A Flexible Contention Resolution Scheme for QoS Provisioning in Optical Burst Switching Networks

Ashok K. Turuk [a], Rajeev Kumar [b,∗,1]

[a]*Department of Computer Science and Engineering, National Institute of Technology, Rourkela, Orissa 769 008, India*

[b]*Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur WB 721 302, India*

Abstract

Contention resolution is a major issue in bufferless optical burst switching (OBS) networks. The existing contention resolution schemes consider priority and arrival time to resolve contention. For most multimedia applications priority and delay are key parameters for QoS provisioning. In this paper, we propose a new signaling protocol for reducing contention in OBS networks and consider three parameters, namely, priority of the burst, number of hops traversed, and burst size into account to resolve contention. The source node in the proposed protocol can be informed of the contention up to halfway along the path of the burst, and thus, can reschedule the burst accordingly. The scheme is adaptable to both prioritized and delay constrained traffic. We call the scheme OBS-$Flex$. For selecting a data channel, we propose three channel selection algorithms, namely, Least Recently Used (LRU), First Fit (FF), and Priority Set (PS). We simulate OBS-$Flex$ and compare with preemptive priority just-enough-time (PPJET) contention resolution scheme. We show that OBS-$Flex$ outperforms PPJET in terms of burst loss rates. For simulation, we have considered Poisson and bursty traffic models.

Key words:
Optical burst Switching, wavelength division multiplexing, QoS, contention resolution, blocking probability, channel selection algorithm, Poisson and bursty traffic.

∗ Corresponding author.
Email addresses: akturuk@nitrkl.ac.in (Ashok K. Turuk), rkumar@cse.iitkgp.ernet.in, raj@iitk.ac.in (Rajeev Kumar).
[1] Present address: Department of Computer Science and Engineering, Indian Institute of Technology, Kanpur UP 208 016, India

1 Introduction

There has been a phenomenal increase in the number of Internet users and the variety of Internet applications in recent years. This has resulted in exponential growth of Internet traffic and demanding a huge bandwidth at the backbone network. To meet this growing demand for bandwidth, wavelength division multiplexing (WDM) network has become the de-facto choice for the backbone network. IP over WDM networks have drawn much attention among researchers, and many integration schemes between IP and WDM layers have been proposed e.g., [1, 6, 17].

To carry IP traffic over WDM networks three switching technologies have been studied: optical circuit switching, packet switching and burst switching. Optical circuit switching and packet switching have their own limitations when applied to WDM networks. Circuit switching is not bandwidth efficient unless the duration of transmission is greater than the circuit establishment period [12]. It is shown that establishment of circuits (lightpaths) in optical networks is an NP-hard problem [3]. Many heuristics and approximation algorithms exist for establishing lightpaths in optical networks e.g., see [5] and the references therein. Packet switching is hop-by-hop store and forward scheme and needs buffering and processing at each intermediate node. It is flexible and bandwidth efficient. However, technology for buffering and processing in optical domain is yet to get mature for this scheme to commercialize. Fiber delay lines proposed in the literature provide limited buffer and are suitable only when delays are fixed.

In this context, optical burst switching (OBS) is emerging as a new switching paradigm for next generation optical networks. It combines features of both circuit and packet switching. As such there exists no formal definition of OBS; the features defined by Yoo and Qiao [20] for OBS have become the de-facto standards. OBS schemes are based on either one-way (for most cases) and two-way reservation protocols. The burst-size granularity (which lies between circuit and packet switching), separation of control and data bursts, one-way or two-way reservation scheme, and no optical buffering are the important characteristics of OBS paradigm. In a bufferless optical network one of the contending bursts is dropped. Therefore, burst loss that should be minimized in OBS networks is the key design parameter.

Several techniques have been proposed to reduce burst loss and provide QoS in OBS networks. In one-way reservation protocol data burst follows the control burst after a predetermined offset-time. Examples of such protocols are Tell-n-Go [13, 16], Just-Enough-Time (JET) [18, 19] and burst segmentation [14, 23]. Two-way reservation protocols require an explicit release of reserved resources [4, 15]. An example of a two-way reservation protocol is a Just-In-Time (JIT) [15] scheme. The offset time in one-way reservation protocol is taken to be the sum of processing delay of control packet at each intermediate node. This time is too short to reschedule the transmission in case of contention and the contending burst is dropped. In two-way reservation protocol, as in JIT, data burst is sent before receiving an acknowledgment, resources are reserved from the time request is received and remain reserved until a release message is received. Loss of a release message leads to the wastage of bandwidth. Since the burst is sent before receiving an acknowledgment, in case of contention it is dropped.

In Priority-Just-Enough-Time (PJET), Yoo et al. [21] assigned an additional offset time, in addition to the base offset time for each class of traffic to reduce burst loss. The higher priority traffic is assigned an additional offset in time. Kaheel and Alnuewiri [7] proposed a preemptive prioritized JET (PPJET) scheme. In PPJET, a higher priority request can preempt the reservation of lower priority request if their transmission has not started. This can, however, increase the loss of lower priority traffic significantly. Vokkarane et al. [14] proposed a prioritized burst segmentation approach. In their approach when a contention occurs the overlapped portion of the burst is dropped. This method needs a complex implementation technique to segment the burst for a drop. Zhang et al. [22] proposed a wavelength grouping and early drop scheme where a low priority traffic is dropped first in case of congestion in the network. In their scheme, bursts belonging to lower priority class are dropped intentionally with a pre-determined probability before possibly contending with the burst of a higher priority class. To provide guaranteed services, core routers have to maintain traffic statistics for each supported class of traffic.

There are few other studies too. For example, Lu et al. [10] proposed an intermediate node initiated reservation where an intermediate node along the path can initiate an reservation to reduce burst loss due to contention. Boudriga [2] assigned different delay units to each class of traffic in order to isolate higher priority class from the lower priority class. Lee and Griffith [9] presented traffic engineering technique to support QoS in optical Internet. The mechanism proposed by them tries to utilize the available wavelength efficiently in order to provide lower delays. Kim et al. [8] proposed deflection routing mechanism to reduce burst losses. They defined threshold function to reroute the contending bursts. In their scheme, deflected bursts may take longer path to reach and require large buffers at its destination. Most of the researchers have attempted to reduce blocking probability of different class of traffic in order to provide priority based services. To reduce burst loss different delays are pre-assigned to each class of traffic.

In all of the above mentioned protocols, in case of contention, one of the bursts is dropped. There is no way to reschedule the burst in case of contention either because the offset time is too short to reschedule as in one-way protocol, or the burst is sent before receiving an acknowledgment as in two-way protocol. In this paper, we present a new signaling protocol to reduce burst loss due to contention in OBS networks. The source node in the proposed scheme can be informed of the contention up to halfway along the path of the burst, and can thus, reschedule the burst accordingly. We call the proposed scheme OBS-$Flex$. We consider packet loss and number of hops traversed, in addition to priority, for resolving contention. The scheme is generic and can easily be adapted to satisfy delay constraints. The main aim of this work is to reduce blocking probability of the bursts arising due to resource contention at intermediate nodes as well as to meet the delay constraints of the delay sensitive traffic. The proposed OBS-$Flex$ guarantees that the burst succeeds when contention occurs up to halfway along the path; the contention should be resolved in accordance with satisfaction of QoS parameters. To select data channel in OBS-$Flex$, we propose three channel selection algorithms – (i) *Least Recently Used* (LRU), (ii) *First Fit* (FF), and (iii) *Priority Set* (PS). Channel selection algorithms are run at the ingress routers to select a data-channel for reservation and for subsequent transmission. We evaluate the proposed OBS-$Flex$ with the above three channel selection algorithms, and present results for Poisson and bursty traffic models.

Rest of the paper is organized as follows. Section 2 explains the proposed contention resolution technique including assumptions and notations used in this work. Channel selection algorithms are explained in Section 3. Simulation results are presented in Section 4 and compared with PPJET scheme. Finally, conclusions are drawn in Section 5.

2 Contention Resolution Scheme OBS-$Flex$

2.1 Assumptions and Notations

We model an optical network by means of an undirected graph $G(V, E)$ where V is the set of vertices (nodes) and E represents the set of links/edges in the network. Two types of nodes (hereafter, we use the terms node and router interchangeably), namely, edge and core routers are identified. Every edge router has $(n_e - 1) \times P$ electronic buffers where n_e is the number of edge routers, and P is the number of priority classes supported in the system. Each buffer belongs to a specific pair of priority class and an egress router. The core router has no buffer; this is a desirable feature of OBS networks. Besides processing and forwarding the control packet, core router has the capability of generating its own control packet. A core router acts as a transit router for data traffic. Thus, data traffic remains in optical domain from ingress to egress router. We consider propagation delay, t, to be the same between every pair of adjacent vertices in the graph G; this assumption simplifies the simulation. Though this is a highly restricted assumption, this is reasonable because main aim of the simulation is to demonstrate low burst loss in case of contention. Processing delay of the control packet at each router is assumed to be δ. Few more notations used in rest of the paper are defined below:

$original\ burst$: A burst for which resources are already reserved at the core router,

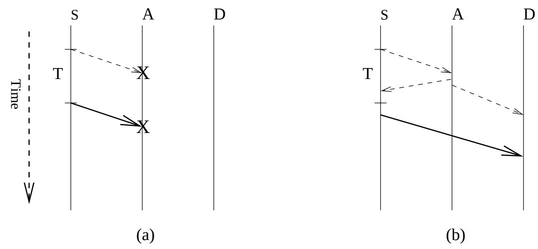

Fig. 1. Principle of OBS-$Flex$

$contending\ burst$: A burst whose reservation request has resulted in a resource contention at the core router,
$H^{sd}(z)$: Total number of hops for the request z between the source-destination pair (s, d), and
$H_i^{sd}(z)$: Remaining number of hops for the request z between the source-destination pair (s, d) at node i.

