A Feedback-Based Contention Avoidance Mechanism for Optical Burst Switching Networks

Farid Farahmand[†], Qiong Zhang[‡], and Jason P. Jue[‡]
[†]Department of Electrical Engineering
[‡]Department of Computer Science
The University of Texas at Dallas, Richardson, TX 75083-0688
{ffarid, qzhang77, jjue}@utdallas.edu

Abstract

Optical burst switching (OBS) has been proposed as a promising technology to support the next generation optical Internet. In this paper we describe a feedback-based OBS network architecture in which core switch nodes send explicit messages to edge nodes requesting them to reduce their transmission rate on congested links. Within this framework, we introduce a new contention avoidance mechanism called source flow-rate control (SFC). Through admission control, the SFC proactively attempts to prevent the network from entering the congestion state. Basic building blocks, scheduling policy, and performance trade-offs of SFC are the main focus of this paper. In addition, we elaborate on architectural variations of our proposed contention avoidance mechanism and discuss the pros and cons for each case. Our simulation results show that the proposed contention avoidance techniques improve the network utilization and reduce the packet loss probability.

Keywords*: Admission control, contention resolution, feedback control, optical burst switching, traffic shaping.*

1. Introduction

The amount of raw bandwidth available on fiber optic links has increased dramatically with advances in dense wavelength division multiplexing (DWDM) technology; however, existing optical network architectures are unable to fully utilize this bandwidth to support highly dynamic and bursty traffic. Optical burst switching (OBS) [1]-[2] has been proposed as a new paradigm to provide the flexible and dynamic bandwidth allocation required to support such traffic. In OBS networks, incoming data is assembled into basic units, referred to as data bursts (DB), which are then transported over the optical core network. Control signaling is performed out-of-band by control packets (CP) which carry information such as the length, the destination address, and the QoS requirements of the optical data burst. The control packet is separated from the data burst by an offset time, allowing the control packet to be processed at each intermediate node before the data burst arrives. OBS provides dynamic bandwidth allocation and statistical multiplexing of data. By aggregating packets into large sized bursts and providing out-of-band signaling, OBS eliminates the complex implementation issues of optical packet switching. For example, no optical buffers are necessary at core nodes, headers can be processed at slower speeds in electronic domain, and synchronization requirements are relaxed in OBS. On the other hand, due to packet aggregation, OBS incurs higher end-to-end delay and higher packet loss per contention than optical packet switching.

Recently, considerable attention has been given to address and study various important issues in OBS networks. For example, many articles have focused on signaling and scheduling mechanisms for reserving and releasing resources in OBS. First-Fit, Horizon, Latest Available Unscheduled Channel (LAUC), and Latest Available Unscheduled Channel with Void Filling (LAUC-VF) are among the proposed scheduling algorithms [2],[4]. In both LAUC and LAUC-VF scheduling algorithms, a burst chooses the unused channel that becomes available at the latest time. When void filling (VF) is allowed, gaps between two scheduled data bursts can also be utilized. In these schemes the data burst reservation time starts at the beginning of the actual burst arrival and lasts until the end of the burst. Some studies have been dedicated to OBS architecture issues, including the signaling protocols and scheduler architecture [3], [4]. Others have proposed various ways to implement Multi-Protocol Label Switching (MPLS) and TCP/IP over OBS [7], [29].

A major concern in OBS networks is high contention and burst loss due to output data channel contention, which occurs when the total number of data bursts going to the same output port at a given time is larger than the available channels on that port. Contention is aggravated when the traffic becomes bursty and when the data burst duration varies and becomes longer. Contention and loss may be reduced by implementing *contention resolution policies*, such as time deflection (using buffering [5],[8]), space deflection (using deflection routing [9],[10],[11],[12]), and wavelength conversion (using wavelength converters) [13]. When there is no available unscheduled channel, and a contention cannot be resolved by any one of the above techniques, one or more bursts must be dropped. The policy for selecting which bursts to drop is referred to as the *soft contention resolution policy* and is used to reduce the overall burst loss rate, BLR, and consequently, to enhance link utilization. Several soft contention resolution algorithms have been proposed and studied in earlier literature, including the shortest-drop policy [4], segmentation [14], and look-ahead contention resolution [16].

The contention resolution policies are considered as *reactive* approaches in the sense that they are invoked after contention occurs. An alternative approach to reduce network contention is by *proactively* attempting to avoid network overload through traffic management policies. Consequently, *contention avoidance policies* attempt to prevent a network from entering the congestion state in which burst loss occurs. An ideal contention avoidance policy must serve several concurrent objectives: minimize the throughput, minimize the

average end-to-end packet delay, operate with minimum additional signaling requirements, and guarantee fairness among all users.

In general, contention avoidance policies can be implemented in either non-feedback-based or feedback-based networks. In a non-feedback-based network, the ingress nodes have no knowledge of the network state and they cannot respond to changes in the network load. Therefore, without requiring any additional signals in the control plane, each node regulates its own offered load into the network through traffic shaping (e.g. forcing the data bursts to enter the OBS network at a more regulated rate) or traffic rerouting and load balancing based on a predefined traffic description. One way to perform the traffic shaping is through a burst assembly mechanism such as the ones proposed in [18]-[21]. In [22], the authors propose regulating data bursts by combining periodic traffic reshaping at the edge node and a proactive reservation scheme. Traffic rerouting on alternative shortest paths (or load splitting) can also be implemented as a way to reduce link contention. The main challenge in implementing the contention avoidance policies in non-feedback-based OBS networks is to define the traffic parameter, such as peak rate and average rate at each edge node, in order to avoid or minimize link contention.

In a feedback-based network, contention avoidance is achieved by dynamically varying the data burst flows at the source to match the latest status of the network and its available resources. Thus, as the available network resources are changed, a source should vary its data burst transmission rate (or its offered load) to the network, accordingly. The main issues in feedback-based networks include defining the feedback mechanism and determining what type of information must be conveyed to the source [23]. Once the node receives the proper information, the main design issues include how to interpret the conveyed information and how to react to the current network state.

