Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Networks 45 (2004) 505–521
COMPUTER NETWORKS
www.elsevier.com/locate/comnet

TCP and UDP performance for Internet over optical packet-switched networks ☆

Jingyi He [a,*], S.-H. Gary Chan [b,*]

[a] Department of Electrical and Electronic Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong
[b] Department of Computer Science, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

Received 25 November 2003; received in revised form 17 February 2004; accepted 25 February 2004
Available online 10 April 2004

Responsible Editor: S. Low

Abstract

A strong candidate for the future Internet core is optical packet-switched (OPS) network. In this paper, we study the impact of mechanisms as employed in OPS networks on the performance of upper layer Internet protocols represented by TCP and UDP. The mechanisms we investigate are packet aggregation, deflection routing, and ingress buffering. We show that packet aggregation in general improves TCP throughput, and the improvement increases with the aggregation interval (or optical packet size). With the packets destined to the same egress optical switch, aggregation may be done at different granularities: aggregating all the packets (*full aggregation*), aggregating packets from the same traffic class (*per-class aggregation*), and aggregating packets from the same flow (*per-flow aggregation*). We show that with per-class aggregation and per-flow aggregation some flows may be severely penalized in throughput at large aggregation intervals, resulting in significant degradation in TCP fairness, because of the synchronization problem with shared queueing. By using weighted fair queueing (WFQ) at the ingress buffer, in contrast, we show that differentiated QoS (in terms of throughput) can be provisioned for both TCP and UDP traffic even with deflection routing. Deflection routing avoids packet losses, but results in out-of-order packet delivery and increased packet delay jitter. We show that TCP throughput can be significantly improved by deflection routing in spite of the packet reordering, and the UDP packet delay jitter introduced by deflection routing can be alleviated by packet aggregation and ingress buffering. We also show that ingress buffering significantly improves TCP throughput and the ingress buffer only needs a small size (in terms of the number of optical packets).

Keywords: Optical packet-switched networks; Optical Internet; Packet aggregation; Deflection routing; Ingress buffering; TCP; UDP; Fairness; Differentiated QoS

☆ This work was supported, in part, by the Competitive Earmarked Research Grant of the Hong Kong Research Grant Council (HKUST6199/02E) and Area of Excellence Scheme in Information Technology from the University Grant Council of Hong Kong (AoE/E-01/99).

* Corresponding authors. Tel.: +852-2358-6990; fax: +852-2358-1447.
E-mail addresses: eehjy@ust.hk (J. He), gchan@cs.ust.hk (S.-H.G. Chan).

1389-1286/$ - see front matter
doi:10.1016/j.comnet.2004.02.011

1. Introduction

Driven by the ever-increasing demand for bandwidth, the core of the Internet has been evolving from an electronic network to an optical one (i.e., the so-called Optical Internet). This is mainly because optical networks offer large bandwidth through the use of wavelength-division multiplexing (WDM). There are in general three different ways to route packets in the Optical Internet: wavelength routing, optical burst switching, and optical packet switching. Wavelength routing is essentially circuit switching, in which two nodes communicate by setting up an all-optical *lightpath*. Such a wavelength circuit is suitable for delivering high-bandwidth continuous media streams. However, it may be severely under-utilized if the traffic in this circuit is bursty as in a data (IP) network. With optical burst switching, the bandwidth on a path is reserved only for short durations, e.g., the transmission time along the path of a data burst. Therefore, the possible bandwidth underutilization is limited to short periods. With optical packet switching, the wavelength channels on the links are not reserved for any connection: as soon as an optical packet is forwarded, the corresponding wavelength channel can be used to transmit the next packet from other flows. In other words, optical packet-switched (OPS) network offers bandwidth granularity at the packet level, achieving higher bandwidth efficiency. Moreover, because of its packet-switched nature, OPS network can have richer routing functionalities and greater flexibility in supporting diverse services [1–3]. As optical packet switching still faces some technological difficulties such as the lack of optical random access memory, wavelength routing and optical burst switching are regarded as near-term solutions to the Optical Internet. Nevertheless, given its advantages and the promising technological progress in optical packet switching [4], OPS network is a strong candidate for the long-term future Optical Internet backbone.

In an OPS-based Internet, end users are still attached to electronic networks. Their traffic is aggregated before being forwarded to the OPS backbone. We show in Fig. 1 an OPS network which is used to transport the traffic between IP subnetworks. Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are currently the predominant transport protocols used at the end users. These packets will eventually be forwarded in the optical core. The impact of mechanisms as used in the OPS network (e.g., packet aggregation, deflection routing, and ingress buffering) on the performance of the Internet protocol suite is not well addressed yet, and is the subject of this paper.

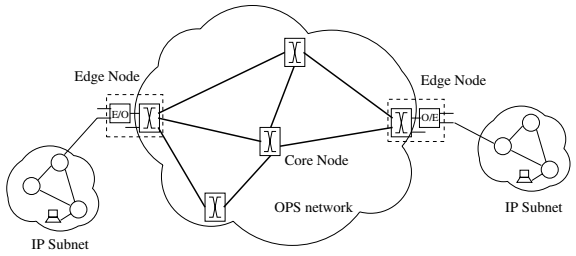


Fig. 1. An optical packet-switched (OPS) network used to transport traffic between IP subnetworks.

Packet aggregation is a process performed at the ingress node of the OPS network that aggregates IP packets destined to the same egress node to form a larger optical packet. [1] This is necessary because in high bit-rate OPS networks processing individual IP packets either imposes too heavy burden to the optical packet switches or is even impractical. For example, in an OPS network operating at 10 Gbps and with IP packet size of 1000 bytes, without packet aggregation the optical switches need to switch a packet once every 0.8 μs for a single input line, while most of the up-to-date all-optical switches have a switching speed in the order of milliseconds. Moreover, when forming each optical packet from IP packets, an optical packet header needs to be added. Obviously, packet aggregation also reduces the optical header overhead. When an optical packet reaches the egress node, it is segregated through optical–electronic (O/E) conversion into individual IP packets, which are then transmitted to their respective destinations. Packet aggregation affects the transmission of IP packets primarily in the following

[1] In this paper, we also use optical packet to refer to its counterpart in the electronic domain before the electronic–optical (E/O) conversion.

two ways. First, additional delay will be incurred for the IP packets other than the last one in the optical packet. Second, as an optical packet may contain multiple packets from the same flow, when it is delivered, dropped or mis-routed in the OPS network, the impact on that flow could be magnified. In this regard, how packet aggregation affects the TCP and UDP performance is an important issue.

Deflection routing is a contention resolution scheme usually used in OPS networks. In packet-switched networks, contention occurs when two or more packets are to be forwarded to the same output port. In electronic networks, such contention is resolved in a store-and-forward manner by temporarily buffering the contending packets in random access memory (RAM). However, there is no effective optical RAM, and optical buffers are usually implemented using fiber delay lines (FDLs). Due to the volume of FDLs, optical buffer usually has a small capacity. Given all these limitations in optical buffers, contention in OPS networks is in general resolved either by wavelength conversion or by deflection routing. In this paper, we only consider deflection routing, which is simpler and much cheaper than wavelength conversion: the packets losing the contention are temporarily mis-routed, or "deflected," to other output ports. Clearly, deflection routing prevents packet from being lost by routing them to longer paths. On the other hand, it also leads to out-of-order delivery of the packets. For a protocol which has reassembly deadline and congestion control mechanism such as TCP, this will adversely affect its performance. Therefore, the overall effect of deflection routing on TCP performance merits a study. Also of interest is its impact on the delay performance of UDP traffic.

Ingress buffering is a technique that uses the electronic buffer to reduce the packet loss rate at the ingress optical switch of the OPS network [5]. Specifically, a newly-formed optical packet may find its preferred outgoing link being used to transmit other optical packets at the mean time. If the optical packet is forwarded to the ingress optical switch right away, it will be lost due to the contention (assuming no optical buffer). With ingress buffering, however, the optical packet can temporarily be buffered in the electronic domain and get forwarded later, thus reducing the packet losses. It is of interest to study the buffer size requirement and the performance improvement that can be achieved for TCP and UDP.

