Burst Segmentation: An Approach for Reducing Packet Loss in Optical Burst-Switched Networks

Vinod M. Vokkarane and Jason P. Jue
Department of Computer Science, The University of Texas at Dallas
Richardson, TX 75083
vinod@utdallas.edu, jjue@utdallas.edu

***Abstract*—In this paper, we address the issue of contention resolution in optical burst switched networks, and we introduce an approach for reducing packet losses which is based on the concept of burst segmentation. In burst segmentation, rather than dropping the entire burst during contention, the burst may be partitioned into multiple segments, and only the overlapping segments are dropped. The segmentation scheme is investigated in conjunction with a deflection scheme through simulation, and it is shown that segmentation with deflection can achieve a significantly reduced packet loss rate.**

Index Terms: Optical burst switching, WDM, contention resolution, deflection, burst segmentation.

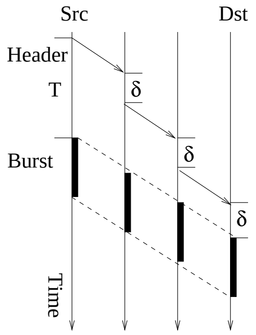

Fig. 1. The use of offset time in OBS.

I. INTRODUCTION

The amount of raw bandwidth available on fiber optic links has increased dramatically with advances in dense wavelength division multiplexing (DWDM). In order to efficiently utilize this bandwidth in a cost-effective manner for IP traffic, an appropriate all-optical transport method must be developed. This transport method must be able to handle asynchronous bursty traffic by quickly provisioning resources while also minimizing the use of optical buffering. Optical burst switching (OBS) is one such method for transporting traffic directly over a bufferless optical core network [1].

In an optical burst switched network, bursts of data consisting of multiple packets are switched through the network all-optically. A control message (or header) is transmitted ahead of the burst in order to configure the switches along the burst's route. The data burst follows the header after some offset time without waiting for an acknowledgment for the connection establishment. The offset time (Fig. 1) allows for the header to be processed at each node while the burst is buffered electronically at the source; thus, no fiber delay lines are necessary at the intermediate nodes to delay the burst while the header is being processed. The control message may also specify the duration of the burst in order to let a node know when it may reconfigure its switch for the next burst, a technique known as *just enough time* (JET) [1]. In this paper, we will consider an optical burst-switched network which uses the JET technique.

A major concern in optical burst switched networks is contention, which occurs when multiple bursts contend for the same link. In this paper, we will refer to the burst which arrives to the switch first as the *original burst*, and the burst which arrives to the switch later as the *contending burst*. Contention in an optical burst switched network is particularly aggravated by the variable burst sizes and the long burst durations. Furthermore, since bursts are switched in a cut-through mode rather than a store-and-forward mode, optical burst-switched networks generally have very limited buffering capabilities.

Typically, contention in traditional electronic packet-switching networks is handled through buffering; however, in the optical domain, it is more difficult to implement buffers, since there is no optical equivalent of random-access memory. Instead, optical buffering is achieved through the use of fiber delay lines [2], [3]. Current optical buffer architectures are severely limited in size; thus, nodes in an all-optical network may be unable handle high load or bursty traffic without alternative contention resolution schemes. Such alternative schemes include *wavelength conversion* and *deflection routing*. With wavelength conversion, contention is reduced by utilizing additional capacity in the form of multiple wavelengths per link [4], [5], [6]. A contending burst may be switched to any of the available wavelengths on the outgoing link. While optical wavelength conversion has been demonstrated in laboratory environments, the technology is not yet mature, and the range of possible conversions are somewhat limited [7]. In deflection routing, contention is resolved by routing data to an output port other than the intended output port. Deflection routing is generally not favored in electronic packet-switched networks due to potential looping and out-of-sequence delivery of packets; however, it may be necessary to implement deflection in all-optical burst-switched networks, where buffer capacity is very limited. While existing contention resolution schemes, such as deflection and buffering, may be utilized in optical burst switched networks, additional schemes may still be necessary in order to further reduce high contention rates and to achieve high network utilization.

A number of previous works have addressed the issue of contention resolution in optical burst-switched networks. In [8], an offset scheme was proposed for isolating classes of bursts, such that low-priority bursts do not cause contention losses for high-priority bursts; fixed and variable fiber delay line buffers were also utilized to further reduce blocking. In [8] and [9] contention is reduced by utilizing additional capacity in the form of multiple wavelengths. In both cases, optical wavelength conversion was assumed.

In the current literature, most approaches to contention resolution address the minimization of burst losses rather than packet losses. In existing contention resolution schemes for optical burst switched networks, when a contention between two bursts cannot be resolved through other means, one of the bursts will be dropped in its entirety, even though the overlap between the two bursts may be minimal. For certain applications which have stringent delay requirements but relaxed packet loss requirements, it may be preferable to lose a few packets from a given burst rather than losing the entire burst. In this paper, we will introduce a new contention resolution technique called *burst segmentation*, in which only those packets which overlap with a contending burst will be dropped.

The paper is organized as follows. Section II introduces the concept of burst segmentation and describes the segment dropping policies. Section III discusses segmentation with deflection. Section IV compares the simulation results for different contention resolution policies in a specific network topology, and Section V concludes the paper.