We identify the following three situations that can occur when an intermediate router receives a reservation request:

- *no contention* (NC):- when no contention occurs for resources at the intermediate core router.
- *contention resolved* (CR):- when a contention occurs at an intermediate core router i and for the contending burst's request $H_i^{sd}(z) > H^{sd}(z)/2$.
- *contention-not-resolved* (CNR):- when contention occurs at an intermediate core router i and for the contending burst's request $H_i^{sd}(z) > H^{sd}(z)/2$.

2.2 Proposed OBS-$Flex$ Scheme

In case of contention, the contending burst is dropped in OBS networks. The basis of this work is that if transmission of burst is delayed at the source for the duration of the contention period then the transmission of the burst would be successful. To delay the burst a control signal is to be sent to the source from the node where contention has taken place. The control signal should reach the source before the expiry of the offset time as shown in Fig. 1. In Fig. 1(a) contention has occurred at node A and the burst is dropped at node A. However, in Fig. 1(b) a control packet is sent from node A where contention has occurred and is received by the source S before the offset time T. Transmission of the burst is further delayed by the source S for the contention period. Thus, transmission of the burst is successful.

We use the above mechanism in the proposed OBS-$Flex$ to resolve contention. Offset time is taken to be the propagation delay between source-destination pair. We use two control packets: *Forward* (F) and *Reverse* (R) control packets. F-control packet is sent to reserve resources along the path, and R-control packet is

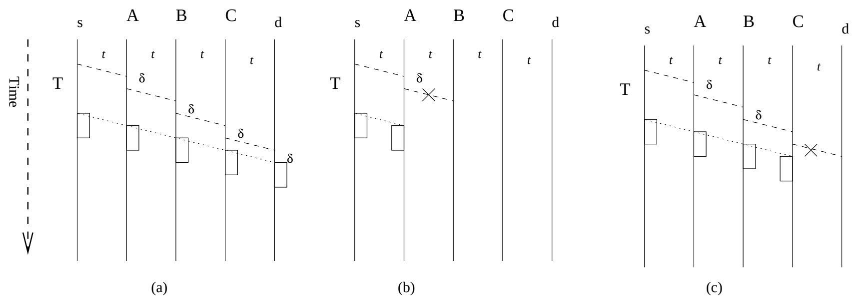

Fig. 2. Timing diagram of a burst switch network: (a) no contention occurs at intermediate nodes, (b) contention occurs at node A, and (c) contention occurs at node C

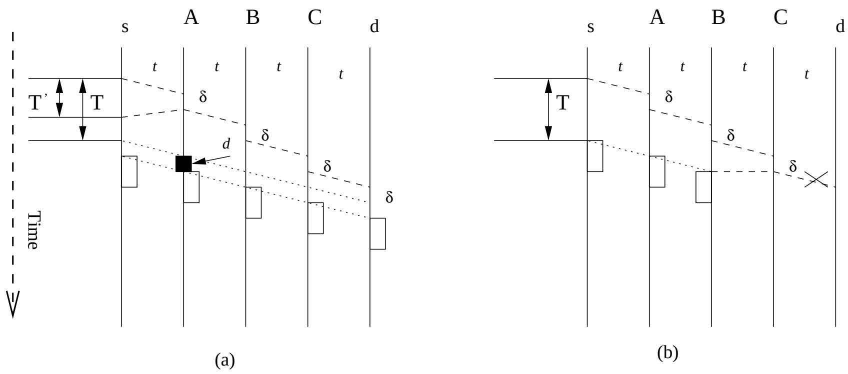

Fig. 3. Timing diagram in OBS-$Flex$: (a) contention at node A is resolved, and (b) contention at node C though the burst is dropped at node B.

sent from the intermediate node where contention has taken place either to delay the transmission at source or to release the reserved resource. Processing of F and R control packets is explained in the following paragraphs.

Working of OBS-$Flex$ is explained with the help of the timing diagrams illustrated in Figs. 2 and 3. The total delay encountered by a control packet for the source-destination pair (s, d) is no greater than $\Delta = \delta \times H^{sd}(z)$. The offset-time, T, in OBS is taken to be at least Δ. In Fig. 2, the number of hops between source-destination pair (s, d) is 4. Therefore, the offset-time T in OBS is 4δ. In OBS, if a contention occurs say at node A or at node C then the burst is dropped at A or at B as shown in Figs. 2(b) and 2(c), respectively. With this offset time a contending burst cannot be further delayed.

In OBS-$Flex$, the offset, T, between the source-destination pair (s, d) is taken to be $(t+\delta)H^{sd}(z)$. In the above example, the offset time between the source-destination pair (s, d) is $4(t + \delta)$. Let us consider Fig. 2(b) where contention has occurred at node A, and d is the duration of the contention period. F-control packet has taken *one* hop to reach the node A from source s, and R- control packet is sent from node A to source s to delay the transmission of burst for the contention period d. The packet will reach s at $T^{'} = 2(t + \delta)$ units after the source s has sent the F-control packet (Fig. 3(a)). The offset-time $T > T^{'}$ i.e., source s will receive the R-control packet to delay the transmission before expiry of the offset-time. Hence, the transmission of the burst is delayed and the burst is not dropped at node A; this is illustrated in Fig. 3(a).

Next, we consider Fig. 2(c) where contention has occurred at node C, and d is the duration of the contention period. R-control packet is sent from node C to source s to delay transmission of the burst for the contention period d; it will reach s at $T^{'} = 6(t + \delta)$. The offset-time $T < T^{'}$, i.e., source s will receive R-control packet after it has transmitted the burst and the burst is dropped at node C. Therefore, in the present case, R-control packet is sent to release resources reserved at the intermediate node rather than delaying at the source. Hence, R-control packet will release the resources reserved at node B before the burst arrives at node B and is dropped at node B rather than at node C. This gives rise to better utilization of the resources on link BC which was earlier occupied by the request.

The OBS-$Flex$ differs from other OBS schemes in two aspects - one, the offset time, and second, the methods adopted for contention resolution. In other OBS schemes, the offset time is $\delta \cdot H$ where δ is the processing delay of control packet at each node, and H is the number of hops between source-destination pair. In OBS-$Flex$, we take the offset time to be $P + \delta \cdot H$ where P is the additional propagation delay between source-destination pair. The minimum latency of burst in other OBS schemes, is $P + \delta \cdot H$ which is the same if a burst is sent along with control packet in optical packet switching. The minimum latency in optical circuit switching is $3P + \delta \cdot H$. In OBS-$Flex$, the minimum latency of a burst is $2P + \delta \cdot H$. In other OBS schemes, where $two\ way$ reservation protocol is used the minimum latency is $2P + \delta \cdot H$. Thus, we can say that the minimum latency of OBS-$Flex$ is identical to the OBS scheme with $two\ way$ reservation protocol. However, OBS-$Flex$ is a $one\ way$ reservation protocol where each burst experiences an additional delay of P units. OBS-$Flex$ is also tunable to delay sensitive traffic. For delay sensitive traffic, the offset time in OBS-$Flex$ is taken to be $\delta \cdot H$ which is the same as that in OBS. However, this offset can be made adaptive to the needs of the applications. In OBS-$Flex$, if a contention occurs and the situation is a *CR* one (as mentioned in Section 2.1) then a burst is further delayed for the contention period. However, this delaying technique of OBS-$Flex$ is not applicable in case of delay sensitive traffic. For delay sensitive traffic if the required resource is not available within that amount of time, the burst is dropped.

Secondly, OBS-$Flex$ differs from other OBS schemes in the method adopted for contention resolution. In other OBS schemes, the resource conflict is resolved on the basis of the request priority and the time instance for which the request is made. In addition to the above two parameters, we take burst size and the number of hops traversed to resolve contention. A high priority request is given a priority. However, for the same priority requests, the one that has traversed the maximum number of hops, is accepted. For same priority and the number of hops traversed the one that has larger burst size is accepted. For all the three parameters having identical values, the instance of reservation is taken for conflict resolution. Therefore, ties in

f − path	r − path	th	T	w	s	d	rid	m	H

Fig. 4. Fields of F-control packet

f − path	th	T	w	s	d	rid	r	H

Fig. 5. Fields of R-control packet

contention resolution are resolved in the following order: priority, number of hops traversed, burst size and the delay.

