node schedules the locally generated (assembled) bursts using voids of each channel, we say it is Burst Overlapping Reduction Algorithm with Fixed-ordered Searching and Void filling (BORA-FS-VF). If the algorithm schedules the locally generated bursts to the open interval (also called horizon) of each channel only (without void filling), we simply refer to it as BORA with Fixed-ordered Searching or BORA-FS.

### B. Destination-based Order Search

According to the above discussion, we can see that the fixed-order channel searching reduces the overlapping degree generated by the locally generated bursts on the outgoing links that are connected to an ingress node. The idea behind this fixed order searching is that by reducing the overlapping degree on the first hop (link) of each OBS path, the overlapping degree on each intermediate link (although bursts belongs to different OBS paths) is also reduced.

In the following, we modify the fixed-order channel searching algorithms into destination-based searching order algorithms that can reduce loss rate further by taking the routing information of different OBS paths into consideration while scheduling the locally generated bursts.

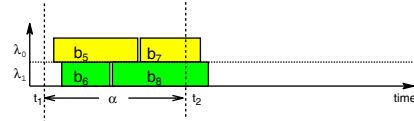


Fig. 5. Pack Bursts Using Destination-based Searching Order

Consider the example shown in Fig. 4 again. If we use BORA-FS (or BORA-FS-VF), bursts  $b_5$  and  $b_6$  are scheduled onto the same channel, burst  $b_7$  and  $b_8$  are assigned to another channel. In this example, if burst  $b_5$  and  $b_7$  take OBS path  $L_0$ , and burst  $b_6$  and  $b_8$  take OBS path  $L_1$ , then for both  $L_0$  and  $L_1$ , the maximum overlapping degree among these four bursts is 2, even on the links that  $L_i$  and  $L_j$  do not share.

If we modify the above algorithm as follows, the overlapping degree can be reduced. When a burst arrives, scheduler will first search the channel(s) preferred by the OBS path the burst will follow (called *home channel(s)*). Let the home channel(s) set be denoted by  $H_{I,J}$ , where  $I$  is the ingress node of the OBS path and  $J$  is the egress node of the OBS path). Only if the burst cannot be accommodated by its home channel(s), scheduler will check other channels. The problem of how to determine home channel(s) for each OBS path [3] is, for the most part, an orthogonal issue, and will be addressed separately.

Fig. 5 shows the scheduling result assuming the home channel for  $L_0$  is  $\lambda_0$  and that for  $L_1$  is  $\lambda_1$ . The main difference

between Fig. 4 and Fig. 5 is due to the fact that when the OBS node schedules  $b_6$ , it starts searching from  $\lambda_1$  (its home channel) instead of  $\lambda_0$ , and schedules the burst on  $\lambda_1$ . Subsequently, it will schedule  $b_7$  on  $\lambda_0$  and  $b_8$  on  $\lambda_1$ . Such a scheduling does not reduce the loss rate on the link connected to the ingress node, but reduces the overlapping degree among the four bursts to 1 on the downstream links that are not shared by  $L_0$  and  $L_1$  and can thus reduce loss rate.

Similar to the definition of BORA-FS, if we schedule the locally generated bursts beyond the horizon of each channel only and check the channels in an order according to the destination of the burst, we name it as BORA-DS. If we schedule the locally generated bursts using the voids of each channel (instead of only horizon), and check the channels in an order according to the destination of the burst, we name it as BORA-DS with Void Filling or BORA-DS-VF.

#### IV. TRAFFIC ENGINEERING RELATED ISSUES

In this section, we discuss some issues related to traffic Engineering when deploying the BORA-DS algorithms. Note that the example shown in Fig. 5 has only two channels and two OBS paths. In a real OBS network, it is likely that each network has dozens of nodes and each link has hundreds of wavelengths. To reduce the overlapping degree (or loss rate) on each link, we need to reduce not only the overlap among the bursts belonging to the same OBS path, but also all the OBS paths using the link. The latter can be achieved by a load-balancing OBS path routing algorithm as a part of traffic engineering effort. In the following discussion, we describe how to select non-home-channels (in case a burst cannot be scheduled to any of the home channels assigned to its corresponding path) such that “interference” to other OBS paths is minimized. We assume that shortest path routing algorithm is applied, the ideas proposed in the following can be extended to the other routing algorithms.

We assume that each ingress node in a OBS network uses a spanning tree to maintain the shortest path to every egress node as shown in Fig. 6. The burst sent out from the ingress node will only pass the links of this spanning tree. Therefore, the scheduling problem becomes how to minimize the burst overlapping degree on each edge of the spanning tree.

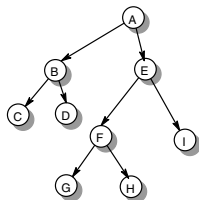


Fig. 6. Spanning tree rooted at node A

If the traffic on each OBS path is steady, then with a appropriate home channel assignment, the scheduling algorithm only need to use the home channel(s). When the traffic on each OBS path is fluctuating, it is possible that when a big number of bursts for one OBS path  $L$  arrives in a short period of time, and all the home channels(s) assigned to OBS path  $L$  are

occupied although other channels are available. To avoid dropping data bursts and utilize bandwidth efficiently, the ingress node can schedule a burst to another OBS path’s home channel. However, this may disturb other OBS paths. To minimize such disturbance, we need to select non-home-channels carefully.