II. BURST SEGMENTATION

To overcome some of the limitations of optical burst switching, we introduce the concept of burst segmentation. In burst segmentation, the burst consists of a number of basic transport units called segments. Each segment consist of a segment header and a payload. The segment header contains fields for synchronization bits, error correction information, source and destination information, and the length of the segment in the case of variable length segments. The segment payload may carry any type of data, such as IP packets or ATM cells (Fig. 2). When two bursts contend with one another in the optical burst-switched network, only those segments of one burst which overlap with the other burst will be dropped, as shown in Fig. 3. If switching time is non-negligible, then additional segments may be lost when the output port is switched from one burst to another.

In order to maintain data and format transparency, the optical layer need not be aware of the actual segment boundaries and segment payload data format. In this case, the optical layer is only aware of information such as the burst source and destination nodes, the burst offset time, the burst duration, and possibly the burst priority. This transparency may lead to sub-optimal decisions with regard to minimizing data loss, as individual segments may end up being split into two parts, resulting in complete data loss for those segments; however, by maintaining transparency, the optical layer remains fairly simple, and no significant additional computational overhead will be required at each node.

If the segment boundaries are transparent in the all-optical core, then the nodes at the network edge must be responsible for defining and processing segments electronically. Furthermore, the receiving node must be able to detect the start of each segment and identify whether or not the segment is intact; thus, some type of error detection or error correction overhead must be included in each segment. One possible implementation of segmentation is to define a segment as an Ethernet frame. If each segment consists of an Ethernet frame, then detection and synchronization can be performed by using the preamble field in the Ethernet frame header, while errors and incomplete frames can be detected by using the CRC field in the Ethernet frame; thus, no further control overhead would be required in each segment other than the overhead already associated with an Ethernet frame.

If segments are not defined as Ethernet frames, then the choice of the segment length becomes a key system parameter. The segment can be either fixed or variable in length. If segments are fixed in length, synchronization at the receiver becomes easier; however, variable-length segments may be able to accommodate variable-length packets in a more efficient manner. The size of the segment also offers a trade off between the loss per contention and the amount of overhead per burst. Longer segments will result in a greater amount of data loss when segments are dropped during contention; however, longer segments will also result in less overhead per segment, as the ratio of the segment header length to the segment payload length will be lower. In this paper, we assume that each segment is an Ethernet frame which contains a fixed-length packet, and we do not address the issue of finding the optimal segment size.

Another issue in burst segmentation is the decision of which burst segments to drop when a contention occurs between two bursts. Two possible approaches include *tail-dropping*, in which the tail segments of the original burst (Fig. 3) are dropped, and *head-dropping*, in which the head segments of the contending burst are dropped. An advantage of dropping the tail segments of bursts rather than the head segments is that there is a better chance of in-sequence delivery of packets at the destination, assuming that dropped packets are retransmitted at a later time. A head-dropping policy will result in a greater likelihood that packets will arrive at their destination out of or-

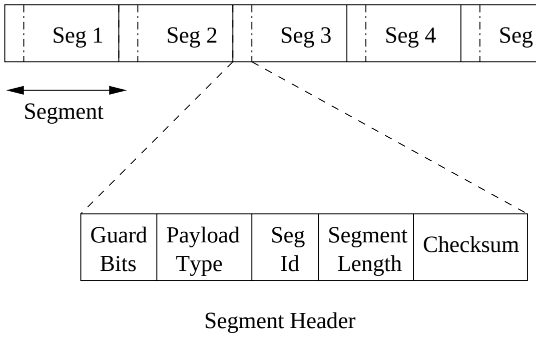

Fig. 2. Segments header details.

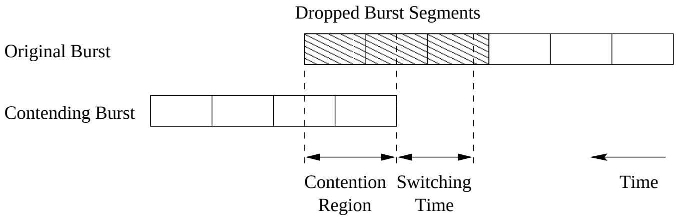

Fig. 3. Selective segment dropping for two contending bursts.

der; however, the advantage of head-dropping is that it ensures that, once a burst arrives at a node without encountering contention, then the burst is guaranteed to complete its traversal of the node without preemption by later bursts.

In this paper, we consider a modified tail-dropping policy when determining which segment to drop. In this policy, the tail of the original burst is dropped only if the number of segments in the tail is less than the total number of segments in the contending burst. If the number of segments in the tail is greater than the number of segments in the contending burst, then the entire contending burst is dropped. This approach reduces the probability of a short burst preempting a longer burst and minimizes the number of packets lost during contention.

One issue that arises when the tail of a burst is dropped is that the header for the burst, which may be forwarded before the segmentation occurs, will still contain the original burst length; therefore, downstream nodes may not know that the burst has been truncated. If downstream nodes are unaware of a burst's truncation, then it is possible that the previously truncated tail segments will contend with other bursts, even though these tail segments have already been dropped at a previous node. These contentions may result in unnecessary packet loss.