2.3 Signaling Protocol and Implementation

We use two types of *forward* (F) and *reverse* (R) control packets in OBS-$Flex$. In the following subsections, we describe F and R control packets and OBS-$Flex$ signaling protocol.

2.3.1 Control Packets

F**-control packet**: An ingress router sends out F-control packet when a burst arrives requesting for reservation of resources at the intermediate core router. Resources are reserved using the delayed reservation technique, analogous to the one discussed in [20]. We sketch the structure of F-control packet in Fig. 4, and explain the fields of F-control packet below:

- f-path is an explicit forward path that F-control packet takes from the ingress to the egress router. Burst follows this path once transmitted.
- r-path is a reverse path of the forward f-path. For example, if f-path is $a \to b \to c \to d$, then r-path is $d \to c \to b \to a$.
- th is the number of hops F-control packet has traversed/completed. When a router receives F-control packet, it updates value of th to $th + 1$; initial value of th is set to *zero*.
- w is the wavelength requested for reservation by the ingress router.
- s is the source/ingress router.
- d is the destination/egress router.
- Value of T indicates the duration of the contention period. Initially the value of T is set to *zero* by the ingress router. When a contention occurs the value of T is set to the duration of the contention period.
- Value of m equals to *one* indicates that F-control packet is modified (initially, value of m is set to $zero$ by the ingress router). An intermediate node modifies F-control packet by setting the value of m to *one*. When value of m in F-control packet is set to *one* the resource reservation is deferred for the contention period

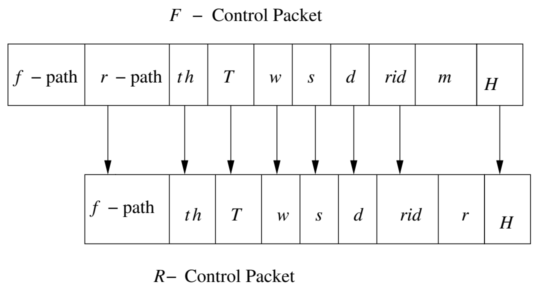

Fig. 6. Formation of R-control packet from F-control packet.

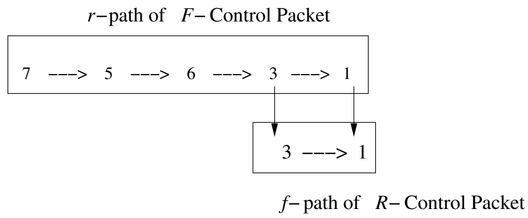

Fig. 7. Copying of f-path to r-path.

mentioned in the T field.

- rid is the request identity.
- Value of r indicates whether the resources are to be rescheduled or released.
- H is the total number of hops the request rid has to traverse.

When an intermediate core router receives F-control packet, one of the following three possible situations arises : (i) NC, (ii) CR, or (iii) CNR as described in Section 2.1. The action taken by the core router depends on the value of m in F-control packet and one of the above three situations. The intermediate core router updates the value of th in F-control packet to $th+1$. The actions taken by the core router for both the values of m and for all the three possible situations are discussed below. In the following paragraphs we list all the possibilities depending on the values of different fields in the control packets.

Case I: *When the value of m in F-control packet is equal to zero and one of the following situations occurs*:

1. NC : Required resources are reserved at the core router and F-control packet is forwarded to the next node in the path.
2. CR : The following actions are taken at the core router for the contending request: (i) the contention period is determined and the reservation is deferred for that period, (ii) the value of T in F-control packet is set to the above found contention period, (iii) the value of m in F-control packet is set to one, (iv) R-control packet is formed (formation of R-control packet is explained in the following paragraphs) and is sent to the ingress router s, and (v) F-control packet is sent to the next node in the path.
3. CNR: The following actions are taken at the core router for the contending request: (i) R-control packet is formed and is sent toward the source s to release

the resources reserved for the contending request, and (ii) F-control packet is dropped.

Case II: *When the value of m in F-control packet is equal to one, and one of the following situations occur*:

1. NC : Defer the reservation request for a period as mentioned in field T. For example, suppose resources are to be reserved at time x and the value set in field T is $x^{'}$. Then the resources will be reserved at time $x + x^{'}$.
2. CR : The following actions are taken at the core router for the contending request: (i) R-control packet is formed and sent toward source s to release the resources reserved for the contending request, and (ii) F-control packet is dropped. In our contention resolution scheme, we reschedule a request only once. If the required resources are not available to the already rescheduled request in any of the subsequent hop then that request is dropped.
3. CNR : The following actions are taken at the core router for the contending request: (i) R-control packet is formed and is sent toward the source s to release the resources reserved for the contending request, and (ii) F-control packet is dropped.

R**-control packet** : R-control packet is formed at the intermediate core router where the resource conflict has occurred. The structure of R-control packet is shown in Fig. 5. Fields of R-control packet are explained below:

f-path is an explicit path that R-control packet takes from the core router to the ingress router s. Semantics of the th, T, w, s, d and rid fields of R-control packet are identical to that of F-control packet. Value of r equal to *zero* indicates resources reserved are to be rescheduled to a later time as specified in field T. For example, if the resources is reserved at a node from time x and value of T field is $x^{'}$ then reservation at the node is rescheduled to a time $x+x^{'}$. A value equal to *one* indicates that the resources are to be released. R-control packet is formed from F-control packet and the formation is explained in the following paragraphs.

r-path of F-control packet is copied into f-path of R-control packet and all the other fields of F-control packet are copied to the corresponding fields of R-control packet (Fig. 6). Value of r is set to $zero$ if resources are to be rescheduled, otherwise set to one if resources are to be released. Copying r-path of F-control packet into f-path of R-control packet is illustrated in Fig. 7. In this illustration, we have assumed that resource conflict has occurred at core router 6. Remaining elements of r-path of F-control packet excluding node 6 are copied into f-path of R-control packet. R-control packet follows this f-path to reach the ingress router 1 for whose reservation request, the resource contention has occurred.

Processing of R-control packet : On receiving R-control packet, a node updates value of th in the control packet to $th + 1$. If value of $H - th \neq 0$ and value of r is *zero* then the resource reservation for request rid from ingress router s to egress router d is rescheduled for the time as specified in T field else resources are released. If the node is ingress router s, R-control packet is dropped after processing. If value of $H - th \neq 0$ then R-control packet is forwarded to the next node in f-path else R-control packet is dropped at that node. If a node on receiving R-control packet finds that the requested resources are subsequently reserved by another request than it does the following. If priority of the request for which R-control packet is generated is higher than the request that has subsequently reserved the resources then it de-reserves the request and re-schedules the request corresponding to R-control packet else R-control packet is dropped.

2.3.2 OBS-$Flex$ Signaling Protocol

The signaling protocol specifies the actions taken by both ingress and core routers.

The following actions are taken at the *ingress* router:

1) F-control packet is sent out when a burst arrives,
2) Burst is transmitted at the time for which resources are reserved, and
3) On receiving R-control packet depending on the value of r-field of R-control packet resources are either released or reservation is rescheduled to a time as specified in the control packet.

The actions taken at the *core* router are :

1) On receiving F-control packet it is processed as explained in previous subsection, and
2) On receiving R-control packet it is processed as explained in previous subsection.

Summarizing, actions that are needed to transmit a burst are: (i) send F-control packet, (ii) process F-control packet, (iii) process R-control packet, if any, and (iv) transmit a burst during the reserved time.

2.4 Correctness of OBS-$Flex$

In this subsection, we show with an illustration that OBS-$Flex$ operates as desired after rescheduling of reservation requests. We consider Fig. 8 for illustration. In Fig. 8(a), burst b_1 has reserved resources for duration t_1 to t_2 at node i and for duration t_5 to t_6 at node j. Burst b_2 has reserved resources for duration t_3 to t_4 at node i and for duration t_7 to t_8 at node j . In the above scenario there is no contention among bursts for resources. So both the bursts are transmitted successfully.

Let us assume that resource contention has occurred at node j for burst b_1 and situation is CR as mentioned in Section 2.1. Let σ be the duration of the contention period. In OBS-$Flex$, R-control packet is sent from the contention node, in this case node j, to the source to delay transmission of burst b_1 for contention period σ. On receiving R-control packet, node i will reschedule the transmission of burst b_1. Reschedule of burst b_1 at node i may overlap with burst b_2 which is already scheduled for transmission at node i, depending on value of α as shown in Fig. 8(a) and duration of contention period σ. There are now *two* possible cases: (i) $\sigma < \alpha$, and (ii) $\sigma > \alpha$. For $\sigma < \alpha$ reschedule of burst b_1 by node i is shown in Fig. 8(b). It can be seen from Fig. 8(b) that burst b_1 after reschedule does not overlap with burst b_2. Only release of resources by burst b_1 and acquire of resources by burst b_2 is decreased from α units in Fig. 8(a) to $\alpha - \sigma$ units in Fig. 8(b). Transmission of burst b_1 will be delayed for the contention period. Thus, both the bursts are transmitted successfully.