The major contributions of this paper are summarized as follows:

- This paper studies three packet aggregation schemes, namely, *full aggregation* which aggregates all the packets destined to the same egress optical switch, *per-class aggregation* which aggregates packets from the same traffic class, and *per-flow aggregation* which aggregates packets from the same flow. We show that packet aggregation in general improves TCP throughput, and the improvement increases with the aggregation interval (or optical packet size). With per-class aggregation and per-flow aggregation, however, some flows may be severely penalized in throughput at large aggregation intervals, resulting in significant degradation in TCP fairness, because of the synchronization problem with shared queueing. We also show that different aggregation schemes do not have big impact on UDP throughput.
- This paper studies the impact of deflection routing on TCP and UDP performance. We show that TCP throughput can be significantly improved by deflection routing in spite of the packet reordering, and the UDP packet delay jitter introduced by deflection routing can be alleviated by packet aggregation and ingress buffering.
- This paper studies the impact of ingress buffering on TCP and UDP performance. We show that ingress buffering significantly improves TCP throughput and the ingress buffer only needs a small size (in terms of the number of optical packets). Moreover, we show that by using weighted fair queueing (WFQ) at the ingress buffer, differentiated QoS (in terms of throughput) can be provisioned for both TCP and UDP traffic even with deflection routing.

The rest of this paper is organized as follows. We first review related previous work in Section 2. We

then present in Section 3 the system description of the OPS network under study and our simulation model. In Section 4, we present illustrative simulation results of the impact of packet aggregation, deflection routing and ingress buffering as used in the OPS network on TCP and UDP performance. We conclude in Section 5.

2. Previous work

Packet aggregation was previously proposed for the Internet with the objective to reduce the number of small packets, e.g., the TCP ACKs for web servers [6], or voice over IP (VoIP) packets [7]. By doing so, the processing overhead at the routers as well as the packet loss rate can be reduced. Packet aggregation in OPS (or OBS) networks starts to attract attention only recently [5,8–10]. These studies assume an aggregation scheme which simply aggregates all the IP packets destined to the same egress node. Our previous work has studied a scheme which aggregates IP packets from the same flow [11]. In this paper, we further study a scheme which aggregates IP packets belonging to the same traffic class, and compare the performance of all these different aggregation schemes. Moreover, the study in this paper considers a system architecture where ingress buffering is used.

Deflection routing has long been proposed and studied. However, it is mainly investigated in *slotted* optical network of *regular* topologies such as the Manhattan Street Network (MS-Net) and the ShuffleNet [12–14]. We believe that in the future Optical Internet, unslotted network is more likely because it supports variable-size packets as characterized in bursty IP traffic and is easier to manage [15,16], and irregular mesh topologies are also more likely to be the case. Recent studies begin to address the deflection routing in irregular OPS networks [17–19]. However, these studies, and as well as others, have not considered the effect of out-of-order packet delivery by assuming that the receiver has an infinite reassembly time and buffer. This is certainly not the case when TCP flows are considered. In particular, TCP has a maximum receiver window of 64 KB (without the TCP window scale option), and has flow control mechanisms featured by duplicated acknowledgement, fast retransmit and fast recovery in response to out-of-order delivery. We have previously explored the possibility of using deflection routing in the Internet and its impact on TCP performance [20]. We show that by judiciously selecting the packet to be deflected in the queue, the effect of out-of-order packet delivery can be mitigated and TCP throughput can be increased significantly. This result, however, does not extend to OPS networks where no RAM is available. In this paper we study the impact of deflection routing on TCP and UDP performance in the context of OPS network with such special characteristics as the use of packet aggregation.

Ingress buffering has been shown to be able to reduce the packet loss rate at the ingress optical switch [5]. The requirement on the ingress buffer size is important to the network design and is of our interest here. Moreover, we explore the possibility of using some fair queueing scheme at the ingress buffer to offer differentiated QoS for the OPS network.

3. System description and simulation model

3.1. System description

As shown in Fig. 1, the OPS network consists of edge nodes and core nodes inter-connected by optical fiber links. The core nodes are basically optical switches that are responsible for routing the optical packets. The edge nodes interface with IP subnetworks, and are responsible for packet classification, aggregation, and segregation, in addition to the switching functionalities.

The edge node structure considered in this paper is shown in Fig. 2. At the ingress side, the incoming IP packets are firstly classified and aggregated by the aggregator to form larger optical packets. The classification can be simply based on the destination egress node of the IP packets, and all the IP packets having the same destination egress node are aggregated together. We term this scheme *full aggregation*. However, there are options with finer aggregation granularity. One

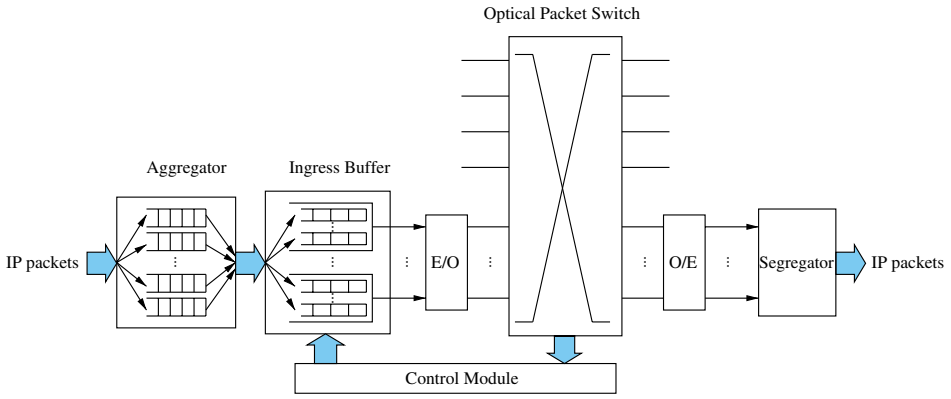


Fig. 2. The structure of the edge nodes in the OPS network.

scheme is *per-class aggregation*, in which among the IP packets destined to the same egress node only those belonging to the same traffic class are aggregated together. This is natural in the framework of differentiated services (DiffServ). Another scheme is *per-flow aggregation*, in which only IP packets from the same flow are aggregated. This scheme makes the handling of individual flows possible. The case without packet aggregation can be thought of as a scheme with the finest aggregation granularity and is a benchmark for evaluating other aggregation schemes.

The aggregator maintains aggregation queues for the IP packets to be aggregated together. At each queue, an optical packet is formed when a fixed aggregation interval expires since the arrival of the first IP packet. Optical packets generated this way can therefore have variable sizes. In practice, one may also want to limit the size of an optical packet. In that case, an aggregation interval expires when the size of the optical packet reaches a predetermined threshold. We show in Fig. 3 an example of the optical packet format.

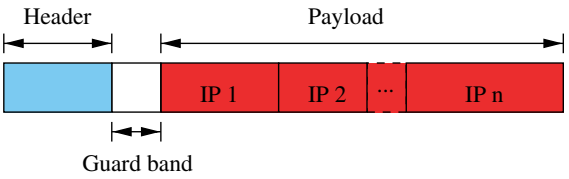


Fig. 3. An example of the optical packet format.

The header consists of such fields as delineation and synchronization bits, label for switching (routing), and payload information. The payload is basically the aggregated IP packets. The guard band between the header and payload accounts for the time needed for header processing and switch configuration. Note that in slotted system, an additional guard band is needed after the payload to form a fixed-length optical packet, as in the KEOPS project [21].