One way to avoid contention in feedback-based OBS networks is to *reroute* some of the traffic from heavily loaded paths to underutilized paths [24]. In this case, a core node sends feedback messages containing the load information of its overloaded output links to the ingress nodes. A similar approach has also been introduced by [25] where the authors consider balancing the data burst traffic between predefined alternative paths. Another way to avoid contention is to implement a TCP-like congestion avoidance mechanism to regulate the burst transmission rate [26]-[28]. In this approach, the ingress edge nodes receive TCP ACK packets from egress edge nodes, calculate the most congested links, and reroute their traffic accordingly. A potential drawback of these schemes is that rerouting the data bursts to alternative paths can potentially cause link congestion elsewhere and thus result in possible network instability. Furthermore, when the round trip delay is large and the network operates at a very high speed, the edge nodes' responses to the network change tend to be slow.

In this article we introduce a contention avoidance policy designed for feedback-based OBS networks where explicit feedback signaling is sent to each source indicating the required reduction in the burst flow rate going to congested links. Hence, the edge node attempts to avoid or minimize contention by adjusting its data burst flow rate to the required level through admission control. We refer to such feedback-based contention avoidance as source flow-rate control (SFC). We consider a label-switched OBS network and describe the architectural details of its feedback mechanism. Furthermore, we also describe different SFC-based architectures. By means of simulation, we examine the performance of our proposed feedback-based contention avoidance mechanism under specific network conditions. We compare our results with those without source traffic control in terms of blocking probability and network throughput. We show that our approach behaves well in practice and responds quickly to any change in network status, while improving the overall network performance.

The rest of this paper is organized as follows. In Section 2, we briefly describe the basic blocks and architecture of the label-switched feedback-based OBS network. In Section 3 we elaborate on details of our proposed contention avoidance policy. Finally, in Section 4 we present performance results obtained by means of simulations followed by concluding remarks in Section 5.

2. Feedback-Based Congestion Control Architecture

Typically, in TCP/IP or ATM networks, a traffic source must control its transmission rate in response to the receiver state as well as the network state [6]. However, in OBS networks, it is generally assumed that the ingress and egress nodes have adequate buffering capacity, and that matching the source rate to the service rate at the destination is not of great importance. Henceforth, the main objective in feedback-based contention avoidance schemes in OBS networks is to dynamically adjust (or regulate) the data burst transmission rate at edge nodes in response to core nodes' feedback signals, such that network overload is avoided or minimized. We refer to such closed loop traffic regulation as *admission control*. The schemes that determine the way the traffic is regulated are called *admission control strategies*.

2.1. Feedback components

Figure 1 identifies two key elements in feedback-based contention avoidance schemes in OBS networks: control and signaling strategies. The feedback control strategy refers to the type of action that the node receiving the feedback messages performs. For example, an edge node can reduce the transmission rate through admission control strategies or reroute data burst flows going through the congested link. On the other hand, the feedback strategy indicates how the current state of the network is measured and is communicated to other nodes (such as ingress or egress edge nodes or intermediate

core nodes). The feedback signaling strategy involves the following taxonomies:

(a) Feedback control type: refers to the type of the control messaging that is used to communicate the current state of the network to the source. The signaling type can be *explicit* or *implicit*. In the former, the feedback signal explicitly indicates the congestion state and the requested transmission rate (or transmission rate reduction). In the latter, the feedback signal indicates the rate of the packet loss on a particular link or in a node.

(b) Feedback triggering mechanism: indicates how often the feedback signaling is sent to upstream nodes. For example, the feedback signals can be transmitted periodically or based on some other node's request. Once the feedback signal is triggered it can be *broadcasted* to all sources or sent to particular nodes.

(c) Feedback point-of-control: refers to the nodes which respond to the feedback messages and take action to avoid congestion occurrence. The responding nodes can be the edge nodes or the adjacent core nodes. We refer to these as end-to-end and hop-by-hop signaling, respectively.

In this paper we only focus on a feedback-based contention avoidance mechanism in which each core node periodically broadcasts explicit link information to all edge nodes requesting them to dynamically adjust their data burst transmission rate if necessary. Thus, upon receiving the feedback information, edge nodes invoke their admission control and reduce the transmission rate of data burst flows passing through the congested link according to the requested rate. Note that all bursts belonging to the same burst flow share identical source and destination nodes. The admission control strategy we adopt in our study is a leaky bucket-based approach in which data bursts are scheduled on available wavelengths and transmitted according to a sustainable rate governed by feedback transmission rate reduction requests from intermediate nodes. We call this feedback-based traffic control mechanism the *source flow-rate control* (SFC) contention avoidance scheme. In this mechanism, the total volume of offered traffic is not changed; rather only the transmitting rate of the data burst flow is regulated through the admission control. The regulated traffic rate (bursts/sec) is directly related to the state of the congested link. Once a link is over-utilized, the reduction in the transmission rate continues until the core node clears out the congestion condition. At this point, the edge node attempts to resume its original transmission rate according to some ramp-up policy. In this paper, we limit our analysis to examining how the proposed mechanism can reduce burst loss probability and prevent throughput degradation in OBS networks. We do not consider issues such as fairness or the impact of control overhead signals.

2.2. Network model

Without loss generality, we consider the label-switched OBS networks using a Generalized Multi-Protocol Label Switching (GMPLS) control plane [8],[7],[29],[30]. In this model the transmitted bursts are routed through individual Label Switch Paths (LSPs). We assume that the intermediate core nodes have no buffering capacity, and that incoming LSPs can either cut through the core nodes or be blocked. When the measured load on an egress port exceeds a predefined load threshold, the congested core node sends back a *flow-rate reduction request* (FRR) signal to ingress edge nodes requesting them to reduce transmission rate of LSPs sharing the congested link. The feedback signaling to the source nodes can be implemented using the Label Distributed Protocol (LDP) employed in GMPLS. In this case, the feedback reduction request messages will be similar to the NACK message and include the following information:

`<LSP Label, Core Switch Address, FRR>.`

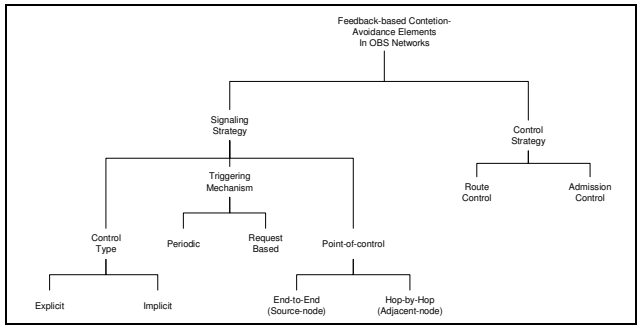

Figure 1. Control elements in a feedback-based contention avoidance scheme.