If a tail-dropping policy is strictly maintained throughout the network, then the tail of the truncated burst will always have lower priority, and will never preempt segments of any other burst. However, for the case in which tail dropping is not strictly maintained, some action must be taken to avoid unnecessary packet losses. A simple solution is to have the truncating node generate and send out a *trailer*, or a trailing control message, to indicate when the truncated burst ends. The trailer is created electronically at the switch where the contention is being resolved, and the time to create the trailer can be included in the offset for header processing time, δ, at each node. The trailer is necessary only if the modified-tail dropping approach is adopted. If head-dropping is employed, then the header of the truncated burst may be updated immediately at the contention node. If strict tail-dropping is employed, then the dropped tail segments will always lose contention and will never preempt other segments.

We note that, even if a trailer is created, the trailer may not be completely effective in eliminating contentions with segments that have already been dropped. Fig. 4 shows the situation in which the trailer packet reaches the downstream node before the header of a contending burst. As soon as the trailer packet is received, the node is updated with the new length of the original burst; hence, when the control header of the contending burst arrives, the virtual contention is avoided. In the case of Fig. 5, the header of the contending burst arrives before the trailer of the original burst at the downstream node; hence the switch detects a contention, even though the tail packets of the original burst have already been dropped. Although the trailer packet does not completely eliminate the situation of a virtual contention, as in the latter case, the trailer can minimize such situations; hence it is important to generate and transmit the trailer as soon as possible at the upstream node.

An additional system parameter which has a significant effect on burst segmentation is the switching time. Since the system does not implement buffering or any other delay mechanism, the switching time is a direct measure of the number of packets lost during reconfiguring the switch due to contention. Hence, a slow switching time will result in higher packet loss, while a fast switching time will result in lower packet loss. Current all-optical switches using MEMs technology are capable of switching on the order of milliseconds, while switches using semiconductor optical amplifier (SOA) technology are capable of switching on the order of nanoseconds. Due to their high switching times, MEMs switches may not by very suitable for optical burst switching, and are more appropriate for circuit-switched optical networks. On the other hand, SOA switches have been demonstrated in laboratory experiments [10], but have yet to be deployed in practical systems. In our simulations, we assume an intermediate and more practical switching time of 10 microseconds.

III. SEGMENTATION WITH DEFLECTION

A basic extension of burst segmentation is to implement segmentation with deflection. Rather than dropping segments of a burst, we can either deflect the entire burst or deflect segments of the burst to an output port other than the intended output port. This approach is referred to as deflection routing or hot-potato routing [4], [5], [6]. Implementing segmentation with deflection (Fig. 6) increases the probability that the burst will reach the destination, and hence, may improve the performance. One problem which may arise is that a burst may encounter looping or may be deflected multiple times, thereby wasting network bandwidth. This increased use of bandwidth

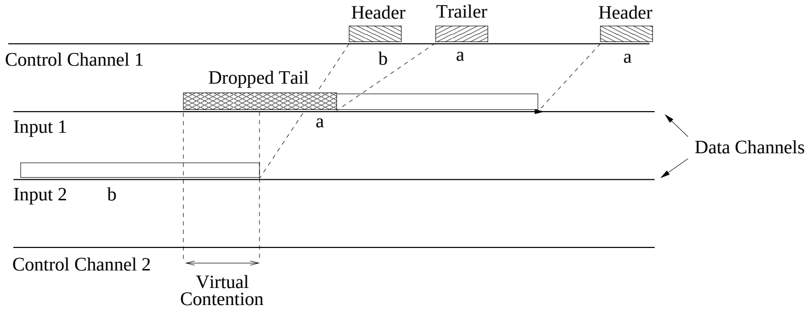

Fig. 4. Trailer packet effective.

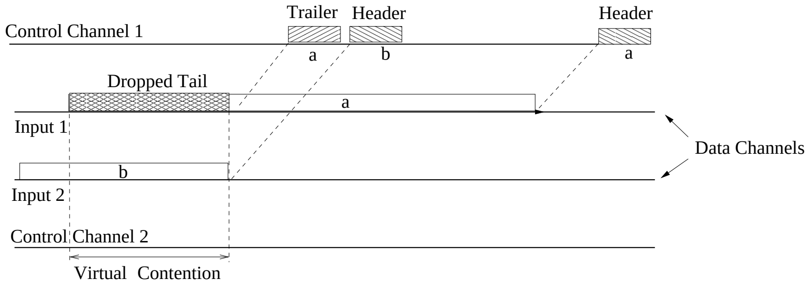

Fig. 5. Trailer packet ineffective.