Consider the spanning tree rooted at node  $A$  in Fig. 6. When a burst destined to  $G$  arrives at node  $A$ , node  $A$  will assign it to  $G$ ’s home channel(s) if possible; otherwise, we propose that node  $A$  try node  $F$ ’s (or node  $H$ ’s) home channels since node  $A$ ,  $F$  and  $H$  share two common links  $\langle A, E \rangle$  and  $\langle E, F \rangle$ . The motivation is that by scheduling the burst in this way, it can prevent another burst to  $F$  (or  $H$ ) being scheduled at the same time, thus keeping the overlapping degree on links  $\langle A, E \rangle$  and  $\langle E, F \rangle$  unchanged.

This algorithm can be formally described as follows. Given a burst destined to node  $J$  and a spanning tree root at the ingress node, the ingress node first reserves the home channel(s) of  $J$ , if successfully, algorithm stops; otherwise, the ingress node checks the home channels that belongs to the children nodes of  $J$ . If successful, the algorithm stop; otherwise, the ingress node checks the home channels of  $J$ ’s parent node, and  $J$ ’s sibling nodes, and then, the home channels of all other nodes (without having to follow any specific order). This process continues until one channel is found, or all channels have been checked and the burst has to be dropped.

#### V. EXPERIMENTAL RESULTS

This section presents our experimental results on the proposed BORA algorithms and their comparisons with the existing algorithm LAUC-VF. Our experiments focus on examining the loss rate. The network topology used in this simulation is the 14-node NSFNET. We assume that both burst length and control packet inter-arrival time follow the Pareto distribution, and the bandwidth reservation is done by using JET. The number of data channels on each link is 13 and the bandwidth of each wavelength is 10Gbps and the average burst duration is  $0.1ms$ . In our simulations, we assume that each node is both an ingress node and a core node. The OBS paths used in our simulation are computed by using a shortest path algorithm. Each node generates the same average amount of traffic which is a fraction of the link capacity. In our simulation, this average amount varies from 0.3 to 0.9. The traffic generated by each node is randomly distributed among all other nodes. This network provides full wavelength conversion.

##### A. LAUC-VF vs. BORA-FS-VF

To compare our BORA-FS-VF algorithm with LAUC-VF, we use these two algorithms to schedule locally generated bursts and compare the loss rate resulted from using these two algorithms. We conduct two sets of simulations, one investigating the relationship between load and loss rate, and the other studying the relationship between the maximum delay bound ( $\alpha$ ) value and loss rate. In both simulations, we use LAUC-VF to schedule transit bursts (i.e. bursts generated at an upstream node that need to bypass this node).

Fig. 7 shows the loss rate of LAUC-VF algorithm and BORA-FS-VF algorithm when  $\alpha$  is  $1.0ms$ . We can see that the

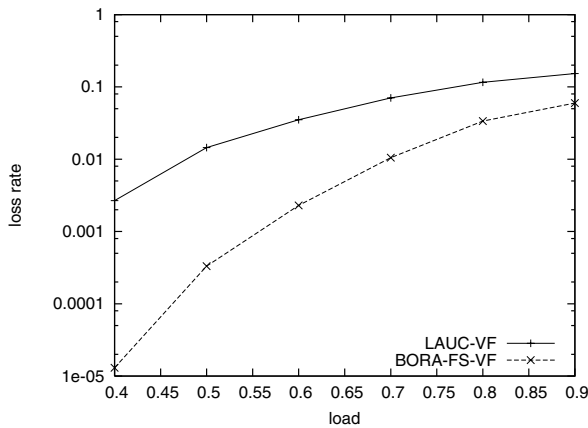


Fig. 7. Loss rate of new algorithm and normal algorithm

loss rate of BORA-FS-VF is much lower than that of LAUC-VF. In particular, when the traffic load is low to medium, the loss rate of our BORA-FS-VF is about 100 times lower than that of the LAUC-VF. When the network is heavily loaded, the system becomes sensitive to the variation of load and some links can be easily overloaded as a result of fluctuation. In this situation, the effect of BORA-FS-VF is mitigated. But even in this heavily loaded situation, the loss rate of our BORA-FS-VF algorithm is still much lower than that of LAUC-VF.