For $\sigma > \alpha$, reschedule of burst b_1 will overlap with burst b_2 as shown in Fig. 8(c). However, when burst b_2 arrives at node j it will contend with burst b_1 for the period $\sigma - \alpha$ as shown in Fig. 8(c). Therefore, node j will not schedule transmission of burst b_2 at t_7 but to a later time at $t_7 + \sigma - \alpha$. Node j also sends R-control packet to the source to delay transmission of burst b_2 for contention period $\sigma - \alpha$. Thus, node i reschedules the bursts as shown in Fig. 8(d). As seen from Fig. 8(d), bursts b_1 and b_2 do not overlap after their reschedule. Thus, they are transmitted successfully.

Therefore, it is illustrated with the help of a diagram that OBS-$Flex$ operates correctly after rescheduling of the reservation request.

3 Channel Selection Algorithms

In this section, we describe three channel selection algorithms, namely, (i) *Least Recently Used* (LRU), (ii) *First Fit* (FF), and (iii) *Priority Set* (PS) algorithms used for the proposed contention resolution scheme (Section 2.2) for channel selection. The channel selection algorithms are run only at the edge routers to find the data channel for which reservation request is to be made and transmit subsequently data burst. In LRU, a data channel which is idle for the maximum duration is selected. In FF, data channels are searched from the lowest index and the one which is available first, is selected. Consider Fig. 9, LRU channel selection algorithm selects data channel 2 as it is idle for the maximum duration where as FF channel selection algorithm selects data channel 0.

In PS approach, we decompose the set of data channels, S, into P subsets, S_i, of data channels where P is the number of priority classes supported. $S = S_0 \cup S_1 \cup \cdots \cup S_{P-1}$. A priority class i selects data channel from set S_i. If no data channel is available in set S_i then it selects from set S_{i-1} and if not available then from set

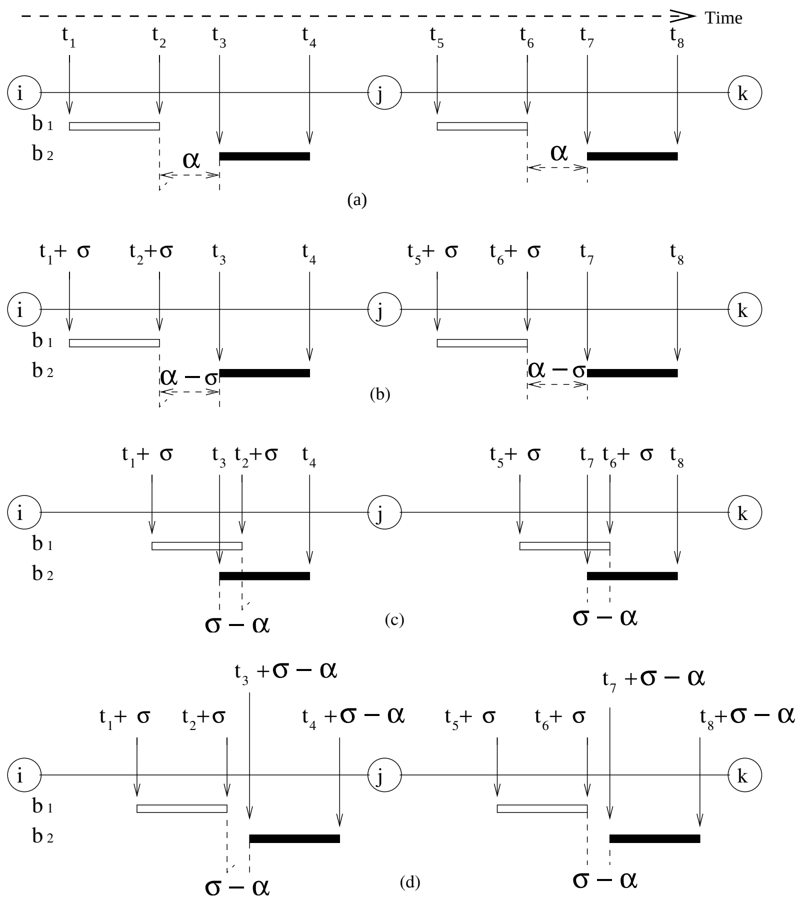

Fig. 8. Illustration of the working of OBS-$Flex$: (a) no contention, (b) no overlap due to rescheduling of burst b_1 [$\sigma < \alpha$], (c) overlap due to rescheduling of burst b_1 [$\sigma > \alpha$], and (d) no overlap due to rescheduling of burst b_1 and b_2 [$\sigma > \alpha$].

S_{i-2}. This process is iterated till the lowest priority set S_0 is searched. If no data channel is available in set S_0 then the burst is dropped at the ingress router. For priority class 0, if no data channel is available in set S_0 then the burst is dropped at the ingress router. To illustrate working of PS approach, we consider two priority classes 0 and 1; class 1 has higher priority than class 0. We divide available data channels as shown in Fig. 10 in two sets $S_0 = \{0, 1\}$ and $S_1 = \{2, 3\}$. Let class 1 burst arrives at t_a and it is to be transmitted at t_s after base offset time t_{offset}. Since all data channels in set S_1 are busy at t_s, channel 0 from set S_0 is selected.

Inputs to the above channel selection algorithms are burst arrival time, t_a, and offset time t_{offset}. PS algorithm has an additional input of burst priority. Output of each of the algorithms is the selected data-channel dc. A negative value of the output indicates that no data channel is available. The function *Channel_Available_Time* in each of the algorithms returns the available time of each data channel. Pseudocodes of LRU, FF and PS algorithms are included in Algo-1, Algo-2 and Algo-3, respectively.

Input: t_a, t_{offset}

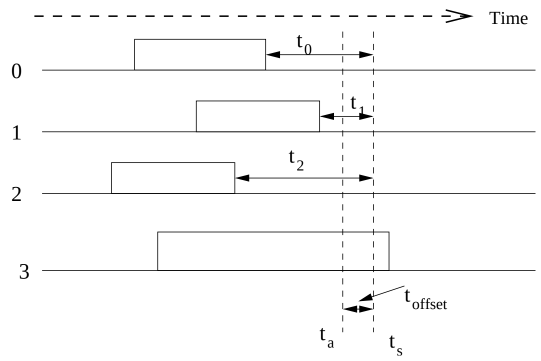

Fig. 9. Illustration for selection of data channel in LRU and FF algorithms.

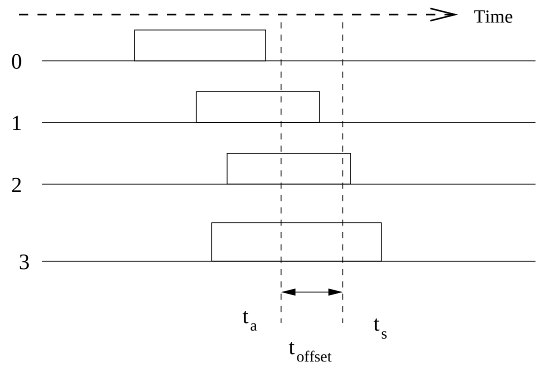

Fig. 10. Illustration for selection of data channel in PS algorithm.

Output: dc
Algorithm:
for $i \leftarrow 0$ to $Num\_DataChannel$ do
$av[i] \leftarrow Channel\_Available\_Time()$
end of for loop.
$dc \leftarrow Find\_LRUChannel(av, t_a, t_{offset})$
if dc is negative *then* drop the burst at the ingress router and report no data channel is available *else* report data channel dc.

Algo-1: Least Recently Used Channel Selection Algorithm

Input: t_a, t_{offset}
Output: dc
Algorithm:
for $i \leftarrow 0$ to $Num\_DataChannel$ do
$av[i] \leftarrow Channel\_Available\_Time()$
end of for loop.
$dc \leftarrow Find\_FFChannel(av, t_a, t_{offset})$
if dc is negative *then* drop the burst at the ingress router and report no data channel is available *else* report data channel dc.

Algo-2: First Fit Channel Selection Algorithm

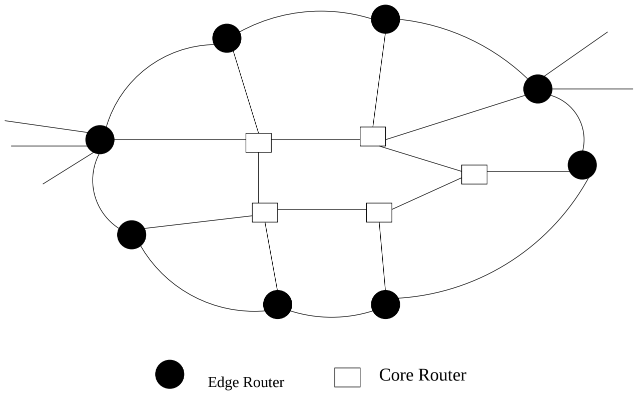

Fig. 11. Simulated Burst Switched Network

Input: t_a, t_{offset}, $priority$
Output: dc
Algorithm:
for $i \leftarrow$ 0 to $Num\_DataChannel$ do
$av[i] \leftarrow Channel\_Available\_Time()$
end of for loop.
$dc \leftarrow Find\_PSChannel(av, t_a, t_{offset}, priority)$
if dc is negative *then* drop the burst at the ingress router and report no data channel is available *else* report data channel dc.