A newly-formed optical packet is forwarded to the (electronic) ingress buffer, which maintains a number of queues for each outgoing link at the node. If the number is one, then all the optical packets to be forwarded to the same outgoing link contend for the single first-in first-out (FIFO) queue. In the case of multiple queues per outgoing link, it is possible to implement some scheduling algorithm to offer DiffServ together with per-class aggregation. A control module is used to monitor the state of the optical switch. An optical packet at the head of a queue can be forwarded into the optical switch (after O/E conversion) only if its preferred outgoing link is free. If a newly-formed optical packet finds the ingress buffer for its preferred outgoing link full, it is deflected to an alternative outgoing link. Note that the deflected optical packet may find the ingress buffers for the alternative links also full, and hence needs to be dropped. Inside an OPS network without any optical buffer at the nodes, the optical packets are transmitted in a "cut-through" manner without

any queueing delay at intermediate nodes. Deflection happens instantaneously when an optical packet encounters contention at the nodes. Optical packet loss is also possible when there is no alternative path available.

When an optical packet reaches the egress node, it is segregated through optical–electronic (O/E) conversion into individual IP packets, which are then transmitted to their respective destinations. Note that once an optical packet is formed, it would not be segregated until it reaches the egress node (i.e., the core nodes do not combine or split optical packets).

3.2. Simulation model

We use the *ns*-2 network simulator to do our simulations [22]. [2] The simulated network model is shown in Fig. 4. The OPS network is used to transport IP traffic which enters the OPS network at the same ingress node r_0 and leaves at the same egress node r_1. At r_0, the IP packets are aggregated with an aggregation interval Δ, and size limit $B_0\Delta$. The optical packet header has a size of 20 bytes (as the IP header), and the guard band is ignored. An ingress buffer of size b (in terms of the number of optical packets) is equipped for each outgoing link.

The OPS network provides a shortest path as well as an additional deflection path for the traffic. For simplicity, we assume that the shortest path is lossless and all the optical packets forwarded to it can reach the egress node. The lossy deflection path with variable delay is used to model the possible deflections in a real mesh OPS network: After entering the OPS network, optical packets may be deflected at different points and for different number of times, and some of them may even be dropped because of contention. We define a parameter α to denote the probability that a deflected optical packet is successfully delivered by the OPS network, namely *successful deflection rate*. The deflection cost t_c, which is the delay difference

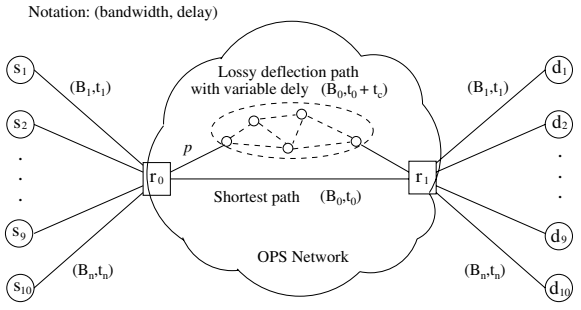


Fig. 4. The simulation model we use.

between the deflection path and the shortest path, is assumed to vary from time to time, with the interval uniformly distributed between 1 and 100 ms. Each time the value of t_c is taken from the positive values of a Gaussian random variable with mean μ ms and standard deviation 4 ms.

The values of other parameters are summarized as follows. The bandwidth of the optical links is $B_0 = 622$ Mbps. The bandwidth of the links outside the OPS network are all the same with $B_i = 155$ Mbps. The shortest path between r_0 and r_1 has a fixed delay of $t_0 = 10$ ms. The delay of the links outside the OPS network is $t_i = 2i$ ms, $i = 1, 2, \ldots, 10$.

The IP traffic is differentiated into five classes, with each class having 100 flows uniformly distributed among the 10 s_i–d_i pairs. Three of the classes are TCP (Reno) flows with bulk data transfer (e.g., FTP), denoted as class 1, class 2, and class 3. The other two classes are UDP flows with same poisson arrival rate, denoted as class A and class B. All the flows have the same packet size of 1500 bytes. The starting times of the flows are uniformly distributed in a time interval of $[0.1, 0.5]$ second to avoid global synchronization. In the reverse direction, r_1 acts as the ingress node and r_0 the egress node. The traffic in this direction (i.e., TCP ACKs) is treated in the same manner as the traffic in the forward direction.

The baseline system has aggregation interval of $\Delta = 1$ ms, ingress buffer of size $b = 2$ optical packets, successful deflection rate $\alpha = 0.8$, mean deflection cost $\mu = 5$ ms, and overall UDP traffic load 400 Mbps. These parameters will be varied one at a time to show their impact on the system performance.

[2] Note that *ns*-2 is for simulating store-and-forward networks. We modified the simulator so as to simulate an OPS network without optical buffer in which the optical packets are transmitted in a "cut-through" manner.

Regarding the performance metrics, in addition to the throughput (overall and flow), we are also interested in the fairness among the flows, and packet delay jitters (of UDP flows). In terms of fairness, we use Jain's fairness index defined as [23]

$$f = \frac{\left(\sum_{i=1}^{N} x_i\right)^2}{N \sum_{i=1}^{N} x_i^2}, \quad (1)$$

where N is the total number of flows and x_i is the throughput of flow i, $i = 1, \ldots, N$. The value of Jain's fairness index is always between zero and one, i.e., $f \in (0, 1]$, with a larger value meaning better fairness and one meaning perfect fairness. We will focus on the fairness among TCP flows and UDP flows respectively. We do not consider the overall fairness among all the flows because it is dependent on the offered UDP traffic load: when the load is high, TCP throughput tends to be much smaller than UDP and hence very bad fairness could result; with a moderate UDP load, however, TCP flows may achieve throughput comparable to UDP flows, leading to very good fairness. We define the delay jitter of a UDP flow as the standard deviation of the packet delays of that flow. The overall UDP delay jitter is simply the average delay jitter over all the UDP flows.

4. Illustrative simulation results

In this section, we present illustrative simulation results regarding the impact of packet aggregation, deflection routing, and ingress buffering on TCP and UDP performance.

4.1. Impact of packet aggregation

4.1.1. On TCP performance

We first study the impact of packet aggregation on TCP performance. On one hand, packet aggregation introduces additional packet delay, which increases both the end-to-end round trip time (RTT) and the retransmission time-out (RTO), and hence decreases the TCP throughput. On the other hand, packet aggregation may also improve TCP performance in several ways.

Firstly, packet aggregation may help TCP operate in larger congestion window *(cwnd)* values. Without packet aggregation, random single packet losses are dominant and TCP recovers from such losses through the fast recovery and fast retransmit mechanism. The short time intervals between two consecutive losses may keep *cwnd* low, because it is halved for each such packet loss and then increases only linearly. On the other hand, packet aggregation may pack multiple successive TCP segments of the same flow into one optical packet. TCP may recover from such an optical packet loss only through the RTO mechanism, with which *cwnd* drops to one after RTO and then increases exponentially in slow start and linearly in congestion avoidance. Exponential increase in slow start quickly opens *cwnd*. Moreover, with the same packet loss rate, a burst loss is followed by a longer lossless period which allows *cwnd* to increase to a larger value. Larger *cwnd* leads to higher sending rate and hence higher throughput. This aggregation benefit has been studied with respect to the packet loss rate in [8]. We present a simple analysis on its relationship to the aggregation interval in Appendix A.

