Multi-layer Analysis to Quantify the Impact of Optical Burst Reordering on TCP Performance

Sebastian Gunreben
Institute of Communication Networks and Computer Engineering - University of Stuttgart
Pfaffenwaldring 47, 70569 Stuttgart, Germany
Tel: (+49) 711 685 67968, Fax: (+49) 711 685 57968,
E-mail: gunreben@ikr.uni-stuttgart.de

ABSTRACT
In this paper we provide a new methodology to quantify the impact of optical burst reordering in OBS networks on the TCP/IP end-to-end performance. We assume a multi-layer network stack consisting on TCP/IP on top of OBS. We propose a multi-layer analysis of burst and packet reordering that is realized by simulation. We apply the packet reordering definition of the IETF IPPM working group and use two different reorder metrics for precise characterization of reordering. We identify the key parameters in a burst and packet reordering scenario and point out TCP critical scenarios. We found that OBS only has an impact on the TCP performance in very dedicated scenarios.
Keywords: TCP over OBS, multi-layer analysis, simulation

1. INTRODUCTION

As TCP is the dominant transport layer protocol in today's networks, it is important to analyze the TCP performance on top of new technologies, e.g. OBS. In OBS networks, contention resolution schemes like wavelength conversion, buffering and deflection routing are needed. Without additional mechanisms buffering and deflection routing cannot guarantee burst in-order delivery, if the time in the buffer or on the deflection route exceeds the burst inter-arrival time. Thus burst reordering causes reordering of IP packets.
The basic TCP protocol [5] suffers from this packet reordering as it may interpret a missing packet as lost independent of its probable late arrival. The critical mechanism in TCP is the fast retransmit algorithm. It is invoked, if the duplicate acknowledgment (dup-ack) threshold at the sender is reached. As a result the missing packet is retransmitted and additionally the sender halves its congestion window, which decreases TCP sending capability.
The impact of OBS on TCP, mostly the OBS burst loss property, has been studied extensively in the literature. Most studies, like [1], [2] or [3] investigated an integrated TCP over OBS scenario, i.e., network layers between TCP and OBS layer are transparent to their investigations. In general, it is hard to identify the direct relationship between OBS network parameters and TCP throughput performance. This paper fills out this missing investigation on burst reordering and its impact on the TCP end-to-end throughput performance.
We model burst reordering in a generic burst deflection scenario, where we predefine the burst reordering and conclude on the packet reordering. In our scenario only the payload data is reordered while the sequence of acknowledgements is kept in order. With these results we are able to estimate also the TCP performance. We do a layer separated analysis taking each layer into account starting at the optical burst layer. The paper structure is as follows: In section 2 we introduce IETF's reordering metrics. In section 3 we describe the simulation scenario and identify the relevant parameters. In section 4 we show our results on the impact of burst reordering a common OBS scenario. The concluding section 5 summaries our work.

2. REORDERING METRICS

In this section we introduce the IP packet reordering metrics of the IETF working group IPPM [4]. The reordering metrics also hold for generic packet-switched networks like OBS networks. In the following, we use the term packet to be in line with the metrics definition.
Definition of Reordering: At the source node each packet is assigned a unique sequence number seq. The destination node maintains the counter $NextExp$, which denotes the sequence number of the next expected packet. The previously received packet determines the value of $NextExp$ (the first packet is in-order by definition). An arriving packet with sequence number seq is reordered, if $seq < NextExp$. In this case the value of $NextExp$ does not change. On the other hand, if $seq \geq NextExp$ the packet is considered to be in order and $NextExp$ is set to $seq + 1$.

The work reported in this paper was supported in part by the NoE e-Photon ONe+ FP6-IST-027497 and in part by the IP NOBEL Phase 2 FP6-IST-027305.

i	1	2	3	4	5	6	7
s[i]	1	2	6	7	3	4	5
NextExp	-	2	3	7	7	7	7
n-reord	-	-	-	-	2	0	0
e	-	-	-	-	2	3	4

Table 1. Reordering metrics

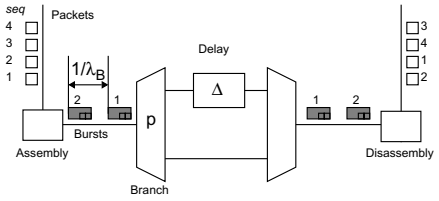

Figure 1. Simulation model

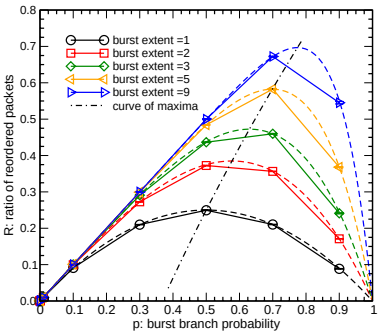

Figure 2. Reordering Ratio

Metric Reordering Ratio: A reordering ratio larger than null indicates reordering in the network. The Reordering Ratio metric quantifies the ratio of reordered packets and is defined by R = (Number of reordered packets) / (Total number of received packets).