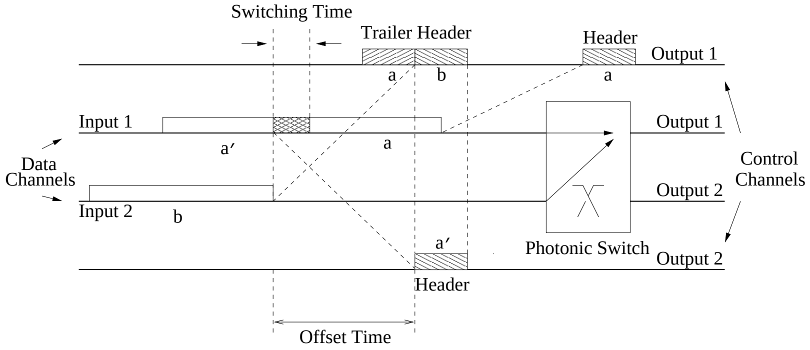

Fig. 6. Segmentation with deflection policy for two contending bursts.

can lead to increased contention and packet loss under high load conditions [11]. Due to deflection, the burst may also traverse a longer route, thereby increasing the total processing time. This may lead to a situation in which the initial offset time is insufficient to transmit the data burst all-optically without storage. In order to avoid these problems, the burst will be dropped when the hop-count of the burst reaches a certain threshold.

When a burst is deflected, a deflection port must be selected. There may be one or many alternate deflection ports. The alternate deflection ports can either be determined ahead of time using a fixed port-assignment policy, which chooses the port based on the next shortest path, or determined dynamically using a load balanced approach, which deflects the burst to an under-utilized link. In this paper, we consider only one alternate deflection port, and choose the port which results in the second shortest path to the destination.

Selection of which burst (or burst-segments) to deflect during contention may be done in one of two ways. The first approach is to deflect the burst with the shorter remaining length (taking switching time into account). If the alternate port is busy, the burst may be dropped (Fig. 6). The second approach is to incorporate priorities into the burst. In this case, the lower-priority burst is deflected or segmented.

When combining segmentation with deflection, there are two approaches for ordering the contention resolution policies, namely, *segment-first* and *deflect-first*. In the segment-first policy, if the remaining length of the original burst is shorter than the contending burst, then the original burst is segmented and its tail is deflected. In case the alternate port is busy, the deflected part of the original burst is dropped. If the contending burst is shorter than the remaining length of th original burst, then the contending burst is deflected or dropped. In the deflect-first policy, the contending burst is deflected if the alternate port is free. If the alternate port is busy and if the remaining length of the original burst is shorter than the length of the contending burst, then the original burst is segmented and its tail is dropped. If the contending burst was found to be shorter, then the original burst is dropped.

An example of the segmentation-deflection scheme is shown in Fig. 6. Initially when the header for *Burst A* arrives at the switch, it is routed onto Output Port 1. Once the header of *Burst B* arrives at the switch there is a contention. Since the offset time is common to all of the bursts, the header indicates when and where the bursts will contend. Therefore, by taking the switching time into consideration, and by using the segment-first policy, one of the bursts will deflected (or segmented and deflected) to the alternate port if the alternate port is free and will be dropped if the alternate port is not free. Here, the remaining length of *Burst A* is less than the length of *Burst B*. Hence *Burst A* is segmented and its tail is deflected to the alternate port as a new burst. A header is created for the deflected new burst and is sent on Output Port 2. This new header is generated at the time that the header of *Burst B* is processed. A trailer is created for the segmented *Burst A* and is sent on the control channel of Output Port 1. Packets of the segmented burst are lost during the reconfiguration of the switch. In the policy that utilizes both segmentation and deflection, the processing time δ (Fig. 1) at each node includes the time to create a header for the new burst segment in the case of a contention; hence the offset time remains the same as in the case of standard optical burst switching.

A possible side-effect of segmentation with deflection is that, when there is a contention, the shorter remaining burst will be segmented and will be deflected as a new burst. Creating these new bursts may lead to burst fragmentation, in which there are many short bursts propagating through the network. These short bursts will incur higher overhead with respect to switching times and control overhead per burst. Furthermore, having a greater number of smaller bursts in the network will also increase the number of control packets. These additional control packets may overload the control plane; hence, it may be advisable to drop the segmented burst if the new burst length is lower than a minimum burst size.

Fragmentation may be somewhat alleviated by utilizing the modified tail-dropping policy. In the modified tail-dropping policy, the lengths of the two contending bursts are compared and the smaller of the contending burst or the remaining part of the original burst is deflected or segmented respectively. If a deflection port is unavailable, then the segments that lose the contention will be dropped. Thus, the short, fragmented bursts are more likely to be dropped, and will not significantly hinder other bursts.

Another issue in deflecting bursts is maintaining the proper offset between the header and payload of a deflected burst. Since the deflected burst must traverse a greater number of hops than if the burst had not been deflected, there may be a point at which the initial offset time may not be sufficient for the header to be processed and for the switch to be reconfigured before the data burst arrives to the switch. In order to eliminate problems associated with insufficient offset time, a number of different policies may be implemented. One approach is simply to discard the burst if the offset time is insufficient. Counter and timer-based approaches may also be used to detect and limit the number of hops that a burst experiences. If the goal is to minimize packet loss, then the head of the burst can simply be truncated while a switch is being configured, and the tail segments of the burst can continue through the network. Buffering approaches using fiber delay lines (FDLs) may also be applied; however, such approaches increase the complexity of the optical layer.