Algo-3: Priority Set Channel Selection Algorithm

4 Simulation Results

We simulated burst switching network as shown in Fig. 11; dark circles indicate edge routers (ingress and egress router) and squares indicate core routers. We made the following assumptions in the simulation. The propagation delay, t, between any two adjacent nodes in the burst switching network is assumed to be $1ms$. This assumption is carried to simplify the simulation task; main aim of simulation is to demonstrate effectiveness of OBS-$Flex$ strategy in reducing burst losses, therefore, this is a reasonable assumption. Processing time of the control packet at the router is assumed to be $2\mu s$. We assume that there is no wavelength conversion and there exists no optical buffer in the switch. For simplicity and without loss of generality, we consider two classes of traffic: class 0 (low priority) and class 1 (high priority). We generate high priority traffic with a probability of 0.4. Traffic is generated at the edge router only, and the load is measured in Erlang. Performance of a network is strongly influenced by the statistics/patterns of the arriving traffic. To study the effect of traffic on the network performance, we consider the following two cases:

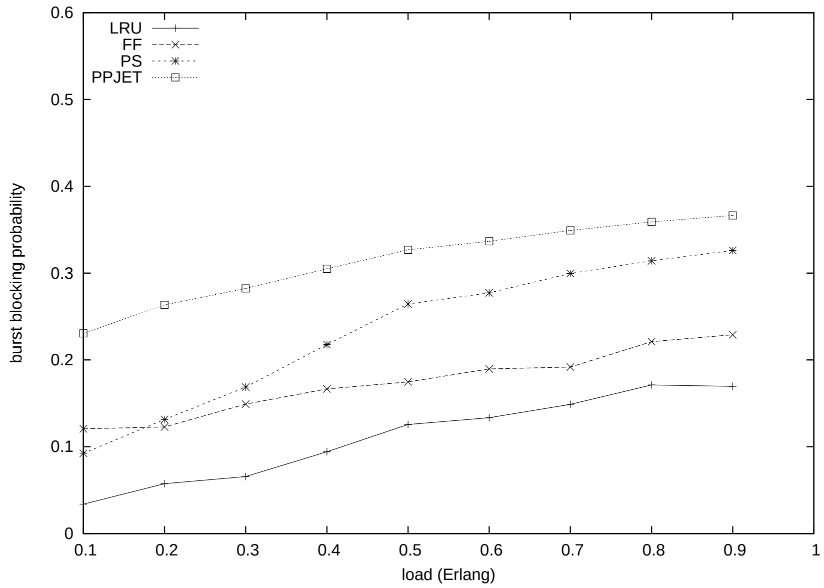

Fig. 12. Overall burst loss in OBS-$Flex$ with different channel selection algorithms and PPJET. Pareto distributed burst size and Pareto distributed inter arrival of burst is considered.

Case-I: Bursty Traffic: Pareto ($\alpha = 1.1$) burst length distribution and Pareto ($\alpha = 1.1$) inter arrival time distribution, and
Case-II: Poisson traffic: We consider two sub-cases – (a) Poisson distributed burst size and exponential inter arrival time distribution, and (b) Fixed burst size and exponential inter arrival time distribution.

We compare the simulation results obtained with the proposed scheme to that of PPJET [7]. We consider burst blocking probability as the performance metric for comparison. We have taken *seven* number of wavelengths available on each link.

4.1 Bursty Traffic

Traffic in the Internet is reported to be bursty in nature [11]. We consider Pareto ($\alpha = 1.1$) distributed burst length and Pareto ($\alpha = 1.1$) distributed inter arrival time. We include plots for burst loss under three situations, namely, overall, high priority and low priority for OBS-$Flex$ with the three proposed channel selection algorithms, in Figs. 12, 13 and 14, respectively. We also include burst loss obtained from PPJET in each of the graph for comparison.

The overall burst loss increases with increase in load as shown in Fig. 12. It is observed from the figure that the overall burst loss in OBS-$Flex$ is lower than that in PPJET. Of the three proposed channel selection algorithms, LRU gives compara-

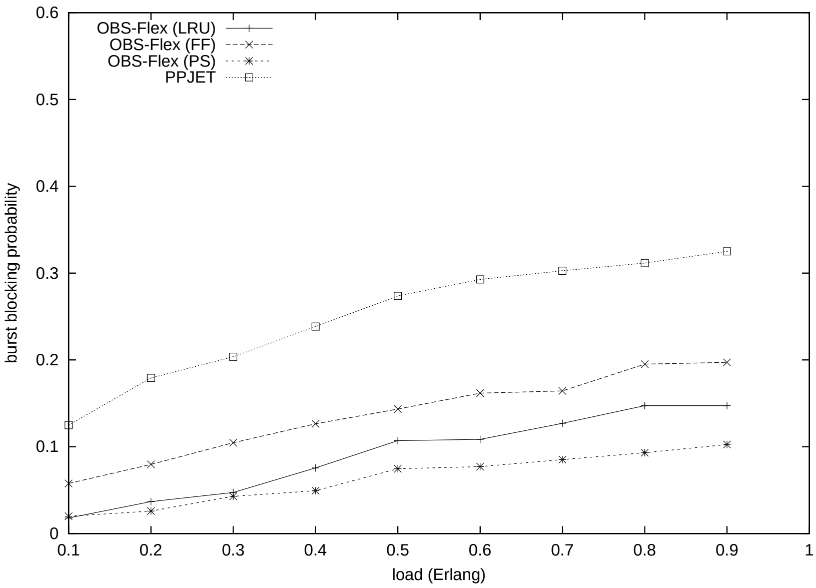

Fig. 13. High priority burst loss in OBS-$Flex$ with different channel selection algorithms and PPJET. Pareto distributed burst size and Pareto distributed inter arrival of burst is considered.

tively lower and PS gives higher overall burst loss. The higher overall burst loss in PS is due to the higher low priority burst loss shown in Fig. 14.

The plot for higher priority burst loss with load is included in Fig. 13. It is observed from the figure that OBS-$Flex$ has lower higher-priority burst loss than in PPJET. Of the three proposed channel selection algorithms, PS algorithm has lower and FF algorithm has higher high-priority burst loss. Low high-priority burst loss in PS is due to the channel selection strategy that is adopted in PS. In PS algorithm, a high priority traffic can select a channel that is marked for low priority traffic.

Burst loss for low priority traffic is plotted in Fig. 14. It is observed from the figure that OBS-$Flex$ has lower burst loss for LRU and FF algorithms than in PPJET for all loads. However, OBS-$Flex$ with PS algorithm experiences higher low-priority burst loss than in PPJET at higher load. The increase in the burst loss at higher load is attributed to the data channel consumed up by the higher priority traffic from those marked for low priority traffic.

From Figs. 12, 13 and 14, we can conclude that OBS-$Flex$ gives lower burst loss than that in PPJET for bursty traffic. For bursty traffic, if a low overall burst loss is desired then OBS-$Flex$ with LRU algorithm can be used. If low burst loss of high priority traffic is desired then OBS-$Flex$ with PS algorithm is the obvious choice.

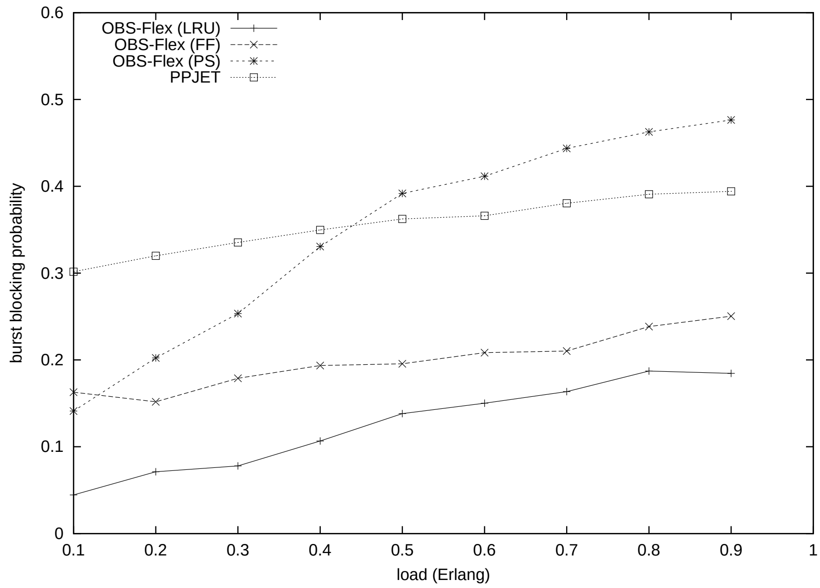

Fig. 14. Low priority burst loss in OBS-$Flex$ with different channel selection algorithms and PPJET. Pareto distributed burst size and Pareto distributed inter arrival of burst is considered.

