Abstracts: Proposed in this paper is a novel two-stage packet assembly scheme for the reduction of self-similarity of traffic in optical packet switching (OPS) networks. Instead of one buffer in single-stage packet assembly, two buffers are used to assemble packets at each assembly queue in two-stage packet assembly. Based on a generic model of edge nodes, the proposed two-stage assembly is examined by simulations under self-similar IP input traffic. The simulation results show that, compared to single-stage assembly, two-stage assembly could achieve much more efficient reduction of the self-similarity of traffic at little cost of packetization efficiency and assembly delay.

Keywords: optical packet switching, packet assembly, self-similarity, packetization efficiency, assembly delay.

1. INTRODUCTION

During the last decade, numerous researchers have reported and confirmed the presence of self-similarity in diverse network traffic, such as Ethernet [1], wide-area networks [2] and so on. The self-similarity of traffic has considerable impact on queueing performance [3] and leads to persistent periods of congestion and heavily concentrated loss [4], increased single node delay and thus network delay [5]. In [6, 7], it is confirmed that the performance of an optical packet switching (OPS) network is also degraded dramatically under self-similar traffic load.

Recently, packet assembly implemented at edge nodes in OPS networks is employed to mitigate the deleterious effects of self-similarity. By aggregating multiple packets from client networks to create a larger optical packet for transmission through the OPS network, packet assembly can reduce the switching burden and overheads at the optical layer. More importantly, packet assembly designed properly can reduce the self-similarity of traffic as traffic shaper which is helpful to improve the end-to-end performance of TCP/IP networks [8]. The main challenge in designing packet assembly is to balance several opposing objectives: first, increasing the burst size, helpful to reduce the switching burden; second, maintaining acceptable assembly delay introduced during packet assembly; thirdly, reducing
the self-similarity in case of self-similar input traffic. Of various assembly algorithms, length-threshold assembly [9] and timer-based assembly [10] are two fundamental ones. The former has smaller variance for optical packet size and higher packetization efficiency but has little impacts on the self-similarity of traffic [11]. In comparison, the latter can efficiently reduce the self-similarity of traffic [11] but have scattered packet size distribution. Hybrid assembly [8, 12], a combination of the above two schemes, is a trade-off between assembly delay, packetization efficiency, and the reduction of the self-similarity. However, the all proposed assembly mechanisms are based on single-stage scheme, or one buffer at each assembly queue, and the optimum balance between the three performance parameters is very difficult to achieve.

In this paper, we have proposed a novel scheme, called two-stage packet assembly (TSPA), where two buffers are used to assembly packets at each assembly queue. Firstly, the first stage assembly buffer increases the packetization efficiency as much as possible by keeping assembly length-threshold dominant during assembly. Next, the self-similarity of traffic is reduced efficiently when the packets outputted from the first stage assembly buffer are further assembled by the second assembly buffer where assembly time-out interval is dominant. Our simulation results show that, compared to single-stage hybrid packet assembly, two-stage hybrid packet assembly could reduce efficiently the self-similarity of traffic at little cost of packetization efficiency and assembly delay.

2. OPS NETWORK AND TSPA SCHEME

In the paper the unslotted OPS network is considered which consists of edge nodes and core nodes, as depicted in Fig. 1. As adapters between electronic layer and optical layer, edge nodes undertake the task of assembling diverse client traffic such as IP packets, ATM cells
and Ethernet frames into larger optical packets before they are injected into the OPS core. At core nodes, packets are switched in optical domain.

