Efficient Predictable Probe of Optical Burst Switched For Wireless Feeler Bond

Mr.Pradeep Kumar.A [1], Dr. Dinesh Senduraja Ph.D. [2]

Research Scholar, Department of Computer Science, Defence Institute of Advanced Technology, Pune-411 025.
Researcher, MED & COS Defence Research & Development Organization (DRDO) Pune- 411 021

Abstract:-
Optical Burst Switching (OBS) is a promising paradigm for high speed transmission of data. In OBS, a key problem is to schedule bursts with minimum loss. Single method is not sufficient to improve performance. So, our performance model includes some feasible methods to improve OBS performance without significantly increasing the implementation complexity. The methods are addition of simple fiber delay lines (FDLs), increasing random extra offset time, window based channel scheduling (WBS) and Burst Delay Feedback scheduling (BDFS). Additional FDLs can only eliminate the negative impact caused by the variation of the offset time between control packets and data bursts. The random extra offset time approach does not require any additional hardware in the nodes. WBS provides better throughput improvement when FDLs are used in the nodes to compensate the processing time. Finally Burst Delay Feedback Scheduling in addition with these methods can significantly improve OBS throughput and reduce transmission delay.

Keywords: Feedback Scheduling, Control overhead, offset time, Optical burst switching.

Introduction

The rapid growth of the optical internet will result in an increased demand for scheduling data to reduce loss rate. Moreover, for bursty traffic, Optical Burst Switching has better performance than the Optical Circuit Switching (OCS) and Optical Packet Switching (OPS). OBS can provide better transmission services in optical networks without sophisticated optical hardware [2–4]. In OBS data traffic discarded at immediate nodes is retransmitted by the sources. The one-way resource reservation is used in OBS that effectively reduces the hardware complexity and is not sensitive to the propagation delay between nodes. The OBS node must also complete the routing and channel assignment computations of the transit data bursts before their arrivals. In general, only a first-come-first-served (FCFS) approach can be used to schedule the incoming data bursts. Data bursts in OBS networks with large hop number paths will suffer from a larger loss rate. Many approaches have been proposed to improve

The performance of OBS, for example, adding optical
Buffers (switchable fiber delay lines) to OBS nodes [8], burst segmentation, centralized control and two-way resource reservation, and dynamic routing [9]. However, many of these proposals are not practical because they inevitably require much more sophisticated implementation than the original OBS scheme. In FCFS, System throughput is reduced because more transmission bandwidth is required to retransmit the data bursts with large hop number paths. In order to improve OBS performance without significantly increasing the implementation complexity, it is necessary to delineate the relationship among control processing time, one-way resource reservation, and OBS performance such that the merits of different improvement approaches can be fully understood. With this understanding, different performance improvement methods canbe combined to further improve the OBS performance.We study combining different methods to improve OBS throughput and delay performance. These schemes are practical to

implement. Our main contributions include the following:

- Commonly used discarded-traffic clear approach in OBS performance evaluations can overlook problems that occur only in a few paths of a network and lead to incorrect conclusions. Thus all performance evaluations use discarded-traffic-retransmit approach.
- The compensation factor β in the fiber delay line (FDL) overcompensation approach is critical to OBS performance improvement. But, we use FDLs only for Tcp compensation.
- Burst Delay Feedback Scheduling is a new approach that can improve system performance than random extra offset time even if the OBS has zero Tcp or Tcp is fully compensated by FDLs.
- Window-based channel scheduling (WBS) that is suitable for OBS with both window time Twd and Tcp compensated by FDLs. Proposed BDFS can provide better performance improvement in addition with FDL compensation and WBS, when compared with WBS and FDL compensation alone.

- By combining burst scheduling with those three methods we can obtain further increased OBS throughput performance. By using these methods we can also reduce delay and average loss rate.

Factors affecting Throughput degradation in Optical Burst switching

Optical Burst switching is the high speed switching technique that takes advantage of both Circuit and Packet switching. In OBS, the data bursts consisting of multiple data packets. When data packets arrive at an OBS node, data bursts are generated to carry the data packets to their destinations. For each newly generated data burst, a control packet is first sent to the destination of the data burst. The control packet reserves the resources at the intermediate nodes on the path of the data burst. No acknowledgment is sent back in order to minimize the delay time of sending out the data burst at the source. After an offset time, the source node sends out the data burst following the same routing path of the control packet. The minimum offset time between the control packet and the data burst is

$$Toff = H \times Tcp + Tsw, (1)$$

whereTsw is the required switch reconfiguration time at each node, Tcp is the processing time of a control packet in a node, H is the number of hops to the destination from the current location of the control packet. Hence, H is equal to the total hop count of the path when the control packet is at the source and decreases by one for each intermediate node the control packet passes.