4.2 Poisson Traffic

Next, we consider Poisson distributed burst size and exponential inter arrival of bursts. The mean exponential inter arrival of burst is assumed to be $1ms$. The plots for burst loss under three situations – overall, high priority and low – for Poisson distributed burst size and exponential inter arrival time are included in Figs. 15, 16 and 17, respectively.

Overall burst loss is plotted in Fig. 15. It is observed from the figure that OBS-$Flex$ with LRU algorithm gives lower burst loss, for all loads, than in PPJET. OBS-$Flex$ with FF and PS algorithms gives higher burst loss in PPJET for all load. Of the three channel selection algorithms, LRU gives the lowest burst loss for all load.

The higher priority burst loss is plotted in Fig. 16. From the figure, it is observed that OBS-$Flex$ with PS algorithm has lower burst loss than in PPJET for all load. OBS-$Flex$ with LRU algorithm has almost the same burst loss as that of PPJET for all load. Of the three channel selection algorithm, PS algorithm gives lower burst loss than with LRU and FF algorithms. The lower burst loss in PS can be attributed to the selection of data channels which were marked for low priority traffic.

Burst blocking probability for lower priority traffic is plotted in Fig. 17. From the figure, it is observed that OBS-$Flex$ with LRU algorithm gives the lower blocking than in PPJET for all load. OBS-$Flex$ with FF algorithm and PPJET scheme have almost the same blocking for all load. Of the three channel selection algorithms, PS has the higher blocking probability; this is due to the selection of data channel

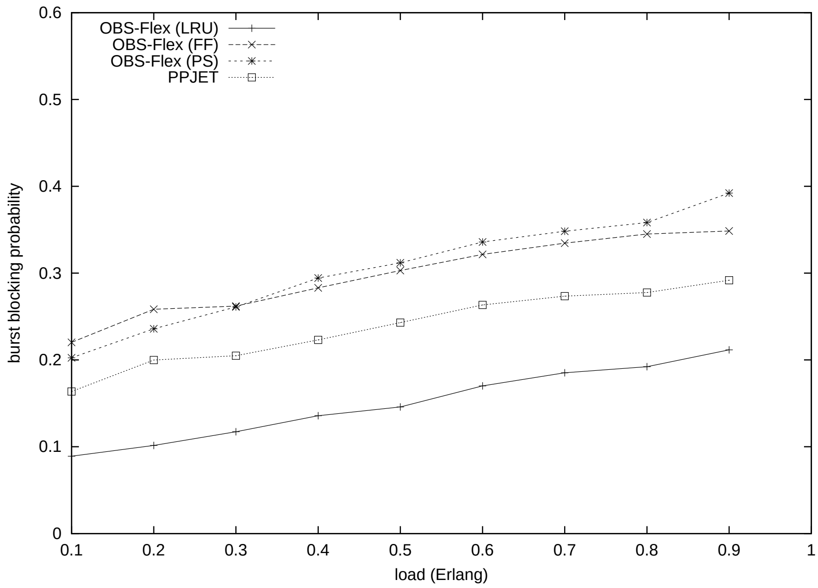

Fig. 15. Overall burst loss in OBS-$Flex$ with different channel selection algorithms and PPJET. Poisson distributed burst size and exponential inter arrival of burst is considered.

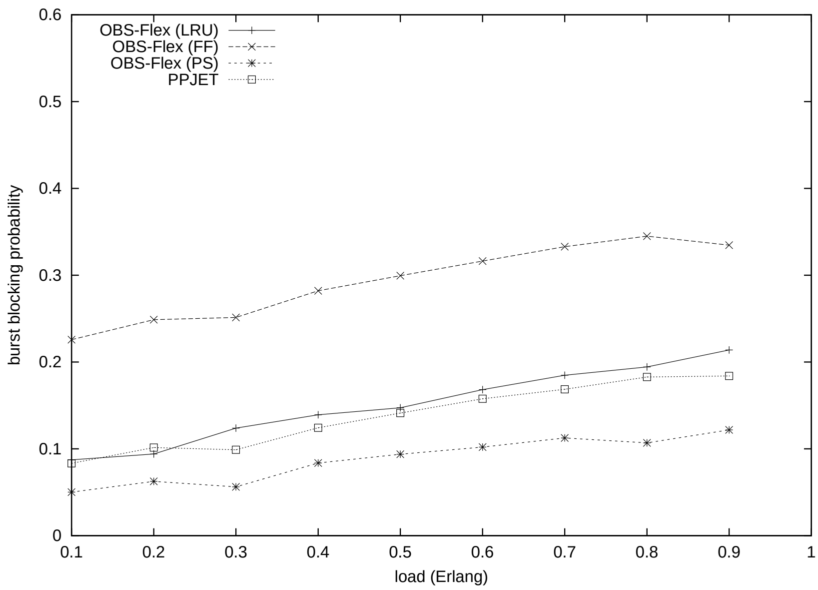

Fig. 16. High priority burst loss in OBS-$Flex$ with different channel selection algorithms and PPJET. Poisson distributed burst size and exponential inter arrival of burst is considered.

by higher priority traffic from the data channel marked for lower priority traffic.

From Figs. 15, 16 and 17, we can conclude that for traffic with Poisson distributed burst size and exponential inter arrival of bursts, OBS-$Flex$ gives lower burst loss than that in PPJET. If low overall burst loss is desired then OBS-$Flex$ with LRU algorithm may be the choice. If low high priority burst loss is desired then OBS-$Flex$ with PS algorithm will be the choice.

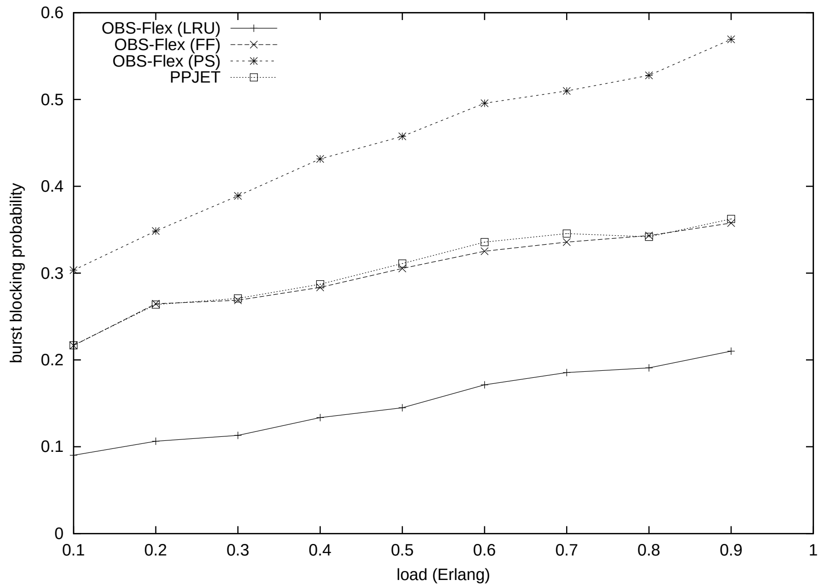

Fig. 17. Low priority burst loss in OBS-$Flex$ with different channel selection algorithms and PPJET. Poisson distributed burst size and exponential inter arrival of burst is considered.

Next, we consider fixed size bursts and exponential burst inter arrival time of bursts. We keep size of the bursts to be $100\mu s$. Plots for overall blocking probability, blocking of high and low priority traffic are included in Figs. 18, 19 and 20, respectively.

Fig. 18 shows the overall blocking probability of OBS-$Flex$ and PPJET with load. It is observed from the figure that OBS-$Flex$ with LRU algorithm gives lower blocking than in PPJET for all load. OBS-$Flex$ with PS algorithm has lower blocking than in PPJET at lower load though at higher load the differences become marginal. Of the three channel selection algorithms, LRU algorithm gives lower blocking and FF gives higher blocking.

The blocking probability of high priority bursts is plotted in Fig. 19. From the figure it is observed that blocking probability in OBS-$Flex$ with PS and LRU algorithms is lower than in PPJET for all load. Of the three channel selection algorithms, PS algorithm has lower blocking probability; this is attributed to the selection of the channel marked for low priority traffic.

The blocking probability of low priority bursts is included in Fig. 20. From the figure, it is observed that the OBS-$Flex$ with LRU algorithm gives lower blocking probability than in PPJET for all load. It is also observed from the figure that LRU algorithm gives lower blocking probability than other proposed channel selection algorithms.