Secondly, appropriate packet aggregation can have traffic shaping effect and hence reduce packet losses. For example, with full aggregation the ingress optical switch only needs to switch one optical packet to an outgoing link every fixed aggregation interval. In other words, the burstiness of IP traffic as originally seen by the optical switch without aggregation has been smoothed out. Note that per-flow aggregation, however, does not help reduce the contention at the ingress optical switch. This is because the aggregation timer of each flow are not synchronized, and hence the optical packets formed from different flows are forwarded to the optical switch still in a random manner. Moreover, because the optical packets have a larger size, packet losses may be even more severe than that without aggregation. That explains why TCP throughput could be even lower with per-flow aggregation than without aggregation (when Δ is not very large), as shown in Fig. 5(a). For per-class aggregation, the traffic shaping effect is between per-flow aggregation and full aggregation. We hence observe that it achieves

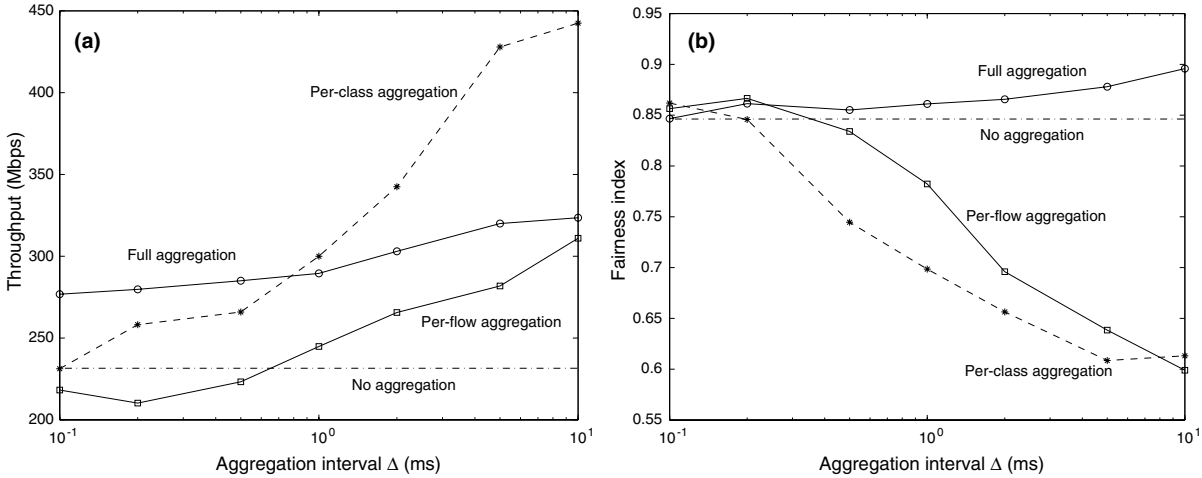


Fig. 5. TCP performance versus aggregation interval Δ with different aggregation schemes: (a) throughput; (b) fairness index.

throughput larger than per-flow aggregation while lower than full aggregation at small aggregation intervals ($\Delta < 1$ ms).

Thirdly, packet aggregation reduces TCP retransmissions. This is a joint effect together with deflection routing, which we elaborate in the following. With deflection routing, TCP may easily enter fast retransmit if more than three packets of larger sequence numbers than the deflected one(s) arrive at the destination earlier through the shortest path. With packet aggregation, the multiple packets from the same flow in an optical packet are in order among themselves. When such an optical packet is deflected, the block of in-order packets of that flow which will arrive late does not cause more TCP retransmissions than a single deflected TCP packet of that flow, because TCP fast retransmit is triggered by the duplicated ACKs generated by the arrival of larger sequence number packets through the shortest path. With a larger aggregation interval, deflection occurs less frequently, resulting in fewer retransmissions. Therefore, the TCP congestion window can be kept at a larger value on average, as well as there are fewer duplicated packets wasting the bandwidth. The overall effect is larger TCP throughput for larger aggregation interval. This is true for all the three aggregation schemes, as shown in Fig. 5(a). Of course, the aggregation interval could not be too large either, because that would result in too large queueing delay for the packets, which defeats the purpose of pipelining in the TCP sliding window mechanism, as pointed out in [5].

Also of interest is the fairness among the TCP flows, as shown in Fig. 5(b). We observe that full aggregation can slightly improve TCP fairness as compared with no aggregation. This is because with full aggregation an optical packet may contain IP packets from many flows (not necessarily all the flows due to the bursty nature of TCP traffic), and all these packets will be subjected to the same process (switching, deflection, or even loss) in the OPS network. The larger the aggregation interval is, the more flows may be aggregated in an optical packet, and hence all the flows are more fairly affected when transported by the OPS network. Therefore, there is also a slight increase in TCP fairness as the aggregation interval increases for full aggregation.

In contrast, for per-class and per-flow aggregation, TCP fairness index decreases significantly when the aggregation interval is large. The major reason is the synchronization among the aggregated traffic trunks (classes for per-class aggregation and flows for per-flow aggregation). For per-flow aggregation, for example, the optical packets of each flow are formed and forwarded to the ingress buffer every fixed interval (as long as

the size limit of the optical packet is not reached). Although the starting time of each flow is randomly chosen, the relative phase between the optical packets of any two flows is fixed. When contending for the ingress buffer, some flows may persistently lose packets more often than others, hence achieving lower throughput. Therefore, the fairness among the flows may be bad. The larger the aggregation interval is, the more severe the synchronization effect is. For per-class aggregation, similar effect exists among the different classes of traffic. We show in Fig. 6 a representative case of per-class aggregation, where class 2 traffic has substantially lower throughput than the other two TCP classes (class 1 and 3). We have found that fluctuating the aggregation interval (e.g., randomly choosing values in $[0.9, 1.1]\Delta$) can alleviate the synchronization problem, but the alleviation is not as prominent for large aggregation intervals as for small ones. With shared ingress buffering, the problem may not be easily solved when the optical packets are large. A better solution would be some fair queueing scheme, as will be shown later.

It has been widely observed that an unfair share of bandwidth may result in higher overall TCP throughput. For example, if multiple TCP flows share a link, the maximum throughput can be achieved when one TCP flow totally occupies the link while the rest are completely starved. If all flows have a certain share of the bandwidth, there are more packet losses and thereof retransmissions. Therefore, the bandwidth of the link is used less efficiently, and lower throughput results. For per-class aggregation in our case, the decrease in fairness at large aggregation intervals can help to the increase the throughput substantially to values even larger than what full aggregation can achieve, as shown in Fig. 5(a).

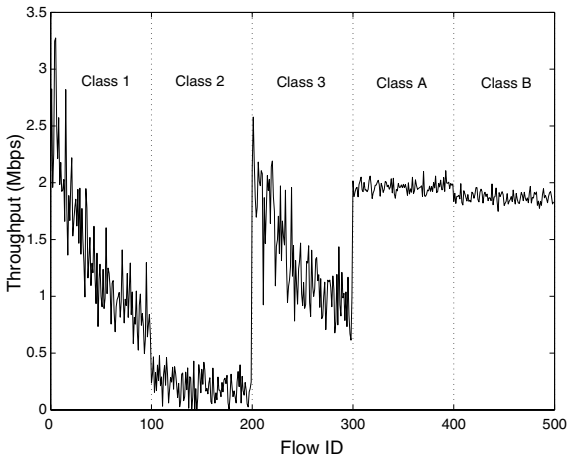


Fig. 6. A representative case of per-class aggregation ($\Delta = 1$ ms): class 2 traffic persistently loses contention and hence has a low throughput.

In the above simulations, the TCP ACKs are also aggregated before entering the OPS network. We have also run simulations without applying aggregation to the ACKs, and the results show that aggregating TCP ACKs does not affect TCP performance compared to otherwise. The major reason is that TCP ACKs have small packet size and do not consume much bandwidth. Therefore, packet aggregation does not help much in reducing the ACK losses. Actually, we rarely observe losses or deflections of TCP ACKs at r_1 even without aggregation.

4.1.2. On UDP performance

We next study the impact of packet aggregation on UDP performance. As shown in Fig. 7(a), UDP throughput does not differ by much under different aggregation schemes. This is because UDP is aggressive and always tries to grab the available bandwidth. The slight decrease in throughput as the aggregation interval increases is because as the TCP throughput increases more UDP packets are lost due to the heavier contention. Under all the aggregation schemes, UDP can always achieve almost perfect fairness among the flows.