Another issue when implementing segmentation and deflection is how to handle long bursts which may span multiple nodes simultaneously. If a long burst passing through two or more switches experiences contention from two or more different bursts at different switches, then, based on the timing of these contentions, the contentions may be resolved in a number of ways. If an upstream node segments the burst first, then the downstream nodes are updated by the trailer packet to eliminate unnecessary contentions. On the other hand, if the contention occurs at the downstream node before the upstream node, and if the burst's tail is deflected at the downstream node, then the upstream contentions will not be affected. If the downstream node drops the tail of the burst, then the upstream node will not know about the truncation and will continue to transmit the tail. The downstream node may send a control message to the upstream node in order to reduce unnecessary contentions with the tail at the upstream node. In the case where more than two bursts contend at the same switch, the contention is handled sequentially.

One possible advantage of segmentation in optical burst-switched networks is that it can provide an additional degree of differentiation for supporting different quality of service (QoS) requirements [12]. When two bursts contend with one another, the burst priority can be used to determine which burst to segment or drop. For example, if a high priority burst arrives to a node and finds that a low priority burst is being transmitted on the desired output, then the low priority burst can be segmented, and its tail can be dropped, in order to transmit the high priority burst. On the other hand, if a low priority burst arrives to a node and finds a high priority burst being transmitted, then the low priority burst will be dropped. When combining segmentation with deflection, and even greater degree of differentiation may be achieved. The choice of whether to deflect the newly arriving contending burst, or the tail of the burst currently being transmitted, can be made based on priorities.

We propose the following five different policies for handling contention in the OBS network:

1. *Drop Policy (DP):* Drop the entire contending burst.
2. *Deflect and Drop Policy (DDP):* Deflect the contending burst to the alternate port. If the port is busy, drop the burst.
3. *Segment and Drop Policy (SDP):* The contending burst wins the contention. The original burst is segmented, and its segmented tail is dropped.
4. *Segment, Deflect and Drop Policy (SDDP):* The original

burst is segmented, and its segmented tail may be deflected if an alternate port is free, otherwise the tail is dropped.

5. *Deflect, Segment and Drop Policy (DSDP):* The contending burst is deflected to a free port if available, otherwise the original burst is segmented and its tail is dropped, while the contending burst is transmitted.

IV. SIMULATION RESULTS

In order to evaluate the performance of the segmentation and deflection schemes, we develop a simulation model. The following have been assumed to obtain the results:

- Burst arrivals to the network are Poisson.
- Burst length is an exponentially generated random number rounded to the nearest integer multiple of the fixed packet length with an average burst length of 100 μs.
- Transmission rate is 10 Gbps.
- Packet length is 1500 bytes.
- Switching time is 10 μs.
- There is no buffering or wavelength conversion at nodes.
- Traffic is uniformly distributed over all source-destination pairs.
- Fixed shortest path routing is used between all node pairs.

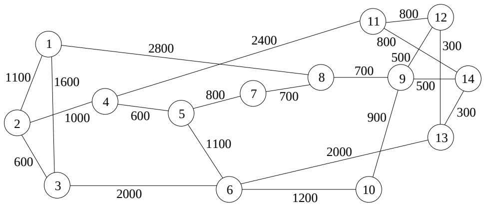

Fig. 7. Picture of NSFNET with 14 nodes.

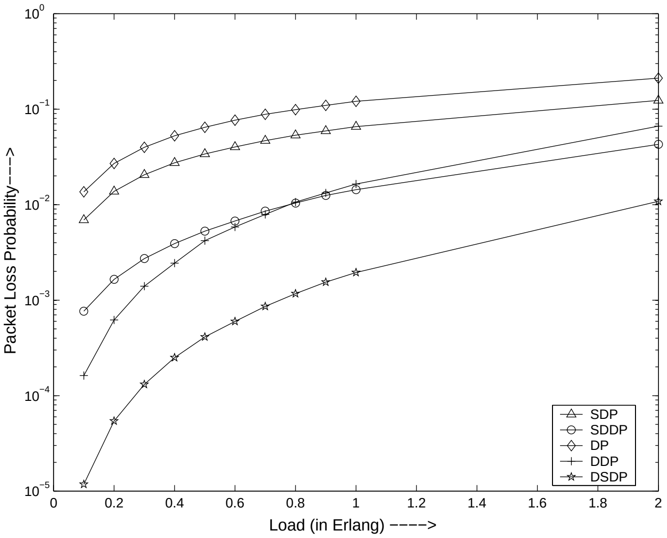

Fig. 8. Packet loss probability versus load for NSFNET at low loads with $\frac{1}{\mu} = 100$ μs and Poisson burst arrivals.

Figure 7 shows the 14-node NSFNET on which the simulations experiments were conducted. The distances shown are in km.