From Figs. 18, 19 and 20, we can conclude that for traffic with fixed burst size and exponential inter arrival of burst OBS-$Flex$ gives lower burst loss than that in PPJET. For low overall burst loss OBS-$Flex$ with LRU algorithm is the choice. If

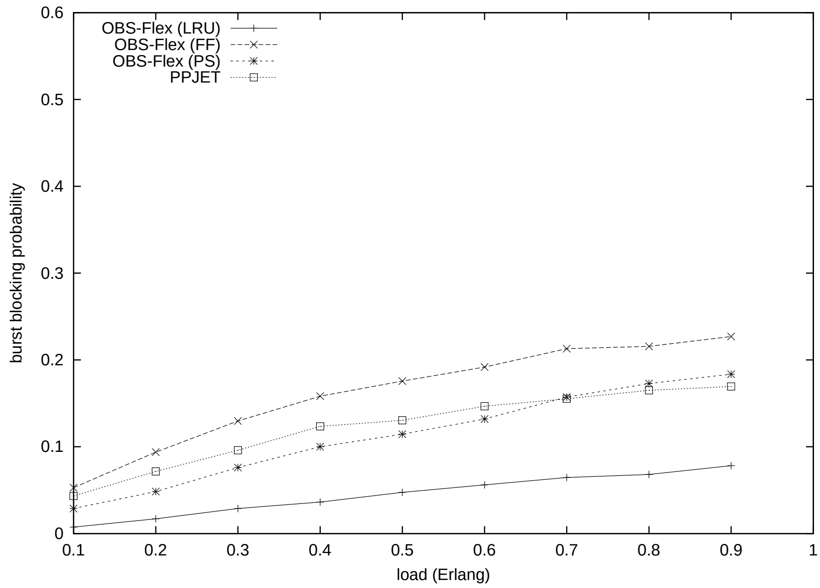

Fig. 18. Overall burst loss in OBS-$Flex$ with different channel selection algorithm and PPJET. Fixed burst size and exponential inter arrival of burst is considered.

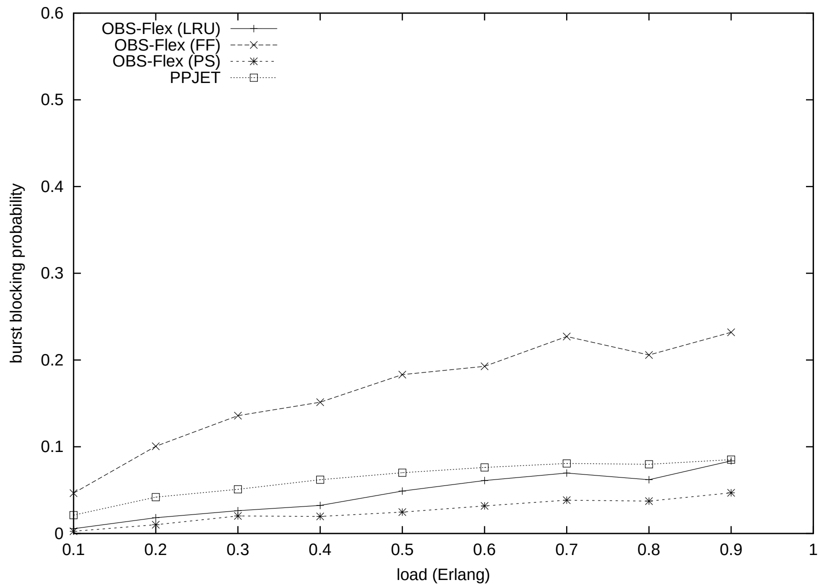

Fig. 19. High priority burst loss in OBS-$Flex$ with different channel selection algorithms and PPJET. Fixed burst size and exponential inter arrival of burst is considered.

lower high priority burst loss is desired, OBS-$Flex$ with PS algorithm may be the choice.

Comparing Figs. 15 and 18, it is observed for traffic with fixed burst size that the overall burst loss is lower than traffic with Poisson distributed burst size. Similarly, from Figs. 16 and 19, it is observed that the burst loss for high priority traffic for fixed size burst is lower than that in Poisson distributed burst size.

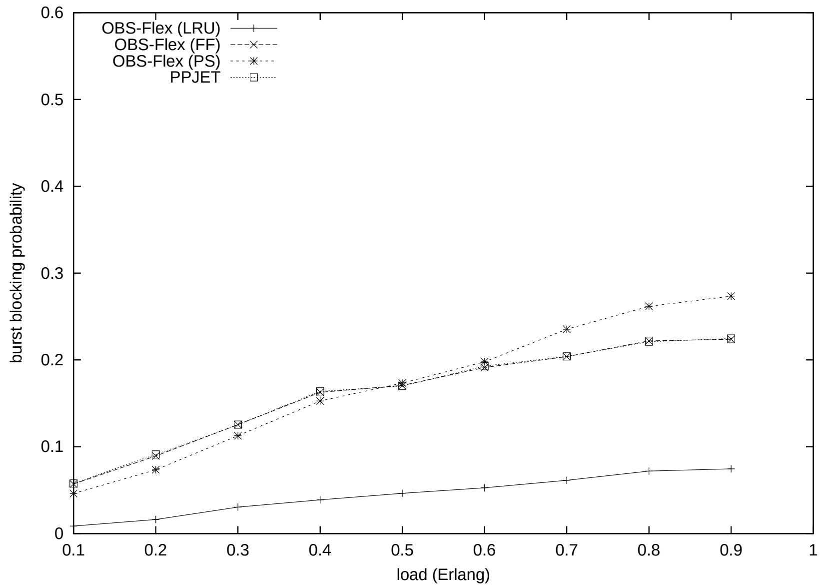

Fig. 20. Low priority burst loss in OBS-$Flex$ with different channel selection algorithms and PPJET. Fixed burst size and exponential inter arrival of burst is considered.

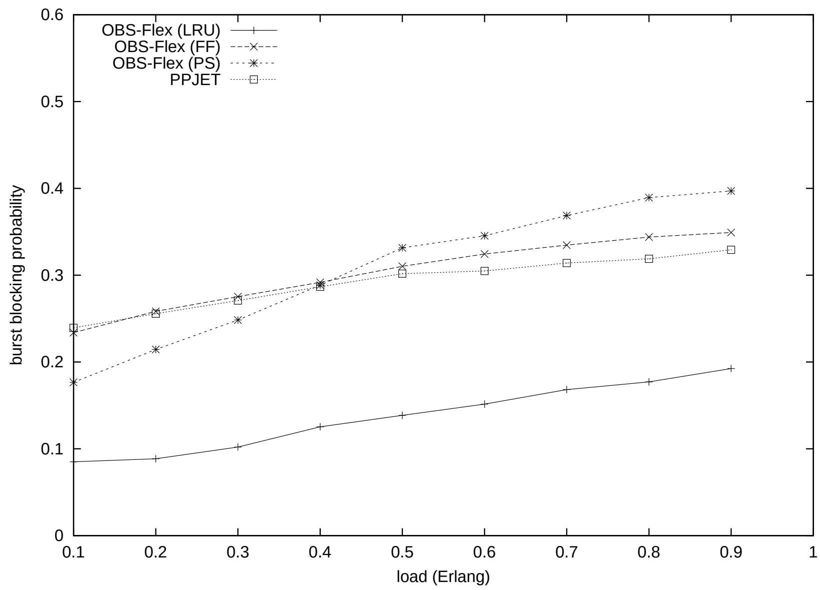

Fig. 21. Overall blocking probability considering delay constrained traffic in OBS-$Flex$ with different channel selection algorithms and PPJET. Pareto distributed burst size and Pareto distributed inter arrival of burst is considered.

4.3 Delay Constrained Traffic

In this section, we simulate to study the effect of contention resolution scheme for delay constrained traffic. In our simulation, we consider 20% of the total traffic as delay constrained traffic. The offset time for delay constrained traffic is taken to be $\delta \cdot H$ and for others $P + \delta \cdot H$, as mentioned in Section 2.2. We include plots for

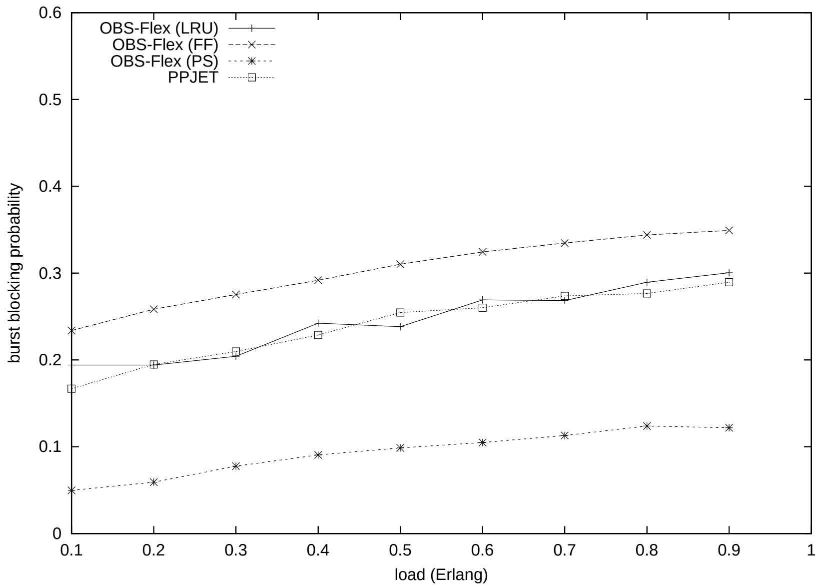

Fig. 22. Blocking probability of delay constrained traffic in OBS-$Flex$ with different channel selection algorithms and PPJET. Pareto distributed burst size and Pareto distributed inter arrival of burst is considered.

overall blocking probability, and blocking probability for delay constrained traffic in Figs. 21 and 22, respectively.