Packet delay jitter is another important performance metric for UDP traffic. We show in Fig. 7(b) the average delay jitter over all the UDP flows versus the aggregation interval. Not surprisingly, larger aggregation interval leads to larger delay jitter. The interesting observation is that full aggregation and per-class aggregation can achieve even lower delay jitter than no aggregation. The reason is that these two schemes generate large optical packets by aggregating a lot of flows. With ingress buffering, the large queueing delay can partially cancel out the deflection cost. Therefore,

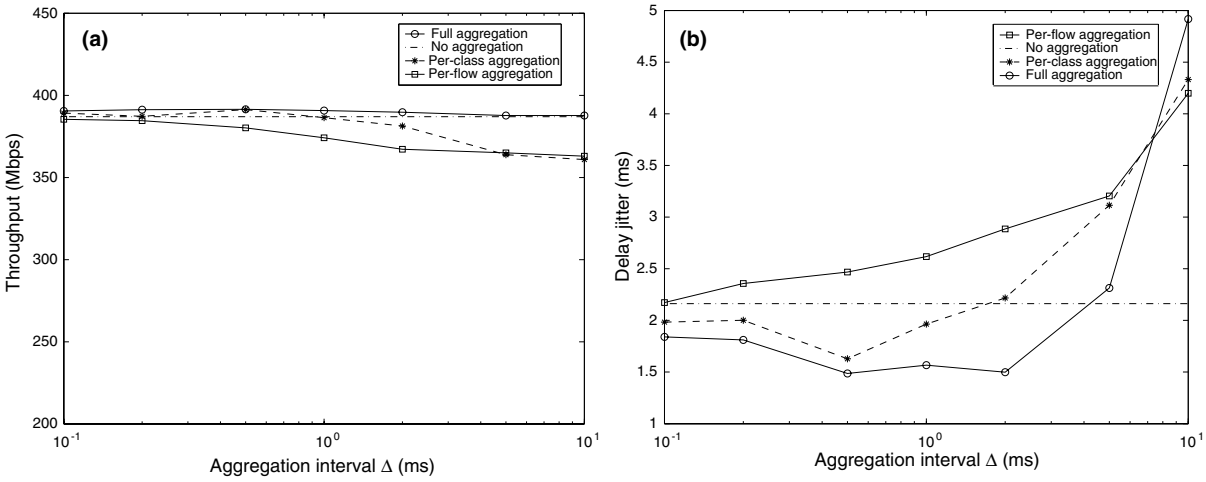


Fig. 7. UDP performance versus aggregation interval Δ with different aggregation schemes: (a) throughput; (b) packet delay jitter.

the overall delay jitter is reduced. When the aggregation interval is very large, the cancelling effect is negated by the large aggregation delay and the overall effect is a positive addition to the delay jitter.

4.2. Impact of deflection routing

In this subsection, we study the impact of successful deflection rate α and the mean deflection cost μ on TCP and UDP performance. We take full aggregation and no aggregation (which have the coarsest and finest aggregation granularity, respectively) as comparative examples.

We show in Fig. 8 TCP and UDP throughput versus α. Successful deflection rate $\alpha = 0$ is essentially the case without deflection routing, while $\alpha = 1$ means the deflected packets can always be successfully delivered without loss. For UDP, reception of deflected packets directly accounts for the increase in throughput. Therefore, with both full aggregation and no aggregation, UDP throughput increases linearly with respect to α. The improvement is more significant for no aggregation than full aggregation, because without packet aggregation the contention is more severe and the usefulness of deflection routing is more prominent. For TCP, we observe that deflection routing can significantly improve its throughput. For full aggregation, deflection routing with $\alpha = 1$ can double the throughput; for no aggregation, the increase can be nearly four times. Although we also see that the loss of deflected packets severely degrades the usefulness of deflection routing, we can still observe a throughput improvement of more than 20% at $\alpha = 0.8$ with full aggregation. In other words, deflection routing improves TCP throughput in spite of the out-of-order packet delivery.

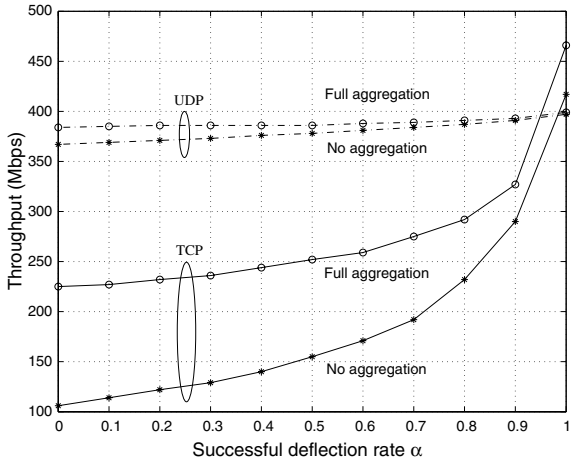


Fig. 8. TCP and UDP throughput versus the successful deflection rate α.

The deflection cost reflects the additional delay introduced by deflecting an optical packet. We

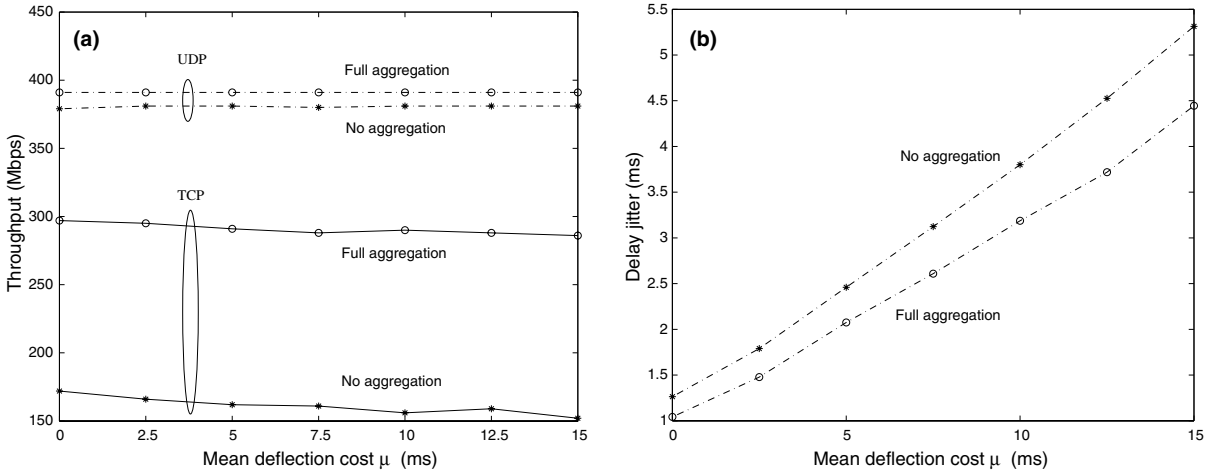


Fig. 9. Impact of the mean deflection cost μ: (a) throughput; (b) packet delay jitter.

show in Fig. 9(a) TCP and UDP throughput versus mean deflection cost μ. We observe that UDP throughput is independent of μ. This is because with constant offered load and successful deflection rate, the number of UDP packets delivered in a certain period is constant at the steady state. For TCP, there is a slight decrease in throughput as μ increases, because the RTT increases with μ. Similar as the throughput, the fairness index is also independent of μ and is not shown here. We show in Fig. 9(b) UDP packet delay jitter versus μ. As can be seen, the delay jitter increases linearly with μ. Also observed is that full aggregation achieves lower delay jitter than no aggregation. This is because the large queueing delay with full aggregation partially cancels out the effect of the deflection cost, as also demonstrated in Fig. 7(b).

4.3. Ingress buffer requirement and differentiated QoS provisioning

Ingress buffering has been shown to be able to reduce the packet loss rate at the ingress optical switch [5]. The requirement on the ingress buffer size is important to the network design and is of our interest here. As in many router implementations the queue size is in terms of the number of packets instead of the actual physical size (bytes) [6], we study the ingress buffer size in terms of the number of optical packets.

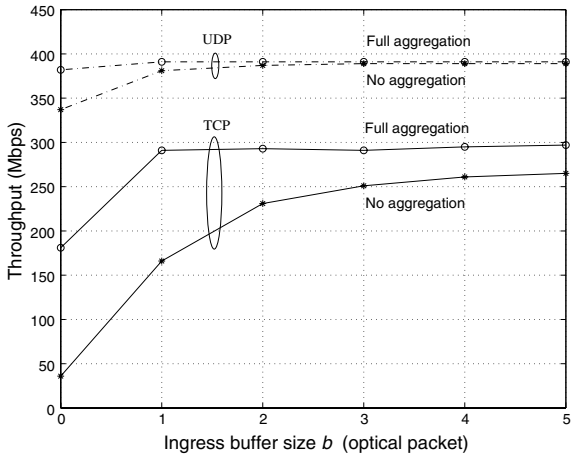


Fig. 10. TCP and UDP throughput versus the ingress buffer size.