The FRR message consists of two fields: control field and rate reduction request value field. The control field is used to indicate the status of the FRR. For example, the FRR state can be set to *idle* or *no-change* state by enabling the i-bit or the nc-bit in the control field of the FRR message, respectively. The idle state indicates that links are congestion-free and sources can increase their transmission rate. The no-change state, on the other hand, is a state in which sources must not increase their transmission rate of data bursts passing through specified links. The value field, denoted by $R_{j,k}$, indicates the actual rate reduction value required by the switch on link *(j,k).*

The core switch address is provided in case the ingress edge node was allowed to use an alternative path for transmitting the affected LSP. The actual feedback messaging can also be implemented via Resource reSerVation Protocol (RSVP) [6]. In this case the FRR messages are encapsulated into the RESV messages propagating to upstream nodes. It must be noted that the feedback signaling can also be deployed independent of the RSVP or LSP control planes. In the rest of this paper we refer to an LSP and a burst flow interchangeably.

3. Source Flow-rate Control Congestion Avoidance

In the SFC contention avoidance mechanism, each core node maintains the load information on each of its egress link, *(j,k)*,

denoted by $\rho_{j,k}$. This is calculated by measuring the duration of all incoming data bursts destined to egress port *j*, over some fixed interval Δ. If the measured load for the egress port is greater than some predefined load threshold, ρ_{TH}, then the flow-rate reduction request (FRR) will be generated. The value of FRR *explicitly* indicates the percentage by which edge nodes must reduce the transmission rate of all burst flows (or LSPs) sharing link *(j,k)* in the immediate future, and it is equivalent to $R_{j,k} = (\rho_{j,k} - \rho_{TH})/\rho_{j,k}$, $R_{j,k} \in [0,1]$. In order to reduce the number of feedback signals between nodes, we can consider sending the FRR request only when the measured load on an egress port is changed or when it continues to be larger than ρ_{TH}. Otherwise, the FRR can be set to idle and transmitted once every several Δ time intervals. Sending an idle FRR is also a convenient way to ensure that the links are *alive*. For example, if the ingress edge nodes did not receive any FRR within a designated time interval, the edge node can assume that the link is disrupted and thus it must suspend transmitting the related burst flows.

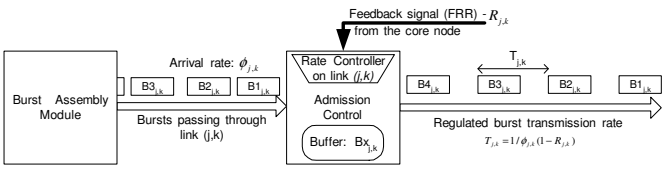

Figure 2. Basic idea of the source flow rate-based contention avoidance scheme implemented at the edge node.

The basic idea of the SFC contention avoidance scheme is shown in Figure 2. Once the FRR messages are received, edge nodes determine the most congested link *(j,k)* along a path (or multiple paths sharing the link) and subject all data bursts (or LSPs) passing through the congested link ($B1_{j,k}$, $B2_{j,k}$, etc.) to admission control. Meanwhile, new assembled data bursts sharing link *(j,k)* will be buffered until they their transmission time comes. In practice, each edge node maintains a matrix containing the average transmission rates at which it has been sending data bursts on individual WDM links *(j,k)*, $TRM[\phi_{j,k}]$, where $\phi_{j,k}$ is the transmission rate of data bursts on link *(j,k)*. Similarly, all the latest reduction requests are stored in matrix $RRM[R_{j,k}]$. By referring to RRM matrix, the edge node determines the most congested link along each path; note that in our network model we assume all bursts are source routed. Consequently, the edge node reduces the transmission rate on link *(j,k)* by $R_{j,k}$ in the next time interval Δ. As a result, the initial burst transmission rate on the congested link *(j,k)*, $\phi_{j,k}^{\Delta}$, is stored and it is reduced to $\phi_{j,k}^{\Delta+1} = \phi_{j,k}^{\Delta}(1-R_{j,k})$ in the next time interval. When the admission control is invoked, data bursts are transmitted at a sustainable rate equivalent to $\phi_{j,k}^{\Delta+1}$. Thus, the interarrival time will be $T^{\Delta+1} = 1/\phi_{j,k}^{\Delta+1}$. In other words, every time the source sends a burst on link *(j,k)* it sets a timer value with a timeout equal to the inverse of the required transmission rate, and it transmits the next burst traveling on the same link when the timer expires. Assuming, due to persistent network congestion, several FRRs arrived for the same link *(j,k)*, the edge node will have to continue reducing the sending rate of affected data bursts for some $x.\Delta$ time intervals until the FRR turns to idle or no-change state. Consequently, the data burst transmission rate on link *(j,k)* reduces to

$$\phi_{j,k}^{x.\Delta} = \phi_{j,k}^{\Delta} \prod_{i}^{x} (1 - R_{j,k}^{i\cdot\Delta}),$$

where $\phi_{j,k}^{\Delta}$ is the latest sending rate prior to admission control. Hence, the minimum data burst interarrival time after $x.\Delta$ will be increased to

$$T_{j,k}^{x.\Delta} = 1/(\phi_{j,k}^{\Delta} \prod_{i}^{x.} (1 - R_{j,k}^{i\cdot\Delta})) \ ;$$

that is, an edge node will be spacing consecutive assembled data bursts on the same channel at least as wide as $T_{j,k}^{x.\Delta}$ time units. Data bursts subject to admission control must be scheduled on available wavelengths (channels). The SFC's scheduler performs as follows: an admitted data burst, $B_x(j,k)$, will be scheduled on the latest available wavelength where the interarrival time between the burst *x* passing through link *(j,k)* and the last scheduled burst, *y*, on the same wavelength is at least equal $T_{j,k}^{x.\Delta}$ time units. If no such wavelength exists, the burst must be further delayed until some units of time later. Note that when data bursts are not subject to admission control, $T_{j,k}^{x.\Delta} = 0$. Clearly, if a burst arrives after $T_{j,k}^{x.\Delta}$ time units, it will be conforming and thus the counters are reset and the burst will immediately be transmitted. These concepts are shown in Figure 3. For example, in Figure 3(a) data bursts B1, B2, B3, and B4 are already scheduled and new bursts B5, B6, and B7 arrive at times t1, t1, and t2, respectively. All new bursts are assumed to be passing through link *(j,k)*. The latest available channel for B5 to be scheduled is channel 1 (or 3). However, B5 cannot be scheduled before t3 on channel 1 (or 3). Thus, B5 will be delayed by one time unit and scheduled on channel 2 at t2. Figure 3(b) demonstrates a case where an incoming data burst, B7, can immediately be scheduled.