Two-stage packet assembly (TSPA) is performed at the ingress edge node, as illustrated in Fig. 2, which comprises a classifier, multiple assembly queues, and a FIFO transmission queue. Client packets (e.g. IP packets here) arriving at the edge node are firstly classified and injected into different assembly queues according to their destinations and QoS classes. Suppose there are $N$ possible egresses, $N$ assembly queues are required in this node to buffer incoming IP packets. If services are to be classified into $M$ QoS classes, $M$ sub-queues are further required in each assembly queue. Here we consider only one QoS class (namely $M = 1$). In each assembly queue in TSPA mechanism, there are two assembly buffers: I stage assembly buffer and II stage assembly buffer, as indicated in Fig. 2. The two stages of assembly buffers employ the same hybrid assembly algorithm [12] but different assembly parameters respectively. Next, I stage assembly buffer in each assembly queue employs the hybrid algorithm where assembly length-threshold is dominant to generate temporary packets and send them to II stage assembly buffer. Then II stage assembly buffer applies the hybrid algorithm where assembly time-out interval is dominant to generate optical packets and an optical label is added to each optical packet. Ultimately, the generated optical packets are fed into a FIFO scheduler for transmission into the OPS core. After passing through the core network, the optical packets will be disassembled into the original IP packets. The operation of single stage packet assembly (SSPA) is similar to that of TSPA except that there are no II stage assembly buffers in SSPA.

The hybrid assembly algorithm utilizes both the assembly time-limit $T_{out}$ (the maximum waiting time) and the assembly threshold $L_{threshold}$ (the maximum assembled packet size) to trigger the generation of assembled packets. Upon arrival of the first IP packet to an empty assembly buffer, an associated time-out interval $T_{out}$ is set. As soon as time-out occurs or the size of all the collected IP packets exceeds the threshold of $L_{threshold}$ bytes, an assembled packet containing all IP packets in the buffer is generated.

3. SIMULATIONS AND RESULTS

3.1. Simulation Scenarios

Based on the proposed model of edge nodes in Section 2, simulations are conducted. Here we consider the edge node with five egresses and one QoS class, which requires five assembly queues. Furthermore, we assume that the destinations of the incoming IP packets are distributed uniformly among the five egresses. The input traffic of edge nodes is synthesized by Sup_FRP (the Superposition of the Fractal Renewal Point Process) self-similar traffic model [12], where 3 parameters are involved, i.e. Hurst parameter $H$, Fractal Onset Time Scale (FOTS), and average packet arrival rate. Hurst parameter $H$ defines the Hurst characteristic of the self-similar traffic source. Larger $H$ indicates the stronger property of self-similarity. FOTS defines the time scale from which the self-similar behavior begins to exhibit in the traffic. In our simulations, we adopt the parameter settings in [12] with the Hurst parameter $H$ of 0.8 and the FOTS of 0.1 ms respectively. The average packet arrival rate is set to 3,328,894.8 packets/s. Another important parameter of traffic source is packet size. The IP packet size follows a negative exponential distribution with the mean value of 375.5 byte and the maximum of 1500 bytes due to Ethernet’s maximum transfer unit
(MTU). Considering the average packet length of 375.5 bytes, the average packet arrival rate of 3,328,894.8 packets/s corresponds to the incoming data rate of 10 Gbits/s.

Table 1: Assembly parameter settings

<table>
<thead>
<tr>
<th>Assembly scheme</th>
<th>Assembly parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Two-stage hybrid assembly scheme</td>
<td>I stage assembly threshold</td>
<td>50 KB</td>
</tr>
<tr>
<td></td>
<td>I stage assembly time-limit</td>
<td>0.004 second</td>
</tr>
<tr>
<td></td>
<td>II stage assembly threshold</td>
<td>200 KB</td>
</tr>
<tr>
<td></td>
<td>II stage assembly time-limit</td>
<td>0.0012 second</td>
</tr>
<tr>
<td>Single-stage hybrid assembly scheme</td>
<td>Assembly threshold</td>
<td>200 KB</td>
</tr>
<tr>
<td></td>
<td>Assembly time-limit</td>
<td>0.0016 second</td>
</tr>
</tbody>
</table>

The assembly parameters of TSPA scheme, as listed in Table 1, are set to ensure that the assembly threshold is dominant during I stage assembly while II stage assembly is dominated by the assembly time-limit. The assembly parameter settings for the SSPA scheme are only convenient for the performance comparison between TSPA and SSPA, namely making their average assembly delay and average packetization efficiency equal.