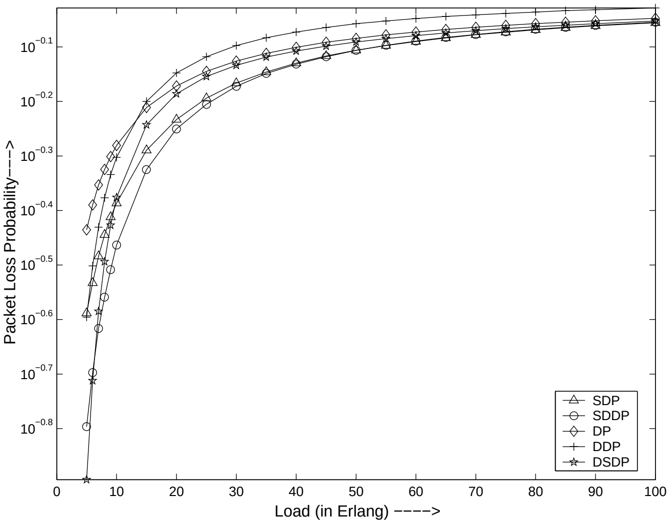

Fig. 9. Packet loss probability versus load for NSFNET at high loads with $\frac{1}{\mu} = 100$ μs and Poisson burst arrivals.

Figure 8 plots the total packet loss probability versus the load for the different contention resolution policies. An average burst length of 100 μs is assumed. We observe that SDP performs better than DP in all load conditions, and that the three policies with deflection, namely DSDP, SDDP, and DDP, perform better than the corresponding policies without deflection at low loads. DSDP performs better than SDDP and DDP at these loads; thus, at low loads, it is better to attempt deflection before segmentation. Also, at low loads DDP performs better than SDDP since there is no loss due to switching time in DDP. We see that policies with segmentation perform better than the policies without segmentation. A logical explanation would be that, in segmentation, on average only half of the packets from one of the bursts are lost when contention occurs. Also, at low loads, there is a greater amount of spare capacity, increasing the chance of successful deflection.

Figure 9 shows the packet-loss performance at very high loads. DSDP performs the best only at low loads. SDDP performs the best when the load is between 6 and 55 Erlang, after which SDP performs equally well, if not better. DDP performs well only at low loads, while at very high loads DP fares better than DDP. We observe that, at very high loads, policies without deflection perform better then the policies with deflection. At high loads, deflection may add to the load, increasing the probability of contention, and thereby increasing loss.

Figure 10 shows the average number of hops versus load for the different policies. In the deflection policies, the number of deflections increases as the load increases, resulting in higher average hop distance at low loads. As the load increases further, those bursts which are further from their destination will experience more contention than those bursts which are close to their destination. Thus, bursts with higher average hop count are less likely to reach their intended destination, and the average hop distance will decrease as load increases. Policies with

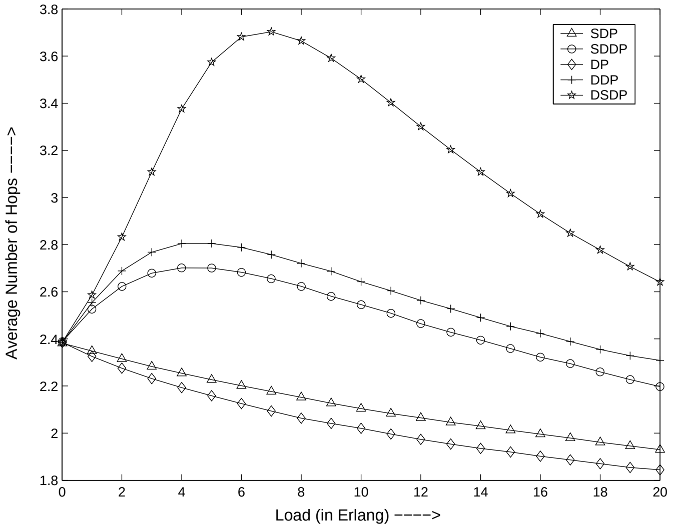

Fig. 10. Average number of hops versus load for NSFNET with $\frac{1}{\mu} = 100\ \mu$s

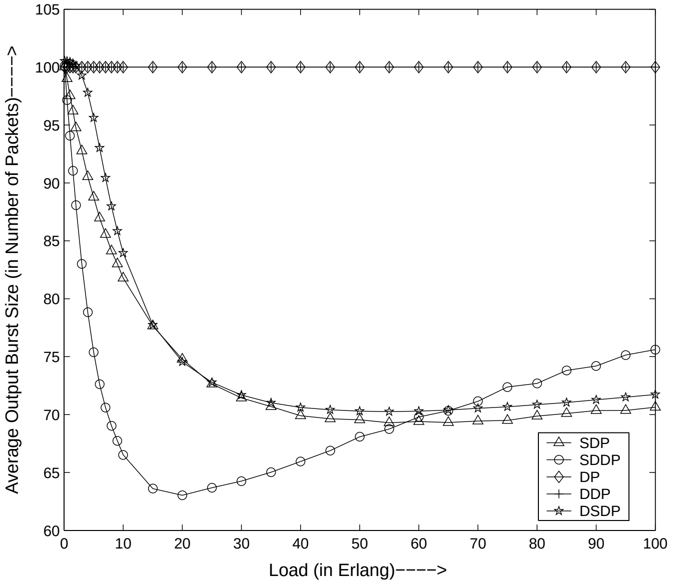

Fig. 11. Average output burst size versus load for NSFNET with $\frac{1}{\mu} = 100$ μs and Poisson burst arrivals.

segmentation have higher hop count compared to their corresponding policies without segmentation, since the probability of a burst reaching its destination is higher with segmentation.