We show in Fig. 10 TCP and UDP throughput versus the ingress buffer size b, with full aggregation and no aggregation respectively. A first observation is that the ingress buffer size need not be large. With full aggregation, a size of only one (optical packet) is enough. This is because with full aggregation and deflection routing, an ingress buffer of size one can already ensure that almost no optical packet gets dropped at the ingress node. Even without aggregation, in which case the traffic is more bursty, the ingress buffer still only needs to be able to accommodate a few packets. The reason

is that a few ingress buffer can already smooth out the burstiness of the ingress traffic so that the bandwidth of the optical links is efficiently used with few idle times. In particular, the burstiness of the ingress traffic is not expected very high: On one hand, the burstiness of UDP traffic is small as we assume constant Poisson arrival rates; On the other hand, the TCP traffic rate variation is alleviated by deflection routing which reduces packet losses. Therefore, larger ingress buffer size offers little additional improvement in reducing packet loss rate and hence increasing the throughput. A second observation, which has also been shown in [5], is that ingress buffering significantly improves TCP throughput. As shown in the figure, the improvement can be more than 50% with full aggregation.

So far in our simulation, the ingress buffer is equipped at the ingress nodes in a "one for each outgoing link" manner. In other words, it is shared queueing. With per-class aggregation, for example, optical packets for different classes destined to the same egress node have to contend for the same ingress buffer. With both TCP and UDP flows existing in such a shared queueing network, how to offer quality-of-service (QoS) to the TCP flows is of great concern, because UDP flows are more aggressive with the resources (e.g., bandwidth and buffer). It is widely observed, as well as confirmed by our own simulation, that TCP traffic may be quenched if the UDP traffic load is excessively high, no matter which aggregation scheme is used. Moreover, even among the TCP flows, some may have substantially lower throughput than others because of persistent loss of contention, as we have shown before.

In the following, we show how differentiated QoS can be provisioned by implementing weighted fair queueing (WFQ) mechanism at the ingress buffer of the OPS network. Specifically, we consider offering proportional bandwidth assignment. Suppose we have a certain number of TCP and UDP classes, each assigned a weight (termed "the class weight") for its fair share of bandwidth. When there is congestion, we would like to proportion bandwidth according to the class weights. Note that for TCP, its fair share of bandwidth on the deflection path may be more than enough, because the throughput improvement by deflection routing is limited (although it can be 20% or even higher in relative value as we have shown). Therefore, it is likely that only a small amount of bandwidth of the deflection path is used for deflected TCP packets. In this case, the free bandwidth can be allocated to UDP flows. In view of this, by differentiated QoS provisioning we mean, in times of congestion:

1. On the shortest path, both TCP and UDP classes are allocated their fair shares of the bandwidth.
2. Overall, the bandwidth allocations among the same kind of traffic (either TCP or UDP) are proportional to their class weights.

In order to achieve this, per-class aggregation is to be used together with WFQ. In order to implement the WFQ mechanism, a separate queue is needed for each class. In other words, the same number of queues as the classes are to be maintained at the ingress node for each outgoing link. Our simulation study is based on the same network model as given in Fig. 4. The three TCP classes (class 1, 2, and 3) are assigned weights 0.4, 0.2, and 0.1, respectively. The two UDP classes (class A and B) are assigned weights 0.2 and 0.1, respectively. We increase the offered UDP load to study how it affects the throughput of different classes when WFQ is implemented. As we know that WFQ maintains the fair share ratios among classes only under high-load (congestion) condition. We introduce 500 Mbps background UDP traffic on the deflection path, leaving 122 Mbps available bandwidth to the deflected traffic. The ingress buffer queue of each class has a size of one.

We show in Fig. 11 per-flow TCP and UDP throughput of different classes versus the offered per-flow UDP load. Clearly, the increase in UDP traffic does not affect much the bandwidth assigned to TCP flows—each TCP flow achieves stable throughput as the UDP load increases. Moreover, their throughput ratios agree well with the their class weight ratios. For the two UDP classes, before their offered loads reach their fair shares of the bandwidth, their throughput increases with the offered loads. Because the two

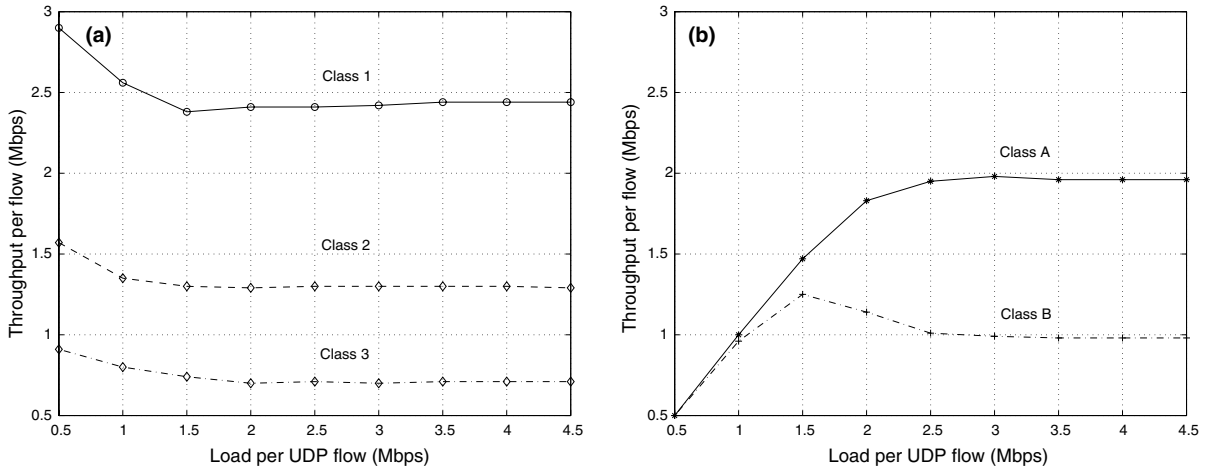


Fig. 11. Differentiated QoS provisioning for three TCP classes (with weights 0.4, 0.2, and 0.1) and 2 UDP classes (with weights 0.2 and 0.1): (a) TCP flows; (b) UDP flows.

UDP classes have the same offered load but different weights, when the offered load reaches the bandwidth fair share of class B, the bandwidth fair share of class A has not yet been reached. Therefore, as the offered load further increases, class B can use the free bandwidth allocated to class A, hence achieving a throughput beyond its fair share. When both of their offered loads increase beyond their fair shares, the bandwidth of the shortest path as well as of the deflection path are allocated to the two classes proportionally according to their weights. This demonstrates that differentiated QoS can be provisioned by applying WFQ to the ingress buffer, even with deflection routing.

5. Conclusions

Optical packet-switched (OPS) networks will likely carry Internet traffic in the future. In this paper, we have examined the performance of Internet protocols (TCP and UDP) over unslotted mesh OPS networks. In particular, we have studied the impact of packet aggregation, deflection routing, and ingress buffering as used in the underlying OPS networks on TCP and UDP performance. The performance metrics we are interested in are throughput, fairness, and delay jitter.

We have studied aggregation schemes of different aggregation granularities, namely, full aggregation, per-class aggregation, per-flow aggregation, as well as no aggregation. We have shown that packet aggregation in general improves TCP throughput, and the improvement increases with the aggregation interval (or optical packet size). Among the aggregation schemes, per-class aggregation and per-flow aggregation may significantly degrade TCP fairness at large aggregation intervals, due to synchronization problem with shared queueing. Therefore, certain fair queueing schemes are needed. We have demonstrated that differentiated QoS can be provisioned for both TCP and UDP traffic even with deflection routing, by implementing weighted fair queueing (WFQ) at the ingress buffer for different classes of traffic. Deflection routing avoids packet losses, but results in out-of-order packet delivery and increased packet delay jitter. We have shown that TCP throughput can be significantly improved by deflection routing in spite of the packet reordering, and the UDP packet delay jitter introduced by deflection routing can be alleviated by packet aggregation and ingress buffering. We have also shown that ingress buffering significantly improves TCP throughput and the ingress buffer only needs a small size (in terms of the number of optical packets).