Once the congestion condition is resolved, the FRR is set to idle and the source nodes can resume transmission at full rate. A sudden simultaneous ramp-up of traffic by several edge nodes can result in severe link congestion again. Therefore, we consider using a random delay before each edge node resumes its full transmission rate. We define the following ramp-up function which governs the transmission rate increase pattern:

$$\phi_{ramp}^{x.\Delta+i} = \phi^{x.\Delta}[\tilde{\Gamma} \cdot u(\phi_{j,k}^{\Delta} - \phi_{j,k}^{x.\Delta+i-1}) + 1], \text{ for } i>0.$$

In this expression $\tilde{\Gamma}$ is a uniformly distributed random number between [0,1], $\phi^{x.\Delta}$ is the rate of transmission when ramping start, and $u(a-b) = 1$ for all b<a and zero otherwise. Note that the above equation suggests that the arrival rate continues

to increase until the pre-reduction transmission rate, ϕ^{Δ}, is achieved. Persistently long congestion conditions and slow ramp-up can significantly impact the average end-to-end packet delay. We will discuss these effects in more detail later.

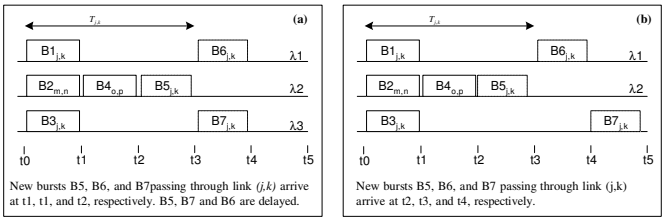

Figure 3. Illustrating the SFC-scheduler operation in the edge node when the new data bursts, B5, B6, and B7, arrive from the burst assembly unit.

3.1. Variations of the SFC scheme

In this section we discuss several variations of the SFC contention avoidance scheme. These schemes can also be implemented simultaneously in order to improve the network performance.

a) SFC with downstream notifier (SFCwDN): A simple improvement to the proposed SFC contention avoidance scheme is to notify the downstream core node that an FRR request has already been sent to the upstream source nodes. In this case, the downstream node computes the FRR based on the expected load value in the next time interval, Δ, and not the actual load it is currently measuring on the congested link. The expected load from congested upstream core node will be equivalent to the load threshold of the upstream switch, ρ_{TH}. Consequently, assuming all switches have the same load threshold, the value of $R_{j,k}$ for a core switch can be formulated as follows:

$$R_{j,k} = \frac{\sum_{i \in Pin} \min(\rho^{i}_{j,k}, \rho_{TH}) - \rho_{TH}}{\sum_{i \in Pin} \min(\rho^{i}_{j,k}, \rho_{TH})},$$

where P_{in} is the number of ingress ports on the switch and $\rho^{i}_{j,k}$ is the contributed load on link *(j,k)* by data bursts arriving on ingress port *i*. Although, in this approach there is no need to have unique counters for each ingress port, special hardware must be added to recognize the ingress port that each data burst is coming from.

b) SFC with individual flow counter (SFCwFC): Another variation of the SFC contention avoidance mechanism is to measure each ingress edge node's contribution to the measured load at the egress port and send a dedicated FRR to each source; this is equivalent to measuring the average load of each LSP passing through the switch. In this case, a core node must maintain *N-1* sets of information for each of its P_{eg} egress ports, where *N* is the number of edge nodes in the network. This will require as many as $(N\text{-}1 \cdot P_{eg})$ individual counters in the switch and considerable increase in the number of feedback messaging communicated between the nodes. The intuitive trade-off of this complexity is, however, achieving a better resource allocation among all edge nodes and protecting well-behaved edge nodes from malicious ones.

c) SFC with early dropping (SFCwED): When a link is congested and a flow-rate reduction request is sent to sources, it can take as many as several time intervals, Δ, before the congestion condition is cleared out. The longer the link congestion persists, the longer additional data bursts will be discarded and more resources will be wasted. Therefore, one way to quickly clear out the link congestion is to ask upstream adjacent nodes to temporarily drop data bursts passing through the congested link. This intentional (or early) burst dropping at upstream nodes will continue until source nodes reduce their transmission rate of burst flows. This scheme appears to be resource efficient in the sense that, if a link is congested, it will not be further used by the incoming data bursts from upstream nodes.

d) SFC with burst spreading (SFCwBS): Link contention can further be reduced by ensuring that the edge node transmits data bursts sharing the same congested link with minimum overlapping in time. Therefore, data bursts traveling through highly utilized links are delayed and spread out over time. The data burst spreading technique can be *random* or *deterministic* in which data bursts are spread out according to a defined variance. Obviously, the major disadvantage of data burst spreading is introducing higher end-to-end average delay.

3.2. Design parameters

The admission control mechanisms can be very sensitive to the parameter settings. In this section we evaluate the importance and affects of some of the design parameters.

The average elapsed time for the FRR to reach an edge node is proportional to the network diameter times the average transmission delay on each link. The transmission delay is defined as the time it takes a signal to travel between two adjacent nodes. Obviously, as this elapsed time increases, the network becomes less responsive to sudden load changes and thus less sensitive to the bursty nature of the traffic. Similarly, determining the value of the feedback trigger time, Δ, is critical. For example, if the value of Δ is too small, the number of feedback signals will increase. On the other hand, if Δ is too large, the feedback mechanism will be insensitive to the moderate changes in network load. Therefore, various factors including the network topology, traffic characteristic, and average transmission delay can be considered in determining the value of Δ.

The value of the switch load threshold, ρ_{TH}, also impacts the system performance. If the value of ρ_{TH} is too high, the admission control becomes less effective. On the other hand, if ρ_{TH} is very small, the admission control will be activated too quickly. This in turn, results in generating higher number of feedback messages, thereby increasing the control overhead in the network and leading to a potentially instable system.