One factor affecting performance of OBS is discarded-traffic clear approach. A data burst will be discarded if it cannot find an appropriate output channel when it arrives at an intermediate node. In this approach that discarded data burst cannot be retransmitted. This will consistently degrade throughput. Such problem can be avoided by using the discarded-traffic-retransmit approach.

Traditional OBS use First Come First Served approach to schedule the incoming data burst. This follows first in first out method. As each process becomes ready, it joins the ready queue. When the current running process ceases to execute, the oldest process in the Ready queue is selected for running. This increases delay that will degrade the performance.

Another factor is control overhead with unacceptable processing time (Tcp). Throughput mainly depends on control packet processing time. If Tcp increases, throughput decreases. Optical buffer is the only choice to introduce delay in between control packet and burst in order to compensate control packet processing time. Just Enough Time (JET) is one of the reservation schemes used in OBS. Here, the size of the data burst is decided before the control packet is transmitted by the source. If bandwidth is available, the control packet reserves channel for a fixed duration of time. The reservation is made from the time when the first bit of payload reaches the node till the last bit of payload is transmitted to the output port. Since, there is no wastage of bandwidth in this scheme, it removes idle bandwidth time. So we use JET scheme for all our performance evaluations.

Methods to improve the OBS Throughput and Delay performance

In spite of the importance of higher throughput and lower delay, a single method is usually insufficient to provide the required throughput improvement. By combining methods, it may be possible to improve the OBS throughput and

delay performance without significantly increasing the OBS implementation complexity. The candidates of performance improvement methods include adding FDLs; random extra offset time, window-based channel scheduling and our proposed Burst delay feedback scheduling approach. We first investigate the nature of each method and how they improve the performance.

Adding Fiber Delay Lines

We know that data bursts with different values of H will have the same probability to block each other if the offset time Toff is a constant. Thus we should install FDLs at the node inputs to delay the incoming data bursts' Tcp time. However, only a single simple FDL (as shown in Fig. 1 is required per node input to compensate the Tcp of all incoming data bursts in all wavelength channels. It may not be easy to use FDLs to exactly compensate the control packet processing time Tcp because of control packet processing time that can vary with the system loading and nodes. We observe that one solution is to set the delay time TFDL of FDLs to the maximum of Tcp and delay the forwarding of the control packet to the next node, if necessary, to keep the offset time Toff to be a constant.

When the offset time Toff is constant, data bursts with different path lengths will have the same channel reservation success probability at an intermediate node. Therefore, data bursts with larger hop count paths will suffer from larger loss rate. We observe that the solution requiring minimum extra effort is to overcompensate the control packet processing time by setting the length of the FDLs to slightly larger than that required for the compensation of Tcp. We define TFDL = Tcp + β, where $\beta \geq$ −Tcpis the compensation factor.

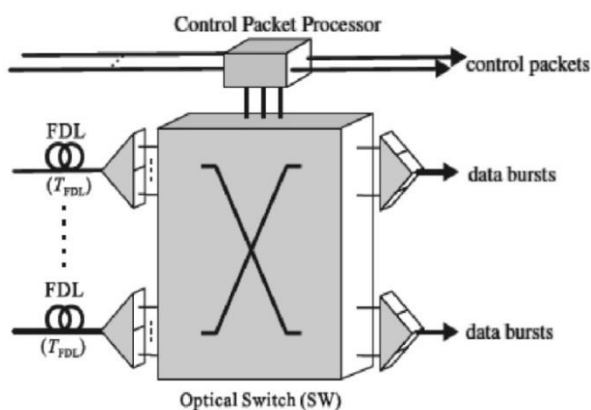
Fig.1. An OBS node with simple FDLs installed at each

input port for control packet processing time compensation

Overcompensation of Tcp will increase the offset time between the control packet and the data burst when they pass the nodes along the path. Data bursts that have passed more nodes will therefore have a greater chance to reserve an output channel at an intermediate node because of the larger offset time. From implementation consideration, it may be better to first use FDLs to compensate the control packet processing time Tcp and then use other methods to further improve the throughput performance.