Appendix A. Analysis of the aggregation benefit to TCP

To better understand the benefit of packet aggregation to TCP performance, we present in this section an approximate analysis. For simplicity, we do not consider deflection routing and focus on a single TCP flow. The system we consider is shown in Fig. 12, where s_0 is the TCP sender and d_0 is the receiver. The packets sent from s_0 are aggregated at r_0, and then forwarded to r_1 where they are segregated. Random losses are introduced to the link between r_0 and r_1.

Without packet aggregation, we use the same steady-state TCP model as used in [24]: a single packet is lost each time the congestion window *cwnd* is increased to W packets, and such packet losses are recovered through fast recovery and fast retransmit. We show in Fig. 13 by dashed lines this repeating pattern. In each cycle, *cwnd* increases from $W/2$ to W by one per RTT. The duration of each cycle is hence $W/2$RTT, and the number of packets sent in each cycle is $W/2+(W/2+1)+\cdots+W \approx \frac{3}{8}W^2$. The packet loss rate, denoted by p, is therefore

$$p = \frac{8}{3W^2}. \tag{A.1}$$

Alternatively, W can be expressed in p as

$$W = \sqrt{\frac{8}{3p}}. \tag{A.2}$$

Denote S the packet size. The TCP sending rate can hence be calculated as

$$T = \frac{\frac{3}{8}W^2 \cdot S}{\frac{1}{2}W \cdot \mathrm{RTT}} = \frac{\sqrt{3/2} \cdot S}{\mathrm{RTT} \cdot \sqrt{p}}. \tag{A.3}$$

With packet aggregation, we assume a similar steady-state model: a burst of n packets is lost each time *cwnd* is increased to W_a packets, and such packet losses are recovered through RTO mechanism, i.e., after RTO *cwnd* first increases exponentially from one to $W_a/2$ (i.e., slow start) and then increases linearly to W_a (i.e., congestion avoidance). We show in Fig. 13 by solid lines this repeating pattern. Each cycle consists of slow start, congestion avoidance and RTO. The duration of slow start is $\log_2(W_a/2)$RTT. The duration of congestion avoidance is $W_a/2$RTT. For simplicity, we assume RTO = RTT. The number of packets sent in each cycle is approximately $W_a + \frac{3}{8}W_a^2$. The packet loss rate, denoted by p_a, is

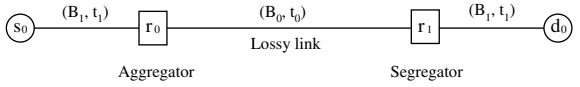


Fig. 12. A system with a single TCP flow and packet aggregation.

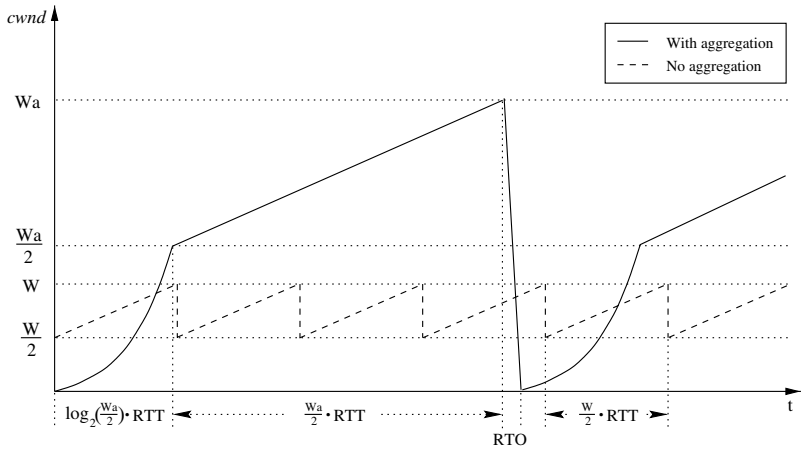


Fig. 13. TCP *cwnd* evolution with and without packet aggregation.

$$p_a = \frac{n}{W_a + \frac{3}{8}W_a^2}. \tag{A.4}$$

Solving this equation for W_a, we have

$$W_a = \frac{\sqrt{16 + 24n/p_a} - 4}{3}. \tag{A.5}$$

The TCP sending rate with packet aggregation is hence

$$T_a = \frac{\left(W_a + \frac{3}{8}W_a^2\right) \cdot S}{(\log_2(W_a/2) + W_a/2 + 1) \cdot \mathrm{RTT}} = \frac{\frac{n}{p_a} \cdot S}{\left(\log_2\left(\frac{\sqrt{4+6n/p_a}-2}{3}\right) + \frac{\sqrt{4+6n/p_a}-2}{3} + 1\right) \cdot \mathrm{RTT}}. \tag{A.6}$$

We define the *aggregation benefit*, β, as the ratio of the TCP sending rate between with aggregation and without aggregation when they have the same packet loss rate (i.e., $p_a = p$). We hence have

$$\beta = \frac{T_a}{T} = \frac{n\sqrt{\frac{2}{3p}}}{\log_2\left(\frac{\sqrt{4+6n/p}-2}{3}\right) + \frac{\sqrt{4+6n/p}+1}{3}}. \tag{A.7}$$

The size of the lost bursts n is a function of the aggregation interval Δ. Larger Δ leads to larger n. For simplicity, we assume a linear relationship as

$$n = c\Delta, \tag{A.8}$$

where c is a constant dependent on the bandwidth B_1 of the ingress link to r_0. Therefore, we have the relationship between β and Δ as

$$\beta = \frac{c\Delta\sqrt{\frac{2}{3p}}}{\log_2\left(\frac{\sqrt{4+6c\Delta/p}-2}{3}\right) + \frac{\sqrt{4+6c\Delta/p}+1}{3}}. \tag{A.9}$$

In the following, we present the simulation result and compare it with our analysis. The parameters in our simulation are $B_0 = 622$ Mbps, $B_1 = 155$ Mbps, $t_0 = t_1 = 10$ ms. The lossy link between r_0 and r_1 has a packet loss rate of $p = 0.01$.

We first show in Fig. 14 the size of the lost bursts n versus the aggregation interval Δ. From the simulation result we determine a fitting curve according to Eq. (A.8) with $c = 7$. With this value of c, we can compute β according to Eq. (A.9). The comparison of the analysis result and the simulation result is shown in Fig. 15. We can see that our simple model reasonably well captures the relationship between aggregation benefit β and aggregation interval Δ.

Note that this section is just aimed to reveal the mechanism of aggregation benefit using a simple model. The simulation results shown in Section 4.1 have different simulation settings (e.g., with

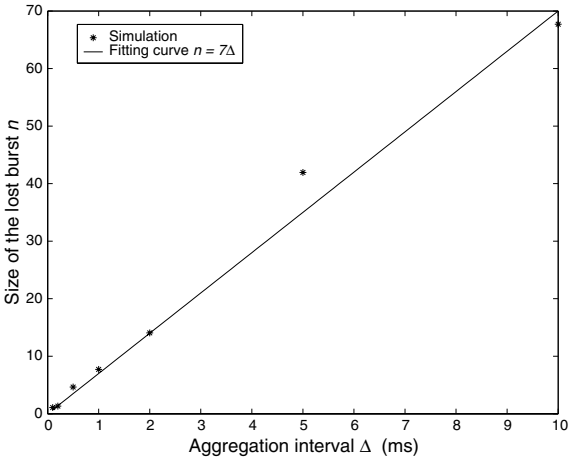


Fig. 14. The size of the lost bursts n versus the aggregation interval Δ.

