

Novel Congestion Control Scheme in Next-Generation Optical Networks*

LaeYoung Kim, SuKyoung Lee, and JooSeok Song

Dept. of Computer Science, Yonsei University, Seoul, Korea
{leon, sklee, jssong}@cs.yonsei.ac.kr

Abstract. In this paper, to improve the burst loss performance, we actively avoid contentions by proposing a novel congestion control scheme that operates based on the highest (called peak load) of the loads of all links over the path between each pair of ingress and egress nodes in an Optical Burst Switching (OBS) network.

1 Introduction

The most important design goal in Optical Burst Switching (OBS) [1] networks is to reduce burst loss. The authors of [2] and [3] propose congestion control schemes pointing out that without controlling the offered load, burst loss rate will eventually go up to a very large number. In [2], a TCP congestion control scheme for OBS networks is studied. In this scheme, as in a normal TCP connection, the ingress node determines its burst sending rate based on the received ACK packet from the egress node. In [3], each ingress node adjusts its burst sending rate continually according to the information about burst loss received from all core nodes. However, these congestion control schemes have a problem that they may be triggered inappropriately because most of burst loss randomly occurs due to the bufferless nature of OBS networks although the network is not congested. In this paper, we propose a novel congestion control scheme that operates based on the highest (called peak load) of the loads of all links over the path between each pair of ingress and egress nodes (referred as flow) in an OBS network.

2 Congestion Control Scheme Based on Peak Load

Note that in OBS networks, when only one outgoing link over the path for a burst is congested even though the other outgoing links over the path are not congested, the burst will be dropped with a high probability in an OBS network without any contention resolution scheme. Thus, we propose a congestion control scheme based on peak load over the path for each flow.

In the proposed scheme, each egress node sends a LOAD message containing *Peak-Load* field set to 0 to all reachable ingress nodes over the backward path

* This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment).

every T_L units of time. The LOAD message is sent on a control channel like a control packet. We assume that T_L for each flow is in proportion to the total number of hops of the flow. When the core node receives the LOAD message which is originated by egress node, E destined to ingress node, I from the link (k, j) , it compares the value of *Peak-Load* field in the received LOAD message and calculated load on link (j, k) for the flow (I, E) . If the calculated load is greater than the value of *Peak-Load* field, the core node copies its calculated load to the *Peak-Load* field in the LOAD message and forwards the message to the next node towards the destination of the message, I . Otherwise, the core node forwards the LOAD message to the next node without any change. In this way, each ingress node can receive the LOAD message including the highest load on the path to each reachable egress node. Each core node, j maintains the following information to calculate the load of outgoing link (j, k) for each flow which shares the link:

$T_{(I,E)}$: Time interval between two consecutive LOAD messages for flow (I, E)
$B_{(I,E)}$: Duration of all incoming bursts which want to use the outgoing link (j, k) during the interval, $T_{(I,E)}$

Whenever the core node receives the LOAD message for the flow (I, E) , it calculates the load on the link (j, k) for the flow (I, E) as $\frac{B_{(I,E)}}{T_{(I,E)}}$. Each ingress node, I maintains the following information for every flow to egress node, E :

$R_{(I,E)}$: Current burst sending rate for flow (I, E)
$NAK_{(I,E)}$: Number of NAK messages received from the congested node (i.e., core node where a burst is dropped) on path (I, E) during the interval between two consecutive LOAD messages

Table 1 shows the procedure of congestion controller in our scheme. As shown in this table, if ingress node, I receives a LOAD message originated by egress node, E , congestion controller at the ingress node compares the value of *Peak-Load* field in the LOAD message with $LOAD_{TH}$ which is a threshold for tolerable level of congestion at high load on each link. If the *Peak-Load* is greater than $LOAD_{TH}$, the congestion controller decreases the burst sending rate. Otherwise, the congestion controller checks whether the value of $NAK_{(I,E)}$ is zero or not. If $NAK_{(I,E)}$ is zero, the congestion controller increases the burst sending rate. Finally, $NAK_{(I,E)}$ is set to 0 as line 6 in Table 1. The proposed congestion control scheme adopts the well-known Additive Increase/Multiplicative Decrease (AIMD) algorithm and the detailed algorithm for burst sending rate adjustment is as follows.

Table 1. The procedure of congestion controller at ingress node, I

1	If (Receive LOAD message from egress node, E) {
2	/* If <i>Peak-Load</i> in the LOAD message is greater than $LOAD_{TH}$ */
3	If ($Peak-Load > LOAD_{TH}$) decrease the burst sending rate;
4	Else /* Check whether $NAK_{(I,E)}$ is zero or not */
5	If ($NAK_{(I,E)} == 0$) increase the burst sending rate;
6	$NAK_{(I,E)} = 0$; /* Reset $NAK_{(I,E)}$ */ }

$$\text{Increase : } R_{(I,E)} \leftarrow \min(R_{(I,E)} + (\alpha \times H), MAX_{RATE}) \quad (1)$$

where α is the increase constant whose unit is a burst and H is the total number of hops between the ingress node, I and the egress node, E . We assume that each ingress node can send bursts up to MAX_{RATE} to achieve fairness among all the flows.