Random Extra Offset Time

Throughput improvement has been observed with an extra random offset time and this is attributed to the traffic shaping effect of the data bursts at OBS source nodes. However, we find that the random extra offset time also significantly weakens the connection between the number of hops to destination H and the offset time, and hence reduces blocking. Random extra offset time can reduce the loss rate of data bursts with large hop count paths and improve the throughput performance.

We first consider the cases of OBS without control packet processing time compensation. When a random extra offset time is added to Eq. (1), the offset time becomes

$$\text{Toff} = H \times \text{Tcp} + \text{Tsw} + \text{Tex}, \qquad (2)$$

where Tex is the random extra offset time. Without the random extra offset time, all data bursts have the same channel reservation success probability s at any node, e.g., s=0.5. After an offset time, Tex has been added/subtracted to/from each pair of control packet and data burst.

Similar to the case of FDL overcompensation, larger values of random extra offset time Tex do not guarantee increased system throughput. However, JET OBS can have throughput improvement with a large range of Tex and the selection of a suitable value of Tex becomes easy.

Window-Based Channel Scheduling

Window-based channel scheduling schemes delay the channel/routing assignment an additional Twd time after reading the information of a control packet. So WBS approach makes

better channel/routing assignment decisions than the FCFS approach. To illustrate, Fig. 2 shows four control packets and their associated data bursts arriving at a node. Assuming that all data bursts are routed to the same output port Ox of the node, we may need three output channels if the channel assignment uses FCFS according to the arrival of control packets, e.g., DB1 → Ox,1, DB2 → Ox,1, DB3 → Ox,2, and DB4 → Ox,3, where Ox,y is the y th channel of output port Ox. With the additional Twd delay time; however, we need to use only two for the channel assignment, e.g., DB1 → Ox,1, DB2 → Ox,2, DB3 → Ox,1, and DB4 → Ox,2. In WBS, additional Twd time will increase the equivalent control packet processing time to Tcp + Twd.
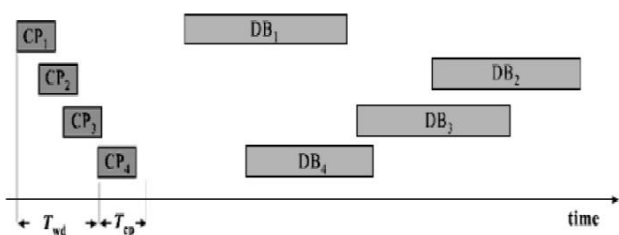

Fig. 2. Arrival of control packets CP1 to CP4 and their corresponding data bursts DB1 to DB4.

We assume that Tcp is much larger than L or the Twd delay time is compensated [1]. WBS OBS assuming large Tcp does not require any additional hardware and only needs to extend the initial offset time at the source. When window time is added to Eq. (1), the offset time becomes

$$T_{off} = H \times (T_{cp} + T_{wd}) + T_{sw} \quad (3)$$

Since the value of Tcp should not be restricted, delay compensation seems to be a more attractive approach. At the moment, adding FDLs at the node inputs is the only practical way to compensate the delay of the control packet in a node.

Next, it is necessary to determine the procedure for assigning the output channel to a control packet CPx, after the Twd time delay. A common approach is to virtually assign output channels to CPx and other control packets that have arrived in the Twd time period according to the arrival sequence of their associated data bursts. The data burst DBx of CPx will get the channel that is assigned to DBx in the virtual channel assignment. DBx will be rejected if it fails to get

a channel in the virtual channel assignment. This approach is effective, e.g., we will need only two output channels in Fig. 2. In such a situation, the data burst arrival sequence is the same as that of the control packets.

We need to assign the output channel based on the impact of the control packet on other control packets (their associated data bursts) arriving in the Twd delay time interval. Consequently, we weigh the data burst DBk of a control packet CPk with a value wk. To assign a channel to a control packet CPx (data burst DBx), we first compute two control packet sets S and R, that arrive in the Twd delay time interval and their associated data bursts will be accepted if DBx has (has not) been assigned a channel. It can assume any channel reservation scheme that is applicable to WBS OBS.

We have tested the wx setting of (1) a nonzero constant c; (2) the data burst length Lx; (3) the inverse of the number of hops to the destination, 1/Hx; (4) the number of passed hops from source hx; (5) the traveling distance from the source dx; and (6) combinations of these parameters. Finally, we chose wx = Lx (1 + Hx), to increase the priority of data bursts with large hop count. This simple wx setting provides slightly better system throughput performance than that from the common approach of channel assignment when WBS OBS is without FDL compensation. But this Scheduling do not guarantee increased system throughput during traffic.