Metric Reordering Extent: The reordering extent quantifies the minimal buffer size, which is needed to restore packet order at the destination. It measures the maximum distance of a reordered packet i to the previously arrived packet j with the smallest value of j satisfying $s[j] > s[i]$. The more formal definition of the extent is $e = i - \min_j(s[j] > s[i] \wedge j < i)$, where j and i denote the arrival position of two packets at the destination and $s[i]$ and $s[j]$ denote their sequence number. In Table 1 an example is depicted. The packet $i = 5$ with sequence number $s[i] = 3$ is reordered. The smallest value of $s[j]$, which is larger than $s[i]$ arrives at $j = 3$ with $s[j] = 6$. The extent is the difference of both arrival positions $e = 5 - 3 = 2$.

TCP-relevant Metric: The TCP-relevant metric quantifies the violation of the dup-ack threshold and allows TCP performance estimation. The TCP-relevant metric returns the ratio of n-reordered packets. An n-reordered packet is a reordered packet, which causes n dup-ack packets to TCP. The formal definition: A packet i is n-reordered if $s[j] > s[i]$ $\forall j$ where $i - n \le j < i$. The degree of n-reordering is measured as $m(n)/L$, where $m(n)$ is the number of n-reordered packets and L is the number of received packets at the destination. The example in Table 1 illustrates this metric and its difference to the extent metric. Packet at $i = 5$ with sequence number $s[i] = 3$ is reordered and 2-reordered. Its extent is also 2. The packet at $i = 6$ with sequence number 4 is reordered and its extent is 3. But it is not n-reordered as there is no n, which satisfies the n-reordering definition. This is inline with TCP, as it does not invoke a dup-ack event

3. SIMULATION SCENARIO

3.1. Simulation model

We consider the OBS deflection scenario as depicted in Figure 1. A single link is congested with probability p. If the link is congested the burst is deflected to an alternative link with a constant additional delay of Δ, which represents the time on the pre-defined deflection path. We assume a constant inter-arrival time of the bursts of $1/\lambda_B$. With both parameters we determine the burst extent to be $e_B = \lambda_B \Delta$. The notation of the burst extent e_B let us draw more general conclusions, as the burst extent is determined by the product of the burst inter-arrival time and the additional link delay.

In our model we assume a greedy TCP source with a constant packet inter-arrival time. We are aware, that TCP transient effects, e.g., adaptation of the congestion window size at the source, are not included in our model. We assume a TCP connection in steady state. The assembly unit is parameterized to generate bursts with a given number of packets n_p per flow per burst. We further assume a constant packet size s_p and a constant burst size s_B. With these parameters the flow sending rate on IP layer is calculated by $r_P = \lambda_B n_P s_P$ (1). In section 4 these parameters are classified in a common OBS scenario.

3.2. TCP throughput model

If the receiver does not use selective acknowledgements, and the sender does use the basic congestion control according to [5] reordering has the same effect as packet loss. Then reordered packets exceeding the dup-ack threshold trigger the fast retransmit algorithm. In these scenarios the TCP throughput can be estimated by Mathis' formula [6]. The TCP throughput B_{TCP} is determined at a minimum of three factors, the end-to-end