Figure 11 shows the average output burst size versus load for the different policies. The output burst size is measured over both dropped and successfully received bursts. Initially, the burst size decreases with increasing load, as there are more segmentations with the increasing number of contentions. As the load increases further, the segmented bursts encounter more contentions, and because the segmented bursts have smaller size (lower priority), the segmented bursts tend to be dropped. The values for DP and DDP are constant for different values of load because the size of a burst is never altered.

The packet loss probability versus load for different values of switching time is shown in Fig. 12. As the switching time increases, the performance of SDDP decreases because a greater number of packets are lost during the re-configuration of the switch. On the other hand, DDP is not affected by the switching time and the loss remains almost constant. At low switching times, the results show that SDDP is better than the standard DDP, while at higher switching times, the standard DDP is better than the new SDDP because of the loss of packets during the switching time.

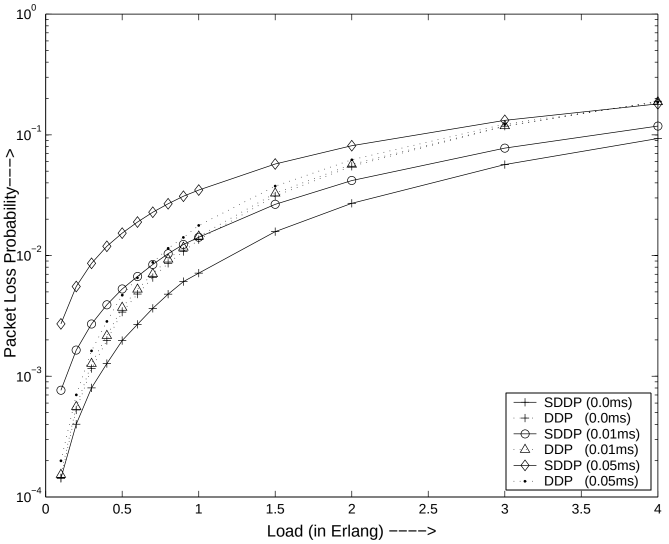

Fig. 12. Packet loss probability versus load at varying switching times for NSFNET with $\frac{1}{\mu} = 100\mu$s and Poisson burst arrivals.

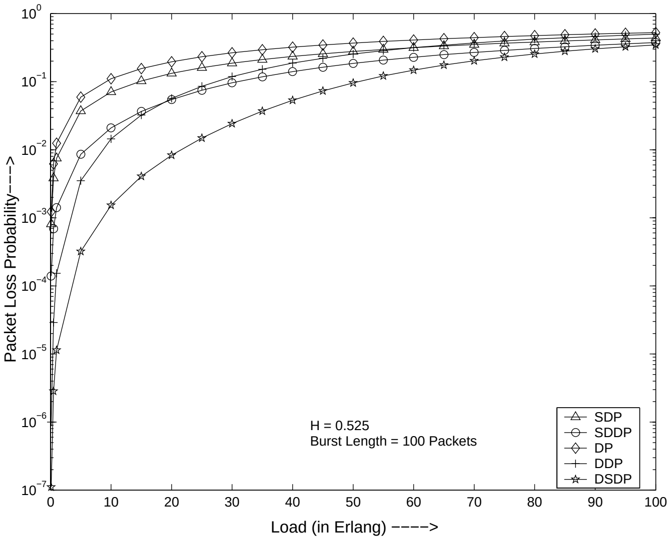

Fig. 13. Packet loss probability versus load for NSFNET with Pareto burst arrivals.

In order to capture the burstiness of data at the edge nodes, we also simulate Pareto burst arrivals with 100 independent traffic sources. The length of the burst is fixed to the average

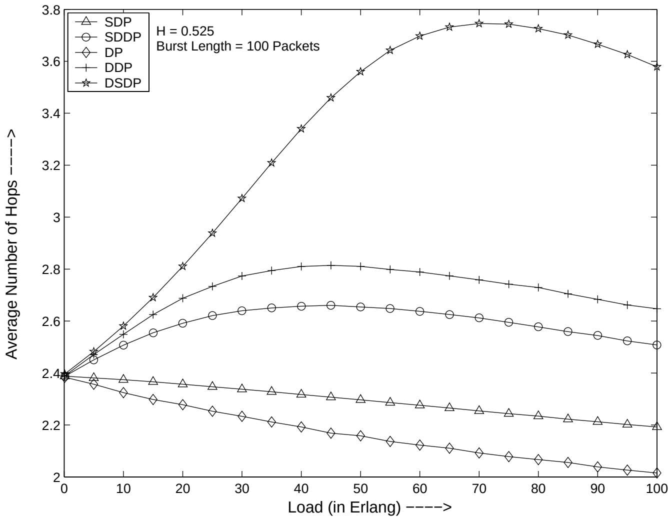

Fig. 14. Average number of hops versus load for NSFNET with Pareto burst arrivals.