From the plots for overall blocking probability (Fig. 21), it is observed that OBS-$Flex$ with LRU algorithm gives lower overall blocking for all load. OBS-$Flex$ with FF algorithm and PPJET scheme have almost identical blocking probability at lower load. However, at higher load, blocking in OBS-$Flex$ with FF algorithm is marginally higher. OBS-$Flex$ with PS algorithm has lower overall blocking than in PPJET at lower load, however, at higher load blocking shows a marginally increasing trend.

The blocking probability for delay constrained traffic is shown in Fig. 22. It is observed that OBS-$Flex$ with PS algorithm has lower blocking than in PPJET for all load. Blocking in OBS-$Flex$ with LRU algorithm and PPJET are almost identical at all load. OBS-$Flex$ with FF algorithm has higher blocking than in PPJET at all load.

From the experiments, we can conclude that the proposed scheme OBS-$Flex$ gives lower burst loss than PPJET if delay constrained traffic is also taken into consideration. If we need a lower overall burst loss then we can use OBS-$Flex$ with LRU channel selection algorithm and if lower loss of delay sensitive burst is desired then OBS-$Flex$ with PS channel selection algorithm is a superior choice.

5 Conclusions

In this paper, we proposed a contention resolution scheme called OBS-$Flex$ for OBS networks. The scheme takes three parameters, namely, priority, number of hops traversed, and burst size into account to resolve contention. The proposed scheme is adaptable to both prioritized and delay constrained traffic. We also proposed three channel selection algorithms called, Least Recently Used (LRU), First Fit (FF), and Priority Set (PS) algorithms to select data channel at the ingress router. We simulate OBS-$Flex$ with each of the three channel selection algorithms. We considered both bursty and Poisson traffic in our simulations. Simulations were carried out for both prioritized traffic and delay constrained traffic. We observed that LRU channel selection algorithm gives lower overall burst loss for both prioritized and delay constrained traffic. In addition, PS channel selection algorithm gives the lowest burst loss for prioritized and delay constrained traffic.

We compared OBS-$Flex$ with another contention resolution scheme called PPJET. We found lower overall blocking probability in OBS-$Flex$ using LRU channel selection algorithm than in PPJET scheme for all load in both types of traffic that we have considered. OBS-$Flex$ using PS channel selection algorithm gives lower blocking for high priority traffic and delay constrained traffic than PPJET. Thus, we can conclude that irrespective of the type of traffic, if a lower overall burst loss is required than OBS-$Flex$ with LRU channel selection algorithm can be used. If a low blocking of high priority traffic or delay constrained traffic is desired then OBS-$Flex$ with PS channel selection algorithm may be the choice.

The lower blocking in OBS-$Flex$ comes with an additional delay for prioritized traffic. In PPJET, an incoming burst is delayed by an amount of time which is equal to the total processing time of the control token at each node. However, in OBS-$Flex$ an additional delay which is equal to the propagation time between source to destination, is involved for the prioritized traffic. Future work includes assessing the effect of increased delays, in order to minimize burst losses due to contention, on multimedia applications involving delay constrained traffic.

References

[1] *EURESCOM Project P918-GI, Deliverable 2.* http://www.eurescom.de.

[2] Noureddine Boudriga. Optical Burst Switching Protocol for Supporting QoS and Adaptive Routing. *Computer Communications*, 26:1804 – 1812, 2003.

[3] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath Communications: An Approach to High Bandwidth Optical WANs. *IEEE Transactions on Communications*, 40(7):1171 – 1182, July 1992.

[4] Michael Düser and Polina Bayvel. Analysis of a Dynamically Wavelength-

Routed Optical Burst Switched Network. *Journal of LightWave Technology*, 20(4):574 – 585, April 2002.

[5] R. Dutta and G.N Rouskas. A Survey of Virtual Topology Design Algorithm for Wavelength Routed Optical Networks. *Optical Network Magazine*, 1(1):73–89, January 2000.

[6] Amaury Jourdan, Domonique Chiaroni, Emmanuel Dotaro, Gert J. Eilenberger, Francesco Masetti, and Monique Renaud. The Perspective of Optical Packet Switching in IP-Dominant Backbone and Metropolitan Networks. *IEEE Communications Magazine*, 39(3):136 – 141, March 2001.

[7] Ayman Kaheel and Hussein Alnuweiri. A Strict Priority Scheme for Quality-of-Service Provisioning in Optical Burst Switching Networks. In *Proceedings of the Eight IEEE International Symposium on Computers and Communication (ISCC'03)*, pages 16 – 21, 2003.

[8] Hyun-Sook Kim, Su-Kyoung Lee, and Joo-Seok Song. Optical Burst Switching with Limited Deflection Routing Rules. *IEICE Trans. Commun.*, E86-B(5), May 2003.

[9] Su-Kyoung Lee, David Griffith, and Joo-Seok Song. Lambda GLSP setup with QoS requirement in optical Internet. *Computer Communications*, 26(6):603 – 610, 2003.

[10] Kejie Liu, Jason P. Jue, Gaoxi Xiao, Imrich Chlamtac, and Timucin Ozugur. Intermediate Node Initiated Reservation (IIR): A New Signaling Protocol for Wavelength-Routed Networks. *IEEE Journal on Selected Areas in Communications*, 21(8):1285 – 1294, October 2003.

[11] Vern Paxson and Sally Floyd. Wide Area Traffic: The Failure of Poisson Modeling. *IEEE/ACM Transaction on Networking*, 3(3):226 – 244, June 1995.

[12] Rajiv Ramaswami and Kumar N. Sivarajan. *Optical Networks: A Practical Perspective*. Morgan Kaufmann, 1998.

[13] E. Varvarigos and V. Sharma. A Ready-To-Go Virtual Circuit Protocol: A Loss-Free Protocol for Multigigabit Network Using FIFO Buffers. *IEEE/ACM Transactions on Networking*, 5(5):705 – 718, October 1997.

[14] Vinod M. Vokkarane, Qiong Zhang, Jason P. Jue, and Biao Chen. Generalized Burst Assembly and Scheduling Techniques for QoS Support in Optical Burst-Switched Networks. In *Global Telecommunications Conference, 2002, GlOBECOM'02.*, volume 3, pages 2747 – 2751, November 2002.

[15] J. Y. Wei, J. L. Pastor, R. S. Ramamurthy, and Y. Tsai. Just-In-Time Optical Burst Switching for Multi-Wavelength Networks. In *Proceedings of 5th International Conference on Broadband Communications (BC'99)*, pages 339 – 352, 1999.

[16] I. Widjaja. Performance Analysis of Burst Admission-Control Protocol. *IEE Proceedings Communications*, 142(1):7 – 14, February 1995.

[17] Shun Yao, S.J. Ben Yoo, Biswanath Mukherjee, and Sudhir Dixit. All-Optical Packet Switching for Metropolitan Area Networks: Opportunities and Challenges. *IEEE Communications Magazine*, 39(3):142 – 148, March 2001.

[18] M. Yoo, M. Jeong, and C. Qiao. A High Speed Protocol for Bursty Traffic in Optical Networks. In *SPIE Proceedings All Optical Communication Sys-*

tems: Architecture , Control and Network Issue, volume 3531, pages 79 – 80, November 1997.

[19] M. Yoo and C. Qiao. Just-enough-time(JET): A High Speed Protocol for for Bursty Traffic in Optical Networks. *IEEE/LEOS Technologies for a Global Information Infrastructure*, pages 26 – 27, August 1997.

[20] M. Yoo and C. Qiao. Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet. *Journal of High Speed Network*, 8(1):69 – 84, 1999.

[21] M. Yoo, C. Qiao, and Sudhir Dixit. QoS Performance in IP over WDM Networks. *IEEE Journal on Selected Areas in Communications, Special Issues on Protocols for Next Generation Optical Internet*, 18(10):2062 – 2071, October 2000.

[22] Qiong Zhang, Vinod M. Vokkarane, Biao Chen, and Jason P. Jue. Early Drop and Wavelength Grouping Schemes for Providing Absolute QoS Differentiation in Optical Burst-Switched Networks. In *Global Telecommunications Conference, 2003, GLOBECOM' 02, IEEE, Volume: 5, 1-5 December 2003*, pages 2694 – 2698, December 2003.

[23] Qiong Zhang, Vinod M. Vokkarane, Biao Chen, and Jason P. Jue. Early Drop Scheme for Providing Absolute QoS Differentiation in Optical Burst-Switched Networks. In *Workshop on High Performance Switching and Routing, 2003, HPSR*, pages 153 – 157, June 2003.