When the measured traffic load on a link is around ρ_{TH}, any small changes in the offered load by the source on that link can

result in FRR oscillation. One way to prevent this is by setting a lower threshold, $\rho_{LOW} < \rho_{TH}$. Hence, the source will not be permitted to increase its traffic to the near-congested link unless the measured load drops below ρ_{LOW}. This can be achieved by setting the no-change bit in the FRR control field. The nc-bit will be set until the measured load at the switch is increased to some values above ρ_{TH} or below ρ_{LOW}. In the latter case, the nc-bit is cleared, and FRR will be set to idle indicating that the source can start ramping-up its transmission rate on the previously congested link. Figure 4(a) plots the measured load on link *(j,k)* by an intermediate switch. When the measured load exceeds ρ_{TH}, the flow-rate reduction requests are generated. The nc-bits are set during the time-intervals Δ4 and Δ5, when the measured load falls below ρ_{TH}.

A major trade-off of SFC is the introduction of admission control delay in order to reduce the data burst sending rates on the congested link *(j,k)*. We refer to this delay as *shaping delay* and express it as $SD_{j,k}$. As mentioned earlier, several parameters can impact the shaping delay, including Δ, efficiency of admission control strategy, and the ramp-up rate when the congestion condition is cleared out. Figure 4(b) shows that the shaping delay can be calculated as the area under the average data burst transmission rate by the edge node. That is

$$SD_{j,k} = \int_{\Delta}^{x\cdot\Delta} \phi_{j,k}^{t} \cdot dt \cdot$$

Obviously, as the shaping delay increases, the average required buffer size becomes larger due to admission control.

Correctly computing flow-rate reduction requests, as well as accurately representing them, are among important issues which deserve attention. Accurate computation of $R_{j,k}$ results in fast convergence and reduction of data burst flow-rate on a congested link and hence lowering the data burst loss. On the other hand, it is critical not to underutilize the network. In the following paragraphs we describe two approaches in which we can calculate the flow-rate reduction request.

a) The rate reduction request can simply represent the carried load on an output link of the switch. In this case the $R_{j,k}$ value does not include the number of unscheduled (or discarded) data bursts. Therefore, in order to reduce the data burst flow on the overloaded link, it may be necessary to send several FRRs. Consequently, load reduction occurs slowly, and more bursts are expected to be lost until the overload condition is resolved. This becomes more critical when the transmission delay through the network is significant.

b) Another approach to calculate $R_{j,k}$ is to compute the total load destined to each output link of the switch. Thus, based on the overall load, including all scheduled and unscheduled data bursts, destined to each link, the explicit reduction rate is calculated and sent back to each edge node. A major advantage of this scheme is its fast convergence property. That is, after the first FRR, we can expect the edge nodes to properly respond to the flow reduction request and reduce their load on the congested link, assuming there is no change in the network. However, the basic drawback of this approach is the need for larger counters monitoring each egress port and thus, higher hardware requirements.

Assuming the transmission delay in the network is significant and the core switches are not synchronized, the FRR generated by each switch only represents the average traffic load observed by the switch in the latest time interval. Therefore, it is conceivable that the downstream switch is not aware of any earlier reduction requests sent to the source by an upstream switch. In fact, depending on the number of physical hops, $|H_{s,n}|$, between the congested node, *s*, and the source node, *n*, due to transmission delay, $Td_{s,n}$, the requested rate reduction cannot be expected until at least $T\_FRR = 2 \cdot \sum_{j,k \in H_{s,n}} Td_{j,k}$ time units later.

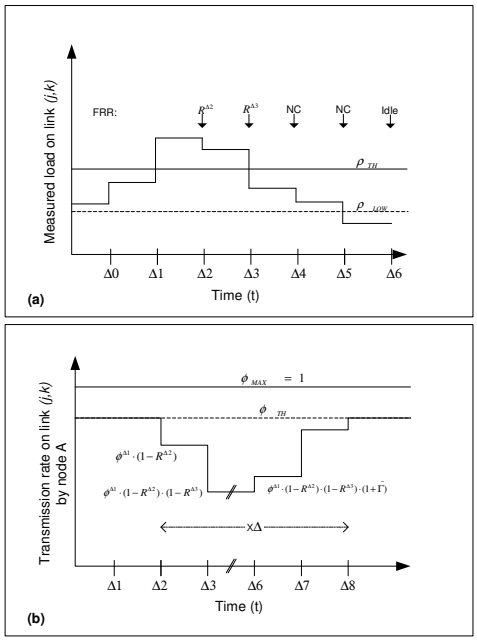

Figure 4. (a) FRR initiated at the switch based on the measured load; (b) changing the transmission rate at node A.

We emphasize that this delay may be several times larger than Δ. Hence, in order to avoid any sending rate oscillation at the source, FRR will not be sent (or the source will simply ignore new FRRs) unless $R_{j,k}^{\Delta+i} > R_{j,k}^{\Delta}$. We illustrate these concepts using the example shown in Figure 5.

Figure 5(a) shows a network of core switches S0-S5 and edge nodes A-E, all sending data bursts at different rates to link (4,5). We assume that links (2,3), (3,4), and (4,5) are congested. Thus, each switch, S2, S3, and S4, transmits a rate reduction request back to its reachable edge node, including node A, requesting it to reduce its transmission rate on the congested link. Figure 5(b) shows the timing diagram for the

FRR signals between the intermediate nodes and edge node A. Note that node A receives different rate reduction requests from each intermediate switch at different times. For example, the FRR from switch S4 will reach node A after $T\_FRR_{S4,A}$.

In general, the reduction request delay between a core node and a source with $|H|$ physical hops in between is bounded by the following expression:

$$|H|\cdot(Td+\Delta)+T_{proc} \le T\_FRR \le Td\cdot|H|+\Delta+T_{proc}$$

,

where T_{proc} is the processing time required for the edge node to adjust its transmission rate.

Assuming that the average transmission rate through the network is unchanged within the time period *T_FRR*, node A continues receiving other FRRs from S4 which could be less than the initial $R_{4,5}$ value. In order to ensure that the source does not continue reducing its burst flow on link (4,5), The nc-bit will be set indicating that a reduction request larger than the current value has already been sent to the source. Hence, when the nc-bit is set, the source may disregard the FRR. When the measured load on the link eventually reaches the desired threshold level, ρ_{TH}, the FRR value is set to idle, indicating that the congestion condition on the link is removed.