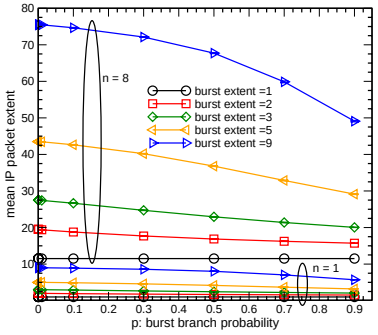

Figure 3. IP packet Extent

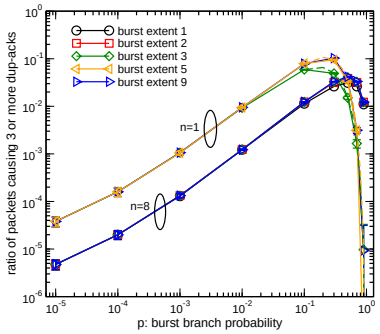

Figure 4. Ratio of packets causing dup-acks

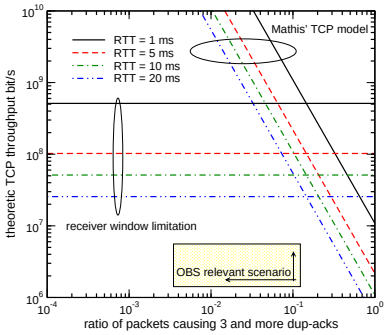

Figure 5. Impact of burst reordering on TCP performance

bottleneck available bandwidth $B_{Bottleneck}$, the congestion window B_{CWD} and the receiver window size B_{RWD}: $B_{TCP} = \min(B_{CWD}, B_{RWD}, B_{Bottleneck})$. The congestion window limitation is given by $B_{CWD} = C\, b_{MSS} / RTT / \sqrt{p}$, where p is the packet loss probability, RTT the round trip time and $b_{MSS} = 1460$ B the maximum segment size (Ethernet). $C = 0.93$ is a constant of proportionality to reflect random loss and delayed acknowledgements. The receiver window size is limited by $B_{RWD} = W_{RWD} / RTT$. The value of the receiver window depends on the host system resources, a typical value is $W_{RWD} = 64$ kB. The bottleneck bandwidth is left out as this is one conclusion of this paper.

4. SIMULATION RESULTS

First, we investigate the ratio of reordered packets dependent on the burst reordering extent. In Figure 2, the ratio of reordered IP packets is depicted with respect on the burst branch probability p for a selected number of burst extent values. For small p the graphs follow the bisecting line, for very large values of p the reordering ratio is zero again. The reordering ratio R_{IP} forms a parable with the equation $R_{IP}(p, e_B) = p(1 - p^{e_B})$ [7], where e_B the assumed burst extent is. In the figure, the calculated curves are drawn with dashed lines respectively.

This graph holds for every number of packets per burst, as all packets in a burst are considered as reordered, if the reordering definition is satisfied. The figure also shows that the number of reordered packets on IP level is limited. Both, the maximum reordering ratio and the point of maximum reordering ratio can be calculated according to [7]. From the figure we also see, that the branch probability with maximum reordering ratio is larger than 0.5. The curve of all maxima is also drawn for convenience. The maximum reordering which can be achieved depends on the burst extent and is shifted towards higher branch probabilities.

In Figure 3 the impact of burst extent on the IP extent is presented for different burst branch probabilities. The graph is given for 1 and 8 packets per burst per flow. We can observe, that the extent increases, with the number of packets per burst. This is expected according to the definition of the extent. If a burst with n_P packets is shifted by e_B bursts, the first packet in the shifted burst has an extent of $e_B n_P$, while the last packet in the burst has an extent of $(e_B + 1) n_P - 1$. The second observation is inline with the outcome of Figure 2. The mean packet extent decreases slightly as the burst branch probability increases. This is due to the fact, that the probability of the maximum possible burst extent decreases.

In Figure 4 we see the influence of burst reordering on the TCP level dependent on the number of packets per burst. In the figure the curves for 1 packet per burst and 8 packets per burst and flow are depicted. The graph show the amount of packets invoking the fast retransmit algorithm at the sending node dependent on the branch probability and the burst extent. The dashed lines again show the calculated results. From [8] the deflection probability is smaller than 10^{-1} in the relevant operating points. The branch probability is given in logarithmic scale to highlight this relevant range. If the branch probability is smaller than 10^{-1} all graphs overlap. Then the burst extent has hardly any influence. After a certain maximum, the dup-ack ratio decreases as the burst branch probability increases.

The second observation is the impact of the number of packets per burst. If the number of packets per burst is increased, the number of packets harming the TCP performance is decreased. As we remember the TCP reordering metric, only the first packet in a burst of consecutive packets causes a dup-ack. Thus, the more packets in a burst, the fewer packets can harm TCP. In contrast to the intuitive assumption, the large number of packets per burst does not have a bad impact on the TCP performance.