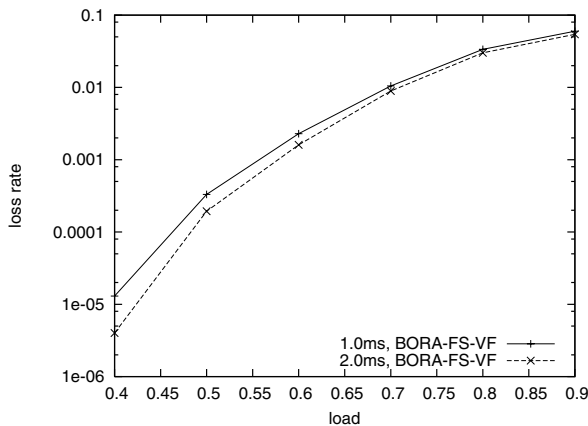


Fig. 8. Effect of different  $\alpha$  value

Different  $\alpha$  value has different effect on the loss rate. Fig. 8 shows the loss rate of BORA-FS-VF algorithm when  $\alpha$  is 2.0ms and 1.0ms. In this figure, we observe that the performance of 2.0ms is better than that of 1.0ms. The reason is that the larger  $\alpha$  has a more significantly smoothing effect, which reduces the burst overlapping degree.

### B. BORA-FS-VF vs. BORA-DS-VF

We have also compared BORA-FS-VF with BORA-DS-VF when  $\alpha$  is 1.0ms. Fig. 9 shows the loss rate of BORA-FS-VF and BORA-DS-VF. It clearly indicates that BORA-DS-VF has a better performance under any traffic load (note that the Y-axis is in log scale, so the difference between two algorithms is quite significant).

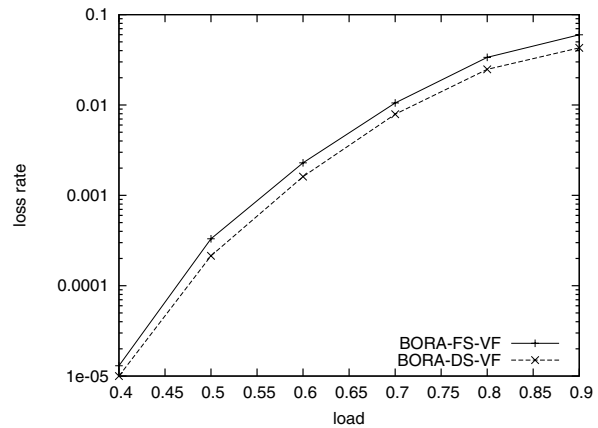


Fig. 9. Loss rate of BORA-FS-VF and BORA-DS-VF

## VI. CONCLUSION

In this paper, we have presented several novel channel scheduling algorithms in OBS networks called BORA. Unlike all the existing algorithms except PWA, the proposed BORA family of algorithms tries to reduce burst overlap at the output links, and thus avoid burst contention and burst loss proactively at downstream nodes. We have shown that BORA can reduce the burst loss rate significantly when compared to LAUC-VF.

Although we have only studies the performance of BORA at the edge nodes, the idea can be applied at core nodes having FDLs as well. More specifically, at a core node with FDLs, even though there is no burst contention, the core node can take advantage of the buffering capability provided by FDLs and reorganize the transit data bursts into a better sequence, and reduce the loss rate at downstream nodes as a result of reduced burst overlapping degree.

## REFERENCES

- [1] M. Yoo and C. Qiao, "A high speed protocol for bursty traffic in optical networks," *SPIE's All-Optical Communication Systems: Architecture, Control and Protocol Issues*, vol. 3230, pp. 79–90, Nov, 1997.
- [2] C. Qiao and M. Yoo, "Optical burst switching(OBS)-a new paradigm for an optical Internet," *Journal High Speed Networks*, vol. 8, pp. 69–84, 1999.
- [3] C. Qiao, "Labeled optical burst switching for Ip-Over-WDM integration," *IEEE Communications Magazine*, vol. 38, no.9, pp. 104–114, 2000.
- [4] Y. Xiong, M. Vandenhoude, and H.C. Cankaya, "Control architecture in optical burst-switched wdm networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1838–1851, 2000.
- [5] J.Turner, "Terabit burst switching," *Journal High Speed Networks*, vol. 8, pp. 3–16, 1999.
- [6] L. Xu, H. Perros, and G. Rouskas, "Techniques for optical packet switching and optical burst switching," *IEEE Communications Magazine*, vol. 39,no.1, pp. 136–142, Jan. 2001.
- [7] A. Detti and M. Listanti, "Impact of segments aggregation on tcp reno flows in optical burst switching networks," in *IEEE Infocom 2002*, pp. 1803–1812.
- [8] C. Hsu, T. Liu, and N. Huang, "Performance analysis of deflection routing in optical burst-switching networks," in *IEEE Infocom 2002*, pp. 66–73.
- [9] J. Turner, "Terabit burst switching progress report (9/98-12/98)," in *Washington University at St.Louis Technical Report*, 1998.
- [10] J. Xu, C. Qiao, J. Li, and G. Xu, "Efficient Channel Scheduling Algorithms in Optical Burst Switched Networks," in *IEEE Infocom 2003*, to appear.
- [11] X.Wang, H.Morikawa and T.Aoyama, "Priority-based wavelength assignment algorithm for burst switched photonic networks," *OFC 2002, THGG108*, pp.765-767