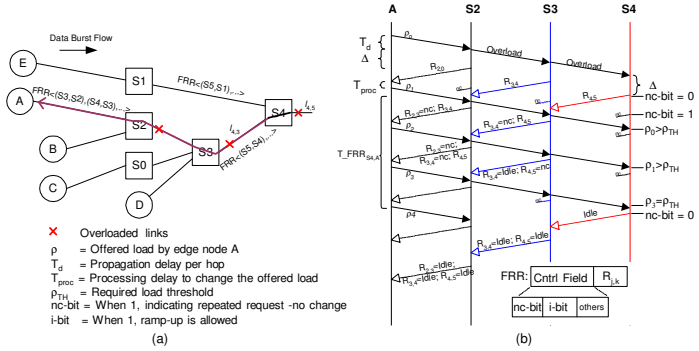

Figure 5. (a) Example of a network with overloaded links; (b) FRR timing diagram between intermediate nodes and edge node A.

4. Performance Results

In this section we discuss the simulation results obtained by implementing the proposed data burst admission control in a feedback-based OBS network. We consider a simple network topology, shown in Figure 6, as our test network and consider the following assumptions for the simulation environment: burst length is fixed and is equivalent to 100 µs, containing 1250 bytes; the transmission rate is 10 Gb/s with 4 wavelengths on each link, the switching time is 10 µs, and the burst header processing time at each node is assumed to be 2.5 µs. Furthermore, we assume full wavelength conversion at every node and adopt the latest available unscheduled channel (LAUC) algorithm to schedule data bursts at the core nodes.

4.1. Traffic model

The traffic model we consider in our study is characterized by two random processes modeling both the spatial and the temporal characteristics of the arriving data bursts. The spatial characteristic, which indicates the distribution of data bursts destinations is modeled by a uniform distribution. On the other hand, the process of modeling the inter-arrival times between successive data burst arrivals is based on a two-state Markov chain, as shown in Figure 7, consisting of a HIGH and LOW state. In the HIGH state, assembled data bursts arrive at rate λ_H, which is higher than the average arrival $\overline{\lambda}$ rate . In the LOW state, fewer IP packets arrive and thus burst arrival occurs at $\lambda_L < \overline{\lambda}$.

In each state we consider exponentially distributed burst inter-arrival times. Similarly, the time that the system remains in each state is exponentially distributed. The average data burst arrival rate in this model is determined by $\overline{\lambda} =_H \cdot\mu_H + \mu_L \cdot \lambda_L$, where the state probabilities μ_L and μ_H are computed as follows:

$$\mu_H = \frac{\mu_{HL}}{\mu_{HL}+\mu_{LH}} \text{ and } \mu_L = \frac{\mu_{LH}}{\mu_{HL}+\mu_{LH}}.$$

Thus, the average data burst arrival rate will be

$$\overline{\lambda} = \lambda_H \frac{\mu_{LH}}{\mu_{HL}+\mu_{LH}} + \lambda_L \frac{\mu_{HL}}{\mu_{HL}+\mu_{LH}}.$$

Three possible scenarios can be considered:

a) $\lambda_H = \lambda_L$; In this case the model is reduced to an exponential arrival with fixed size data bursts.

b) $1 > \lambda_H > \lambda_L > 0$; In this case the arrival rate varies between λ_H and λ_L as the time increases. We refer to $\alpha = \lambda_H / \lambda_L$ as the *traffic persistency factor*. Note that when $\alpha = 1$ we obtain exponential arrival model and as α becomes larger than 1 the traffic burstiness increases.

c) $\lambda_H = 1 \& \lambda_L = 0$; This case represents an ON-OFF bursty traffic model in which bursts of traffic arrive in the state HIGH (ON). No traffic is generated in the LOW (OFF) state.

In this study we only focus on cases (a) and (b).

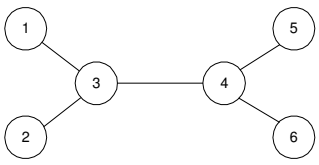

Figure 6. Network model with 5 WDM links.

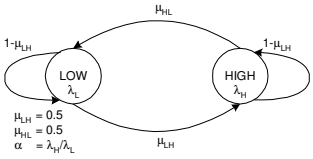

Figure 7. Two-state Markov chain traffic model.

4.2. Simulation results

In our analysis we only consider the basic SFC algorithm introduced in Section 3. In addition, we assume that the FRR

signal reflects the overall load, including all scheduled and unscheduled data bursts, destined to each link. All results are based on monitoring the total data burst flows along the most congested bottleneck link (3,4). If the measured load on the link goes beyond the threshold value, the rate reduction request is generated and sent to the source.

Figure 8 shows the probability of burst loss when the data bursts are exponentially arriving. This figure compares the probability of data burst loss with and without the SFC-based congestion avoidance mechanism. As the load threshold in the switch drops, the loss probability decreases. This occurs as a result of *choking* the source and lowering the maximum transmission rate allowed on the bottleneck link. However, the trade-off is lowering the link throughput, as shown in Figure 9. This figure shows the normalized throughput for an exponentially distributed traffic model with and without the contention avoidance mechanism. The total link throughput (that is the maximum achievable data link capacity) in our model is 40Gb/s. The value of the load threshold of the switch directly impacts the network throughput. For example, as shown in Figure 9, when the threshold is set to 0.7, the throughput of the bottleneck link at high loads will be 0.59(40Gb/s) = 23.6Gb/s, compared to 0.81(40Gb/s) = 32.4Gb/s when no contention avoidance is implemented. However, note that the loss at ρ_{TH} = 0.7 is significantly decreased (from 35 to about 10 percent). Note that as the load increases, the measured load remains above the threshold, and the system remains in continuous choking state. Thus, reduction requests are constantly generated, and the measured load stays below the threshold level. Also note that the throughput about the threshold point is slightly more with the congestion avoidance mechanism in place. This is because when the system is not completely overloaded, some data bursts can be buffered and sent at a later time, resulting in higher overall throughput.

Similar results in terms of data burst loss probably and link throughput can be observed when the traffic is exponentially arriving with high and low instant arrivals, as shown in Figure 10 and Figure 11. When the threshold value is low, such as 0.6, as the measured load on the bottleneck reaches the threshold, the probability of loss continues to increase until the links are overloaded and the system goes into the choke state. The probably of loss in case of exponentially distributed traffic with high/low instant averages experiences more variations around the threshold level. This is because in order for the system to go into chock state higher average load is required.