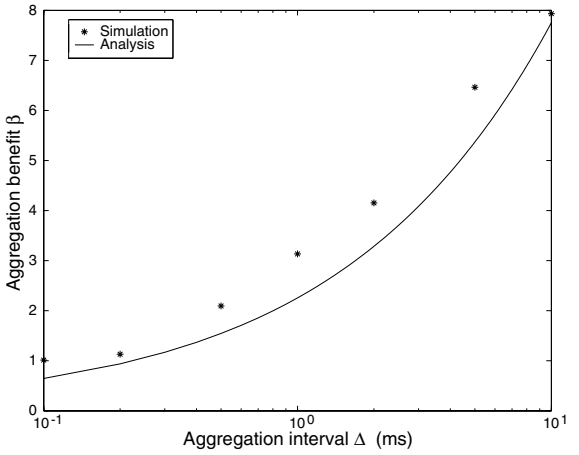


Fig. 15. Aggregation benefit β versus the aggregation interval Δ.

deflection routing, much higher traffic load and hence packet loss rate), and hence are not comparable with the analysis results here.

References

[1] M.J. O'Mahony, D. Simeonidou, D.K. Hunter, A. Tzanakaki, The application of optical packet switching in future communication networks, IEEE Communications Magazine 39 (3) (2001) 128–135.

[2] A. Jourdan, D. Chiaroni, E. Dotaro, G.J. Eilenberger, F. Masetti, M. Renaud, The perspective of optical packet switching in IP-dominant backbone and metropolitan networks, IEEE Communications Magazine 39 (3) (2001) 136–141.

[3] S. Yao, S.J.B. Yoo, B. Mukherjee, All-optical packet switching for metropolitan area networks: opportunities and challenges, IEEE Communications Magazine 39 (3) (2001) 142–148.

[4] S. Yao, B. Mukherjee, S. Dixit, Advances in photonic packet switching: an overview, IEEE Communications Magazine 38 (2) (2000) 84–94.

[5] S. Yao, F. Xue, B. Mukherjee, S.J.B. Yoo, S. Dixit, Electrical ingress buffering and traffic aggregation for optical packet switching and their effect on TCP-level performance in optical mesh networks, IEEE Communications Magazine 40 (9) (2002) 66–72.

[6] B.R. Badrinath, P. Sudame, Gathercast: the design and implementation of a programmable aggregation mechanism for the Internet, in: Proceedings of the Ninth International Conference on Computer Communications and Networks, 2000, pp. 206–213.

[7] H. Tounsi, L. Toutain, F. Kamoun, Small packets aggregation in an IP domain, in: Proceedings of the Sixth IEEE Symposium on Computers and Communications, 2001, pp. 708–713.

[8] A. Detti, M. Listanti, Impact of segments aggregation on TCP Reno flows in optical burst switching networks, in: Proc. IEEE INFOCOM'02, 2002, pp. 1803–1812.

[9] X. Cao, J. Li, Y. Chen, C. Qiao, Assembling TCP/IP packets in optical burst switched networks, in: Proc. IEEE GLOBECOM'02, November 2002, pp. 2808–2812.

[10] S. Gowda, R.K. Shenai, K.M. Sivalingam, H.C. Cankaya, Performance evaluation of TCP over optical burst-switched (OBS) WDM networks, in: Proc. IEEE ICC'03, May 2003, pp. 1433–1437.

[11] J. He, S.-H.G. Chan, TCP and UDP performance for Internet over optical packet-switched networks, in: Proc. IEEE ICC'03, May 2003, pp. 1350–1354.

[12] A.S. Acampora, S.I.A. Shah, Multihop lightwave networks: a comparison of store-and-forward and hot-potato routing, in: Proc. IEEE 1NFOCOM'91, April 1991, pp. 10–19.

[13] A.K. Choudhury, A comparative study of architectures for deflection routing, in: Proc. IEEE GLOBECOM'92, December 1992, pp. 1911–1920.

[14] S.-H.G. Chan, H. Kobayashi, Packet scheduling algorithms and performance of a buffered shufflenet with deflection routing, IEEE/OSA Journal of Lightwave Technology 18 (4) (2000) 490–501.

[15] F. Borgonovo, L. Fratta, J.A. Bannister, Unslotted deflection routing in all-optical networks, in: Proc. IEEE GLOBECOM'93, November 1993, pp. 119–125.

[16] T. Chich, J. Cohen, P. Fraigniaud, Unslotted deflection routing: a practical and efficient protocol for multihop optical networks, IEEE/ACM Transactions on Networking 9 (1) (2001) 47–59.

[17] S. Yao, B. Mukherjee, S.J.B. Yoo, S. Dixit, All-optical packet-switched networks: a study of contention–resolution schemes in an irregular mesh network with variable-sized packets, in: Proc. OptiComm 2000, November 2000, pp. 235–246.

[18] C.Y. Li, P.K.A. Wai, X.C. Yuan, V.O.K. Li, Deflection routing in slotted self-routing networks with arbitrary topology, in: Proc. IEEE ICC'02, April 2002, pp. 2781–2785.

[19] J.P. Jue, An algorithm for loopless deflection in photonic packet-switched networks, in: Proc. IEEE ICC'02, April 2002, pp. 2776–2780.

[20] J. He, S.-H.G. Chan, TCP performance with deflection routing in the Internet, in: Proc. IEEE ICON'02, August 2002, pp. 383–388.

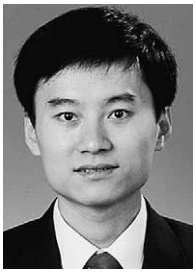
[21] C. Guillemot, M. Renaud, P. Gambini, C. Janz, I. Andonovic, R. Bauknecht, B. Bostica, M. Burzio, F. Callegati, M. Casoni, D. Chiaroni, F. Clerot, S.L. Danielsen, F. Dorgeuille, A. Dupas, A. Franzen, P.B. Hansen, D.K. Hunter, A. Kloch, R. Krahenbuhl, B. Lavigne, A. Le Corre, C. Raffaelli, M. Schilling, J.-C. Simon, L. Zucchelli, Transparent optical packet switching: the European ACTS KEOPS project approach, IEEE/OSA Journal of Lightwave Technology 16 (12) (1998) 2117–2134.

[22] Available from <http://www.isi.edu/nsnam/ns>.

[23] R. Jain, D. Chiu, W. Hawe, A quantitative measure of fairness and discrimination for resource allocation in shared computer systems, DEC Research Report TR-301, September 1984.

[24] S. Floyd, K. Fall, Promoting the use of end-to-end congestion control in the Internet, IEEE/ACM Transactions on Networking 7 (4) (1999) 458–472.

Jingyi He received the B.E. and M.E. degrees both in Optoelectronic Engineering from Huazhong University of Science and Technology, Wuhan, PR China, in July 1996 and June 1999, respectively. He is currently a Ph.D. student in the Department of Electrical and Electronic Engineering at The Hong Kong University of Science and Technology.



S.-H. Gary Chan received the M.S.E. and Ph.D. degrees in Electrical Engineering from Stanford University, Stanford, CA, in 1994 and 1999, respectively, with a minor in business administration. He obtained his B.S.E. degree (highest honor) in Electrical Engineering from Princeton University, Princeton, NJ, in 1993.

He is currently an Assistant Professor with the Department of Computer Science, The Hong Kong University of Science and Technology, Hong Kong, and an Adjunct Researcher with the Microsoft Research Asia in Beijing. He was a Visiting Assistant Professor in networking at the Department of Computer Science, University of California, Davis, from September 1998 to June 1999. During 1992–1993, he was a Research Intern at the NEC Research Institute, Princeton, NJ. His research interest includes multimedia networking, peer-to-peer multicast networks, high-speed and wireless communications networks, and Internet technologies and protocols. He was a William and Leila Fellow at Stanford University during 1993–1994. At Princeton, he was the recipient of the Charles Ira Young Memorial Tablet and Medal, and the POEM Newport Award of Excellence in 1993. He is a member of Tau Beta Pi, Sigma Xi, and Phi Beta Kappa.