Proposed Burst Delay Feedback
Scheduling Algorithm

Our aim is to schedule as many bursts as possible with reduced burst loss. Therefore we propose a new Burst scheduling algorithm named Burst Delay Feedback Scheduler for scheduling the burst with minimum loss. In a feedback-based network, the ingress nodes have knowledge of the network state and they can respond to changes in the network load. This mechanism support quality of service (QoS) for different class of bursts. In feedback scheduling the core node senses the data traffic and sends feedback to the previous core/edge node for delay the incoming burst in order to minimize contention. This feedback contains information that how much time the data burst must delayed to reduce contention. Performance parameters for each burst flow are exchanged by a feedback message to the ingress nodes. According to the

information contained in the message, the edge nodes dynamically adjust its parameters needed to achieve a defined QoS parameter such as bandwidth, throughput and delay. The adjusted parameters are the offset time parameter or the basification rate. Feedback control approach computes accurate burstification rate (i.e., rate by which the bursts are injected into the network) for each class of bursts. Based on the computed burstification rates, the maximum delay is calculated and guaranteed to the deterministic level.
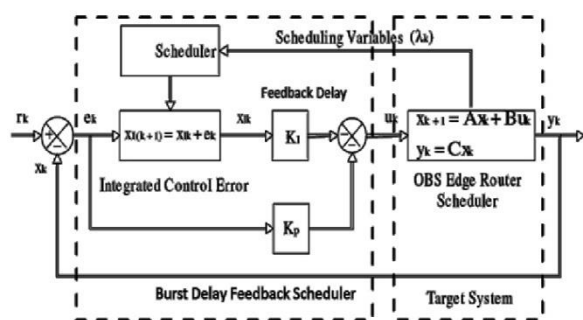

Fig.3.A linear control system with feedback control mechanism using burst delay feedback scheduler.

General diagram for a linear control system implementing closed loop feedback control mechanism is shown in Fig. 3. At first edge node generates bursts by aggregating a number of IP packets directed towards the same core node. The burst manager controller (BMC) controls the Basification rate which resides at every edge node of the network. Every core node sends a feedback to the edge nodes containing a reduction request of the basification rate. The reference Delay is a reference value that the controlled output parameter should be restrained in the network. The error is the difference between the reference Delay and the measured Delay. The burst manager controller (BMC) takes the error value as an input and generates a basification rate accordingly based on a control law. This approach guarantees quality of service in terms of throughput and latency for each class of burst.

Performance Evaluation
We use simulations to verify the throughput and transmission delay using the combinations of the methods discussed in Section III and IV on a 22-node 23-link NS2 topology network. We assume that all links are bidirectional. In the simulations, we assume that data bursts arriving at the nodes follow the Poisson process. When a new data

burst arrives at a node, it randomly chooses a destination from the rest of the nodes in the network and uses shortest-path routing to determine the path. The maximum number of paths per link for the NS2 is 23. Therefore, the maximum throughput per node is 13/23 or around 0.565. This value is our maximum achievable throughput.

Throughput Performance of BDFS with WBS and FDL compensation
In the existing Window Based Channel Scheduling with FDLs for control packet processing time overcompensation, throughput performance was improved. To further improve the throughput and reduce delay, we may combine WBS plus FDL overcompensation with Burst Delay Feedback Scheduling.
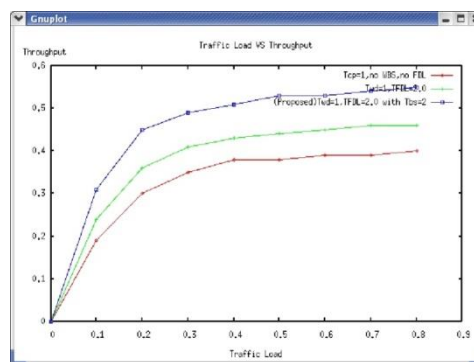

Fig.6. the Throughput curves on NS2 with Tcp = 1 using WBS and FDL compensation with Burst Delay Feedback Scheduling.