5. Conclusion

In this paper we proposed a feedback-based contention avoidance mechanism for optical burst switching networks. Our proposed scheme, known as source flow-rate control (SFC), significantly reduces the packet loss probability in the OBS network. The basic trade-off of SFC is, however, the overall reduction of network utilization due to invoking admission control when the network is congested. Through simulation, we compared the overall data burst loss with and without the SFC contention avoidance mechanism, and show that network throughput reduction, due to admission control, is tolerable. The performance of our proposed mechanism is expected to improve when it is implemented with data burst spreading or intentional data burst dropping on adjacent nodes when a downstream link is subject to congestion. These and other topics, including fairness and SFC's behavior in large systems will be the focus of our future work.

References

[1] F. C. Qiao and M. Yoo, "Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet," *Journal of High Speed Networks*, vol. 8, no.1, pp. 69-84, Jan. 1999.

[2] J.S. Turner, "Terabit Burst Switching," *Journal of High Speed Networks*, vol. 8, no. 1, pp. 3-16, Jan. 1999.

[3] Y. Xiong, M. Vanderhoute, and 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, Oct. 2000.

[4] F. Farahmand, V. M. Vokkarane, and J. Jue, "Practical Priority Contention Resolution for Slotted Optical Burst Switching Networks, " *Proceedings, IEEE/SPIE First International Workshop on Optical Burst Switching (WOBS) 2003*, Dallas, TX, October. 2003.

[5] C. Gauger, "Dimensioning of FDL Buffers for Optical Burst Switching Nodes," *Proceedings, Optical Network Design and Modeling (ONDM) 2002*, Torino, Italy, 2002.

[6] S. Keshav, *An Engineering Approach to Computer Networking*, Addison-Wesley, Reading, Mass., 1997.

[7] S. Verma, H. Chaskar, and R. Ravikanth , "Optical burst switching: a viable solution for terabit IP backbone", *IEEE Network*, vol. 14, no. 6, pp. 48-53, Nov. 2000.

[8] T. Ozugur, F. Farahmand, and D. Verchere, "Single-anchored soft bandwidth allocation system with deflection routing for optical burst switching," *IEEE Workshop on High Performance Switching and Routing (HPSR) 2002*, pp. 257-261, May 2002.

[9] S. Yao, B. Mukherjee, S. J. Ben Yoo, and S. Dixit, "All-optical packet switching for Metropolitan Area Networks: Opportunities and Challenges," *IEEE Communication Magazine*, vol. 39, no. 3, pp. 142-148, March 2001.

[10] S. Kim, N. Kim, and M. Kang, "Contention Resolution for Optical Burst Switching Networks Using Alternative Routing", *Proceedings, IEEE International Conference on Communications (ICC)*, New York, NY, April-May 2002.

[11] X. Wang, H. Morikawa, and T. Aoyama, "A Deflection Routing Protocol for Optical Bursts in WDM Networks," *Proceedings. Fifth Optoelectronics and Communications Conference (OECC) 2000*, Makuhari, Japan, pp. 94-95, July 2000.

[12] C. Hsu, T. Liu, and N. Huang, "Performance Analysis of Deflection Routing in Optical Burst-Switched Networks," *Proceedings of INFOCOM 2002*, pages 66–73, Jun 2002.

[13] X. Wang, H. Morikawa, and T. Aoyama, "Photonic Burst deflection routing protocol for wavelength routing networks", *SPIE Optical Networks Magazine*, vol. 3, no. 6, pp. 12-19, Nov-Dec. 2002.

[14] V. M. Vokkarane, J. P. Jue, and S. Sitaraman, “Burst Segmentation: An Approach for Reducing Packet Loss In Optical Burst Switched,” *Proceedings, IEEE International Conference on Communications (ICC) 2002,* New York, NY, vol. 5, pp. 2673-2677, April 2002.
[15] F. Farahmand and J. P. Jue, “Look-ahead Window Contention Resolution in Optical Burst Switched Networks,” *IEEE Workshop on High Performance Switching and Routing (HPSR) 2003*, Torino, Italy, pp. 147-151, June 2003.
[16] F. Farahmand and J. P. Jue, "Supporting QoS with Look-ahead Window Contention Resolution in Optical Burst Switched Networks," *Proceedings, IEEE GLOBECOM 2003*, San Francisco, CA, December, 2003.
[17] A. Ge, F. Callegati, and L. Tamil. “On optical burst switching and self similar traffic,” *IEEE Communications Letters,* vol 4, no. 3, pp. 98-100, March 2000.
[18] V. M. Vokkarane, K. Haridoss, and J. P. Jue, “Threshold-Based Burst Assembly Policies for QoS Support in Optical Burst-Switched Networks,” *Proceedings, SPIE Optical Networking and Communication Conference (OptiComm) 2002*, Boston, MA, vol. 4874, pp. 125-136, July-Aug. 2002.
[19] X. Yu, Y. Chen, and C. Qiao, “A Study of Traffic Statistics of Assembled Burst Traffic in Optical Burst Switched Networks,” *Proceedings, SPIE Optical Networking and Communication Conference (OptiComm) 2002*, Boston, MA, pp. 149-159, July-Aug 2002.
[20] X. Cao, J. Li, Y. Chen, and C. Qiao, ”Assembling TCP/IP packets in optical burst switched networks” *Proceedings, IEEE GLOBECOM 2002*, vol. 3 , pp. 2808 -2812, November 2002.
[21] S. Oh and M. Kang, “A Burst Assembly Algorithm in Optical Burst Switching Networks,” *Proceedings, Optical Fiber Communication Conference and Exhibit (OFC) 2002*, pp. 771-773, March, 2002.
[22] M. Elhaddad, R. Melhem, T. Znati, and D. Basak "Traffic shaping and scheduling for OBS-based IP/WDM Backbones," *Proceedings, SPIE Optical Networking and Communication Conference (OptiComm) 2003*, Dallas, TX, vol. 5285, pp. 336-345, October 2003.
[23] R. Jain and K. K. Ramakrishnan, “Congestion avoidance in computer networks with a connectionless network layer: concepts, goals, and methodology,” *Proceedings, Computer Networking Symposium 1988*, pp. 134 -143, 11-13 April 1988.
[24] G. Thodime, V. M. Vokkarane, and J. P. Jue, "Dynamic Congestion-Based Load Balanced Routing in Optical Burst-Switched Networks," *Proceedings, IEEE GLOBECOM 2003*, San Francisco, CA, December, 2003.
[25] J. Li, G. Mohan, and K. C. Chua "Load Balancing Using Adaptive Alternate Routing in IP-over-WDM Optical Burst Switching Networks," *Proceedings, SPIE Optical Networking and Communication Conference (OptiComm) 2003*, Dallas, TX, vol. 5285, pp. 336-345, October 2003.
[26] A. Detti and M. Listanti, "Impact of Segments Aggregation on TCP Reno Flows in Optical Burst Switching Networks", *Proceedings of INFOCOM 2002*, New York, NY, June 2002.
[27] X. Cao, J. Li, Y. Chen, and C. Qiao, "Assembling TCP/IP Packets in Optical Burst Switched Networks", *Proceedings, IEEE GLOBECOM 2002*, Taipei, Taiwan, November 2002.
[28] S. Y. Wang, “Using TCP congestion control to improve the performances of optical burst switched networks,” *Proceedings, IEEE International Conference on Communications (ICC) 2003*, vol. 2, pp. 1438-1442, 11-15 May 2003.
[29] C. Qiao, "Labeled Optical Burst Switching for IP and WDM Integration", *IEEE Communications Magazine*, vol. 38, no. 9, pp. 104-114, Sept. 2000.
[30] A. Okada, "All-optical packet routing in AWG-based wavelength routing networks using an out-of-band optical label", *Proceedings, Optical Fiber Communication Conference and Exhibit (OFC) 2002*, Anaheim, CA, March 2002.