$$\text{Decrease : } R_{(I,E)} \leftarrow R_{(I,E)} \times (1 - \beta) \quad (2)$$

where β is the decrease constant. As shown in Table 1, if the peak load is greater than $LOAD_{TH}$, the burst sending rate will be proactively decreased to avoid contentions regardless of burst loss. Although the ingress node has received one or more NAKs during the interval between two consecutive LOAD messages, the burst sending rate will not be decreased if the peak load is not greater than $LOAD_{TH}$. As a result, our scheme tries to avoid an unnecessary decrease of the sending rate, that has been a problem in the existing schemes [2, 3] because they decrease the sending rate only according to burst loss. If the peak load is not greater than $LOAD_{TH}$ and the ingress node has not experienced any burst loss on the path (I, E) during the interval between two consecutive LOAD messages, the burst sending rate will be increased by an amount proportional to the total number of hops on the path as in Eq. 1. Because T_L for each flow is in proportion to the total number of hops of the flow as mentioned above, the congestion controller for a flow with larger hop count is activated less frequently than that for flows with smaller hop count. Thus, for a flow with larger hop count, the congestion controller increases its sending rate more than for flows with smaller hop count to achieve fairness among all the flows.

3 Performance Evaluation

In this section, we evaluate the performance of the proposed Peak Load-based Congestion Control scheme (PL-CC) comparing with existing OBS protocol using TCP Congestion Control (TCP-CC) [2] and conventional OBS without Congestion Control (no-CC). The simulations are performed using the ns-2 simulator. In our simulation, we use the 14-node NSFNet with 21 links where the link capacity is 10 Gbps and the number of flows is 72. Bursts are generated from all the ingress nodes. It is assumed that burst arrivals follow the Poisson process and their lengths are negative exponentially distributed with mean 1 Mbits. We set T_L to $30 \times H$ msec for each flow. T_L is about three times the average Round Trip Time (RTT) of each flow since the average delay per link in the simulation topology is about 5 msec. We set α to 3 (bursts) while setting β to 0.2. From the simulation results obtained by varying the $LOAD_{TH}$ and the MAX_{RATE} , we have learned that the proper values for $LOAD_{TH}$ and for MAX_{RATE} are 0.05 and 150 Mbps, respectively, for our simulation topology.

Fig. 1 (a) plots the burst loss rate versus traffic load for no-CC, TCP-CC and the proposed PL-CC. For TCP-CC, the increase constant is set to 1 while the decrease constant is set to 0.5 as in TCP. We see that the burst loss rate of PL-CC is much lower than TCP-CC as well as no-CC at all loads. From the

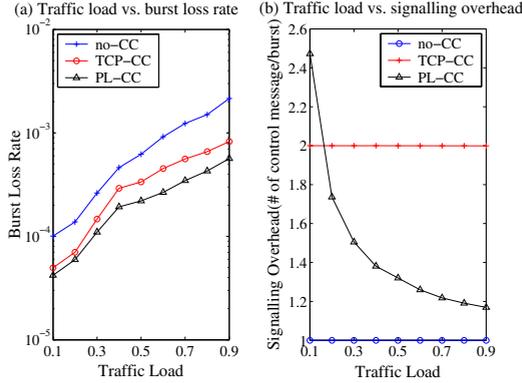


Fig. 1. (a) Traffic load vs. burst loss rate (b) Traffic load vs. signalling overhead

results, the burst loss performance improvement of the proposed PL-CC over TCP-CC ranges from 15.36% at the load of 0.1 to 31.46% at the load of 0.9.

To investigate signalling overhead due to congestion control for our scheme, we run simulation tests for all the three schemes. Fig. 1 (b) plots the count of control messages per burst versus traffic load. At first, we see that for no-CC, a control message per burst is sent regardless of the traffic load because control message contains only control packet for no-CC. For TCP-CC, count of control message per burst reaches nearly 2.0 regardless of the traffic load because whenever the egress node receives a burst, it sends an ACK to the ingress node. For PL-CC, control message contains LOAD and NAK messages as well as control packet. In this scheme, the LOAD message is periodically generated. Thus, we observe from Fig. 1 (b) that for our scheme, the count of control message per burst decreases significantly as the traffic load increases. We also see that for our scheme, additional control messages is generated significantly less compared to TCP-CC at most ranges of the traffic load.

Finally, we investigate throughput fairness for our scheme by using the well-known Jain fairness index. For our scheme, the Jain fairness index decreases as the traffic load increases. Specifically, the Jain fairness index is about 0.998 and 0.956 when the traffic load is 0.1 and 0.9, respectively. However, these results prove that our scheme can maintain acceptable throughput fairness among all the flows at all loads.

4 Conclusions

Simulation results indicated that the proposed congestion control scheme significantly reduces the burst loss rate while maintaining a reasonable fairness in OBS networks. Simulation results also showed that our scheme can balance between the performance gain in terms of the burst loss rate and the operation cost in terms of the signalling overhead.

References

1. Qiao, C., Yoo, M.: Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet, *Journal of High Speed Networks*, Vol.8, No.1 (January 1999) 69–84.
2. Wang, S.Y.: Using TCP Congestion Control to Improve the Performances of Optical Burst Switched Networks, *IEEE ICC '03*, Vol.2 (May 2003) 1438–1442.
3. Maach, A., Bochman, G.V., Mouftah, H.: Congestion Control and Contention Elimination in Optical Burst Switching, *Telecommunication Systems*, Vol.27, Issue 2–4 (October–December 2004) 115–131.