Figure 6 is the traffic load-throughput performance of OBS with WBS of Twd = 1 plus FDL compensation of

Delay Performance of Burst Scheduling with WBS and FDL compensation
By combining BDFS and WBS with FDL compensation, reduced transmission delay was also achieved.
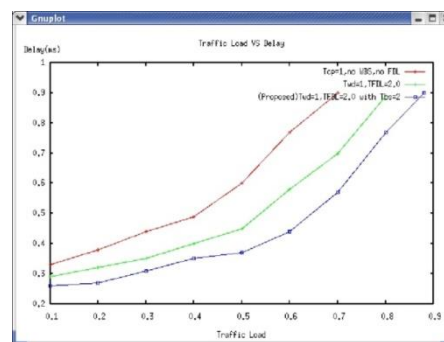

Fig.7. the Transmission Delay curves on NS2 with Tcp = 1 using WBS and FDL compensation with BDFS.

Figure 7 is the traffic load-throughput performance of OBS with WBS of Twd = 1 plus FDL compensation of TFDL = 2.0 with and without proposed burst scheduling, when Tcp = 1. From the simulation results, the Delay reached nearly 1ms only when full load condition, in the Burst scheduling with WBS OBS plus FDL compensation (blue line) method. But, in the WBS OBS plus FDL compensation without BS (green line) method delay reached 1ms for 0.9 traffic load itself. So the proposed method can decrease the OBS delay.

Measurement of Average Loss Rate and Burst Size using Burst Feedback Scheduling:
Figure 8 shows, average loss rate performance of OBS by using burst delay feedback scheduling. When delay time is varied from 0.2 to 1, the average loss rate of the Burst scheduling is varied. When compared to window based channel scheduling with FDL, the performance of burst scheduling was improved. From the figure average loss rate vary from 0.11 to 0.13. Similarly figure 9 shows burst size of OBS for various simulation time.

Conclusion
Optical burst switching system throughput suffers from the traditional discarded-traffic clear approach. But the new discarded-traffic-retransmit approach was used to increase system performance. By combining methods such as adding simple FDLs, random extra offset time and Window Based Channel Scheduling with Burst Delay Feedback Scheduling, performance improvement of OBS can be achieved. All performance evaluations can be done using Just Enough Time (JET) protocol.

Adding a single FDL to an input port of an OBS node is used to compensate the Tcp of incoming data bursts by adjusting FDL delay time TFDL. Burst Scheduling with the FDL can have better throughput improvement than random extra offset time approach with added offset time Tex.

The addition of BDFS with WBS (window time Twd) and FDL (delay time TFDL) can have better throughput and delay performance. From the simulation results, the throughput is increased.

References:

[1] C.Y. Li, P.K.A. Wai, and V.O.K. Li, ―Performance Improvement Methods for Burst-Switched Networks,‖ *J.Opt. Commun. Netw.*, Vol. 3, pp. 104–116, 2011.

[2] C. Qiao and M. Yoo, ―Optical burst switching (OBS)—a new paradigm for an optical Internet,‖ *J. High Speed Netw.*, vol. 8, pp. 69–84, 1999.

[3] Y. Xiong, M. Vandenhoute, and H.C. Cankaya, ―Control architecture in optical burst-switched WDM networks,‖

[4] *IEEE J. Sel. Areas Commun.*, vol. 18, pp. 1838–1851,2000.

[5] C.Y. Li, G.M. Li, P.K.A. Wai, and V.O.K. Li, ―Optical burst switching with large switching overhead,‖ *IEEE J. Lightwave Technol.*, vol. 25, pp. 451–462, 2007.

[6] B.C. Kim, Y.Z. Cho, and D. Montgomery, ―An efficient optical burst switching technique for multi-hop networks,‖ *IEICE Trans. Commun.*, vol. E87-B, pp. 1737–1740, 2004.

[7] M. Yoo, C. Qiao, and S. Dixit, ―QoS performance of optical burst switching in IP over WDM networks,‖ *IEEE Trans. Commun.*, vol. 18, pp. 2062–2071, 2000.

[8] J. Li, C. Qiao, J. Xu, and D. Xu, ―Maximizing throughput for optical burst switching networks,‖ *IEEE/ACM Trans. Netw.*, vol. 15, pp. 1163–1176, 2007

[9] X. Lu and B.L. Mark, ―Performance modeling of optical-burst switching with fiber delay lines,‖ *IEEE Trans. Commun.*, vol. 52, pp. 2175–2183, 2004.

[10] J. Pedro, P. Monteiro, and J. Pires, ―Traffic engineering in the wavelength domain for optical burst switched networks,‖ *J. Lightwave Technol.*, vol. 27, pp.3075–3091,2