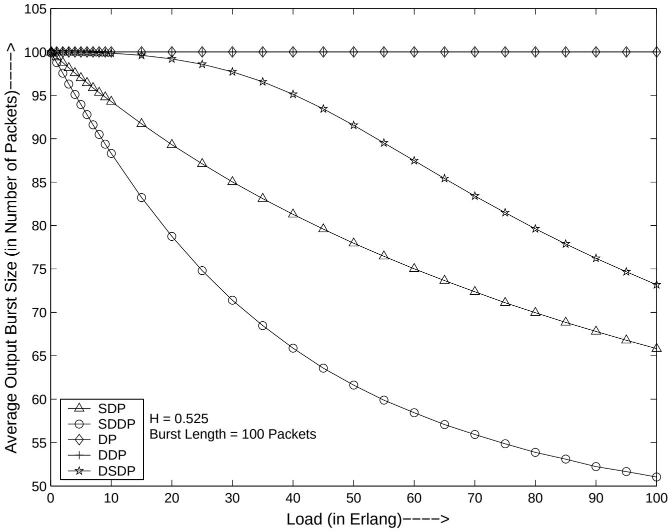

Fig. 15. Average output burst size versus load for NSFNET with Pareto burst arrivals.

burst length in the Poisson case, i.e., 100 fixed-sized packets. The Hurst parameter, H is set to 0.525. The remaining assumptions are the same. We plot the graphs for packet loss probability, average hop count, and output burst size versus load for Pareto inter-arrival time distribution and fixed-sized bursts.

Figure 13 plots the total packet loss probability versus the load with Pareto burst arrivals, for the different contention resolution policies. The results are similar to the Poisson case, except that DSDP is the best policy for the observed load range. We also observe that the policies with deflection perform better than the Poisson case due to the increased burstiness at the source. Deflection is a good option to avoid the contentions at the source.

Figure 14 shows the average number of hops versus load with Pareto burst arrivals for the policies. Figure 15 shows the average output burst size versus load with Pareto burst arrivals, for the different policies. The results are similar to the Poisson case.

V. CONCLUSION

In this paper, we investigated a number of different policies with and without segmentation and deflection. The segmentation policies perform better than the standard dropping policy, and offer the best performance at high loads. The policies which incorporate deflection tend to perform better at low loads.

In this paper, we considered only one alternate output port for deflection, an area for future work is the investigation of policies which consider multiple alternate output ports and in which the selection criteria is based on load and shortest path may also be considered. The segment dropping and deflection policies can also be implemented with priorities. Priorities would be based on a burst's tolerance for segmentation, deflection, and loss. To effectively evaluate the quality of service offered by various priority policies, a retransmission scheme for dropped packets could be implemented in order to measure end-to-end delay. A possible approach would be to implement a TCP layer on top of the optical burst switched layer. In such an implementation, it would also be useful to evaluate how TCP layer congestion control schemes react to and interact with various contention resolution schemes.

REFERENCES

[1] 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] I. Chlamtac, A. Fumagalli, L. G. Kazovsky, et al., "CORD: Contention Resolution by Delay Lines," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 1014-1029, June 1996.

[3] Z. Haas, "The 'Staggering Switch': An Electronically Controlled Optical Packet Switch," *IEEE Journal of Lightwave Technology*, vol. 11, no. 5/6, pp. 925-936, May/June 1993.

[4] A. S. Acampora and I. A. Shah, "Multihop Lightwave Networks: A Comparison of Store-and-Forward and Hot-Potato Routing," *IEEE Transactions on Communications*, vol. 40, no. 6, pp. 1082-1090, June 1992.

[5] A. Bononi, G. A. Castanon, and O. K. Tonguz, "Analysis of Hot-Potato Optical Networks with Wavelength Conversion," *IEEE Journal of Lightwave Technology*, vol. 17, no. 4, pp. 525-534, April 1999.

[6] F. Forghieri, A. Bononi, and P. R. Prucnal, "Analysis and Comparison of Hot-Potato and Single-Buffer Deflection Routing in Very High Bit Rate Optical Mesh Networks," *IEEE Transactions on Communications*, vol. 43, no. 1, pp. 88-98, Jan. 1995.

[7] B. Ramamurthy and B. Mukherjee, "Wavelength Conversion in WDM Networking," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 7, pp. 1061-1073, Sept. 1998.

[8] M. Yoo, C. Qiao, and S. Dixit,"QoS Performance of Optical Burst Switching in IP-Over-WDM Networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, no.10, pp. 2062-2071, Oct. 2000.

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

[10] R. Ramaswami and K. N. Sivarajan, " Optical Networks: A Practical Perspective," *Morgan Kaufmann Publishers, Inc.*, ch. 3, 126-160, 1998.

[11] S. Yao, B. Mukherjee, S.J.B. Yoo, and S. Dixit, "All-Optical Packet-Switched Networks: A Study of Contention Resolution Schemes in an Irregular Mesh Network with Variable-Sized Packets," *Proceedings, SPIE OptiComm 2000*, Dallas, TX, pp.235-246, Oct. 2000.

[12] V.M. Vokkarane and J.P. Jue, "Prioritized Routing and Burst Segmentation for QoS in Optical Burst-Switched Networks," *Proceedings, IEEE/OSA Optical Fiber Communication Conference 2002*, Anaheim, CA, March 2002.