With Figure 5 we classify the obtained results in the world of TCP. As mentioned earlier, the probability to invoke the fast retransmit algorithm is comparable to the packet loss probability. According to Padhye's formula, we draw the resulting TCP throughput performance depending on the receiver window size and the reorder/loss probability. The throughput curve is shown for four different round trip times (1ms, 5ms, 10ms, 20ms). Also the limiting receiver window size throughput is depicted.
We found earlier in this paper, that that amount of packets causing a dup-acks is at maximum in the range of 10^{-2} when we consider deflection/buffer rates of common OBS scenarios. We also determine the TCP rate needed for a certain number of packets per burst per flow. In a typical OBS [8] scenario the mean burst size is 100 kbit and the minimal burst IAT is 10 µs to represent a 10 Gbps link. In our scenario we assume a link operation point at 50% load. In the case of 1 packet per burst we determine by (1) the greedy source with 600 Mbps. In case of 8 packets per burst the sender offers a load of 4.8 Gbps. The intersection of the TCP curve and the found TCP flow rate denotes the actual throughput the TCP sender is able to use.
We found that in any case the limiting factor is the TCP sending rate unless the amount of reordering is higher than 10^{-1} in wide area networks with more than 20 ms round-trip time. In these extreme reordering scenarios, the TCP throughput is affected by the OBS reorder behavior. These scenarios would probably either trigger improved TCP versions [9], which can handle high amount of reordering, or new mechanisms in OBS to avoid such high reordering, e.g. adjusted network dimensioning.

5. CONCLUSIONS

In this paper we applied an alternative method to quantify the impact of OBS burst reordering on the TCP performance. Our methodology included a per layer approach, which classifies and measures TCP relevant reordering on burst as well as on packet level.
We found, that the number of reordered packets, invoking the fast retransmit algorithm, decreases with the number of packets per flow and burst. The total number of packets harming the TCP performance is not critical, when the deflection ration on burst level is kept in strict borders. In this case, the limiting TCP performance is either the receiver window or the sender's access rate. We conclude, that reordering has only a slightly impact on the TCP performance, when the deflection or buffering probability is low. When the deflection probability and the round trip time increases, then burst reordering has a large impact on the TCP performance and new TCP versions have to be improved or burst reordering avoided.

ACKNOWLEDGEMENTS

The author would like to thank Guoqiang Hu, Michael Scharf, Martin Köhn and Joachim Scharf for their valuable discussions and helpful comments. Special thanks to Ju Huang for the implementation work.

REFERENCES

[1] M. Schlosser, E. Patzak, and P. Gelpke. Impact of deflection routing on TCP performance in optical burst switching networks. In *Proceedings of the 7th International Conference on Transparent Optical Networks (ICTON)*, volume 1, pages 220–223, Barcelona, June 2005.
[2] S. Gowda, R.K. Shenai, K.M. Sivalingam, and H.C. Cankaya: Performance evaluation of TCP over optical burst-switched (OBS) WDM networks. In *Proceedings of the IEEE International Conference on Communications (ICC),* volume 2, pages 1433–1437 vol.2, 2003.
[3] Andrea Detti and Marco Listanti. Impact of segments aggregation on TCP Reno flows in optical burst switching networks. In *Proc. IEEE INFOCOM*, 2002.
[4] A. Morton, L. Ciavattone, G. Ramachandran, S. Shalunov, J. Perser: Packet Reordering Metrics, *RFC 4737*, November, 2006.
[5] M. Allman, V. Paxson, W. Stevens: TCP Congestion Control, *RFC 2581*, April 1999
[6] M. Mathis, J. Semke, Jamshid Mahdavi, and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm," ACM Computer Communication Review, vol. 27, no. 3, pp. 67–82, Jul. 1997.
[7] Sebastian Gunreben: First analytic multi-layer analysis of TCP over OBS, University of Stuttgart, Institute of Communication Networks and Computer Engineering, 2007.
[8] Christoph M. Gauger, M. Köhn, and J. Scharf. Comparison of contention resolution strategies in OBS network scenarios. In *Proceedings of the 6th International Conference on Transparent Optical Networks (ICTON)*, volume 1, pages 18–21 vol.1, 2004.
[9] Zhang, M., Karp, B., Floyd, S., and Peterson, L., RR-TCP: A Reordering-Robust TCP with DSACK, *in Proceedings of the Eleventh IEEE International Conference on Networking Protocols (ICNP 2003), Atlanta, GA*, November, 2003.