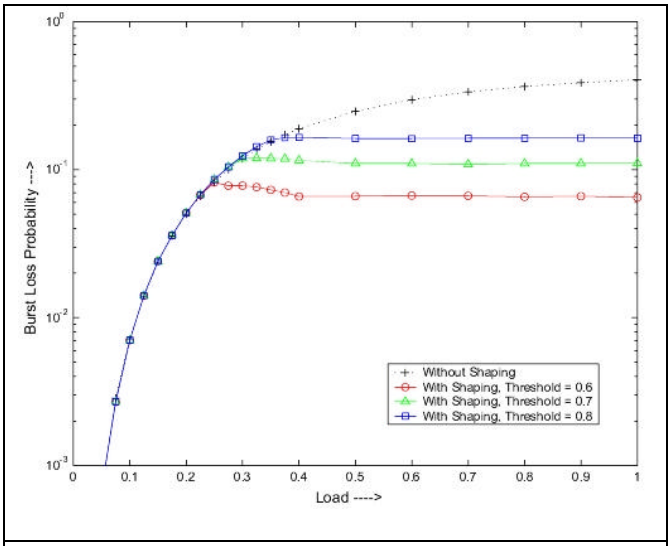

Figure 8. Comparing the probability of data burst loss with and without contention avoidance when the traffic is exponentially arriving.

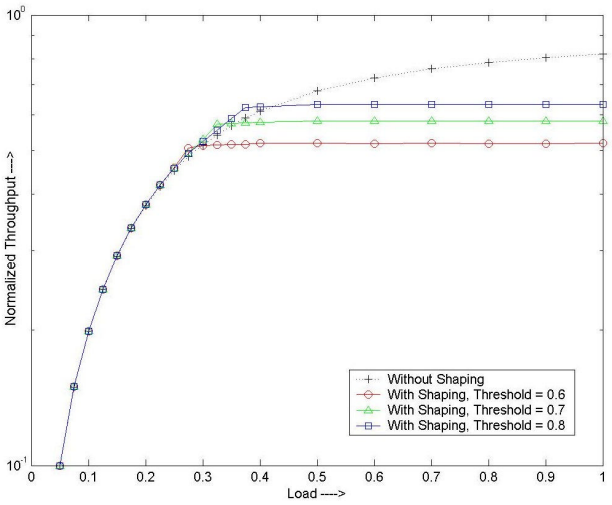

Figure 9. Normalized throughput when the traffic is exponentially arriving.

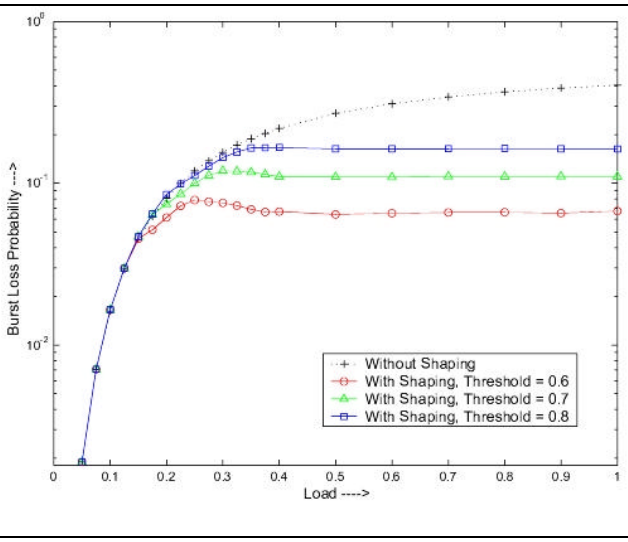

Figure 10. Comparing the probability of data burst loss with and without contention avoidance with variant rate traffic when the persistent factor is 3.

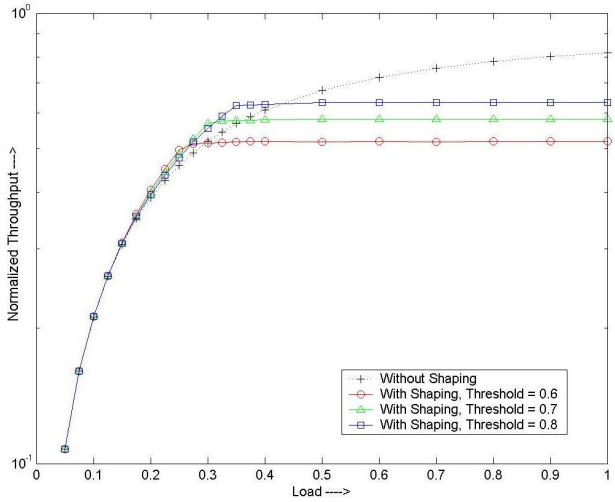

Figure 11. Normalized throughput when the persistent factor is 3.