3.2. Results and Discussion

Fig. 3 shows I stage and II stage assembly delay in the TSPA scheme. The I stage average assembly delay is about 0.4 ms, much less than the set I stage assembly time-limit, while the II stage one is almost constant, equal to the set II stage assembly time-limit. This assembly threshold is dominant during I stage assembly while II stage assembly is dominated by the assembly time-limit. The assembly parameter settings for the SSPA scheme are only convenient for the performance comparison between TSPA and SSPA, namely making their average assembly delay and average packetization efficiency equal.

Fig. 4: Assembly delay trace for (a) TSPA scheme; (b) SSPA scheme.
implies that the assembly parameter settings in TSPA are proper and ensure the assembly length-threshold and the assembly time-limit respectively dominant during I stage assembly and II stage assembly.

Fig. 4 and Fig. 5 presents respectively the assembly delay traces and the packetization efficiency traces for TSPA and SSPA. The calculated average assembly delay of TSPA is about 1.59 ms, as much as that of SSPA about 1.52 ms. In addition, Fig. 4(b) indicates that the assembly length-threshold and the assembly time-limit both took effects during SSPA in our simulations. Packetization efficiency, denoting how much optical packets are filled by IP packets, is defined as the ratio of the size of the generated optical packets over the set assembly threshold, which is 200 KB in the simulations. The calculated average packetization efficiency of the TSPA scheme is about 0.924, approximate to that of the SSPA scheme about 0.986.

Fig. 6 shows the input traffic trace, the output traffic trace for TSPA scheme, and the output traffic trace for SSPA scheme. All the traffic processes were measured based on the time-scale of 0.002 second in the simulations. A key feature of self-similar traffic is that it exhibits the strong property of long-range dependence (LRD) which can be captured by Hurst parameter $H$ [13]. In the case $0.5 < H < 1$, long range dependence occurs and the process exhibits self-similarity. A larger $H$ implies a higher degree of self-similarity. In this study, the Abry-Veitch wavelet-based estimator [14], whose vanishing moment in our estimation was set to 3, was chosen to estimate the Hurst parameters of the traffic traces. The Hurst estimations of the input, the output after TSPA, and the output after SSPA are respectively 0.805, 0.541, and 0.785.
Table 2: Performance comparison between TSPA and SSPA

<table>
<thead>
<tr>
<th>Assembly Scheme</th>
<th>SSPA</th>
<th>TSPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average packetisation efficiency</td>
<td>0.986</td>
<td>0.924</td>
</tr>
<tr>
<td>Average assembly delay</td>
<td>1.52 ms</td>
<td>1.59 ms</td>
</tr>
<tr>
<td>Reduction of the self-similarity</td>
<td>1.5%</td>
<td>67.4%</td>
</tr>
</tbody>
</table>

The performance parameters of SSPA and TSPA are summarized in Table 2. It is confirmed that, compared to the SSPA scheme, the TSPA scheme could achieve more efficient reduction of the self-similarity of traffic at little cost of packetization efficiency and assembly delay under self-similar input traffic. This is because two-stage assembly structure in TSPA eases the restriction between the assembly threshold and the assembly time-limit during the packet assembly and thus takes full advantage of length-threshold assembly and timer-based assembly. Under dynamic traffic load, the assembly time-limit parameters in TSPA should be normalized by the average interarrival of input traffic and thus the input traffic has little effect on the performance of TSPA.

4. CONCLUSION

In this paper, we have proposed a novel two-stage packet assembly scheme for the reduction of self-similarity of traffic in optical packet switching (OPS) networks. Instead of one buffer in single-stage packet assembly, two buffers are used to assemble packets at each assembly queue in two-stage packet assembly. Under the self-similar IP input traffic, compared to single-stage assembly, the proposed two-stage assembly scheme could achieve much more efficient reduction of the self-similarity of traffic at little cost of packetization efficiency and assembly delay. The proposed two-stage packet assembly scheme can also be applicable in optical burst switching network.

ACKNOWLEDGEMENT

This work is supported in part by national Natural Science Foundation Project of China (contract No. 60372100) and in part by ‘863’ High Technology Research and Development Program of China (contract No. 2003AA122530).

REFERENCE
