United assembly algorithm for optical burst switching

Jinhui Yu (于金辉), Yijun Yang (杨毅军), Yuehua Chen (陈月华), and Ge Fan (范 戈)

National Laboratory on Local Fiber-Optic Communication Networks & Advanced Optical Communication Systems, Shanghai Jiaotong University, Shanghai 200030

Received October 22, 2002

Optical burst switching (OBS) is a promising optical switching technology. The burst assembly algorithm controls burst assembly, which significantly impacts performance of OBS network. This paper provides a new assembly algorithm, united assembly algorithm, which has more practicability than conventional algorithms. In addition, some factors impacting selections of parameters of this algorithm are discussed and the performance of this algorithm is studied by computer simulation.

OCIS codes: 060.4510, 060.4250.

Recently, optical burst switching (OBS)[1,2] is proposed to support next generation optical Internet. OBS networks consist of multiple edge nodes and core nodes, which are connected by wavelength division multiplexing (WDM) links. At the ingress edge node, multiple packets that have the same destination and quality of service (QoS) requirements are assembled into a burst. The ingress edge node also assigns a control packet for each burst. During burst assembly, the assembly algorithm determines the characteristics of the generated bursts, and affects the performance of OBS network. So it needs to take account of all requirements of the OBS network, such as end to end delay, channel utilization and switching efficiency, etc. However, current burst assembly algorithms based on assembly time consider parts of above requirements and set only minimum burst length[3] or maximum burst length[4], so those cannot completely control burst assembly according to requirements of the OBS network. In order to solve this problem, we propose a new assembly algorithm, united assembly algorithm (UAA), which controls burst assembly more completely by setting three parameters: assembly time T_a, minimum burst length $L_{\min}$, and maximum burst length $L_{\max}$.

Let T_c be the value of the timer for burst assembly and l_b the total size of all packets in the assembler. The UAA can be described as follows.

1) When a packet with length of l_p (in bit) arrives to the assembler:

if ($l_b = 0$)

{$T_c = 0$; $l_b = l_p$;}

else if ($l_b + l_p < L_{\max}$)

$l_b = l_b + l_p$;

else

{report the generation of a burst with length l_b; $T_c = 0$; $l_b = l_p$;}

2) When $T_c = T_a$

if ($l_b < L_{\min}$)

{pad this burst to length $L_{\min}$, then report the generation of a burst with length $L_{\min}$; $l_b = 0$;}

else

{report the generation of a burst with length l_b; $l_b = 0$;}

As a key parameter in the UAA, T_a is crucial to shorten edge delay[5], especially for real-time traffic, that is to say there is a maximum requirement for the assembly time. As for $L_{\max}$, it mainly reflects requirement for assembler overflow[6]. During burst assembly, a packet will lose when its arrival makes the burst length exceed the maximum capacity B of the assembler. In the UAA, $L_{\max}$ ($L_{\max} < B$) is used to minimize probability of assembler overflow.

In the UAA, $L_{\min}$ reflects requirements for congestion on control channels, channel utilization, bottleneck at core nodes and switching efficiency. The former two factors are discussed in Ref. [4]. So we focus on the latter two factors. At the core node, bottleneck caused by processing time for the control packet or configuring time for the switching fabric limits $L_{\min}$. Let δ be the maximum processing time cost by all components at the core node, and t_{sw} the configuring time, the requirement for burst arrival rate is $\lambda_b \le \min\{1/\delta, 1/t_{sw}\}$. Let ρ be the data channel utilization, we have the average burst length (in μs) $L_b = \rho(C-c)/\lambda_b$, and then the minimum burst length (in μs) will be

$$L_{\min} = \rho(C-c)\cdot\max\{\delta, t_{sw}\}. \quad (1)$$

In general, δ is less than 1 μs, while t_{sw} may be tens of microseconds. Therefore, $L_{\min}$ is mainly determined by t_{sw}.

$L_{\min}$ is also determined by requirement for efficiency of OBS, η, which is defined as $L_b/t_{\rm proc}$, where $t_{\rm proc}$ is the average duration for processing a control packet at the core node. Actually, η presents the gross data switched by one operation at the core node. An OBS network should have a minimum requirement for η, so it sets a lower bound for the average burst length.

The burst characteristics and the performance of the UAA are studied via computer simulation. Here, we consider four key performance issues: the ratio of times beyond maximum burst length to the number of bursts, the ratio of times below minimum burst length to the number of bursts, padding ratio defined as ratio of all padding bits to the total size of all bursts, and mean burst length.

In our simulation, a simple model is used, which consists of a traffic source and an assembler. A self-similar IP packet source is used to provide incoming traffic for the assembler, and the Hurst parameter of the traffic is 0.8. IP packet length is exponentially distributed with average length of 6560 bits. The incoming traffic load is 0.7, and the link rate is 2.5 Gbit/s. The assembler uses the UAA for burst assembly. Without loss of generality, we select $L_{\min} = 20$ kbit and $L_{\max} = 100$ kbit from foregoing discussion.

The interaction of the assembly time and the burst length limitations is studied, as shown in Fig. 1, where the assembly time is from 10 – 60 μs. From Fig. 1, we found that the ratio beyond $L_{\max}$ increases with increasing of the assembly time and the ratio below $L_{\min}$ decreases as the assembly time increases, and there is an over point at about 30 μs. In addition, as shown in Fig. 1, the larger Hurst parameter (denoted by Hurst in all figures) the incoming traffic has, the higher the ratio beyond $L_{\max}$ is. On the contrary, a lower ratio below $L_{\min}$ is got when the incoming traffic has a larger Hurst parameter. This is because the traffic with larger Hurst parameter exhibits more bursty, and it can provide more packets for a burst.

Effect of the assembly time on padding ratio is studied next. The results are shown in Fig. 2, where the padding ratio decreases with increasing of the assembly time. This trend accords with that in Fig. 1. So the padding ratio should be minimized by increasing the assembly time. However, also seen from Fig. 1, increasing the assembly time will induce higher ratio of beyond $L_{\max}$ that means larger probability of assembler overflow. Therefore, there is a tradeoff between padding ratio and assembler overflow when the assembly time is adjusted. From Figs. 1 and 2, 30 μs is a balanced value for the assembly time on above simulation conditions. Furthermore, the average burst length is obtained. As shown in Fig. 3, it increases with increasing of not only the assembly time but also the Hurst parameter of the incoming traffic. This confirms the above analysis for the padding ratio.

Figures 4 and 5 show the probability density function (PDF) of burst length when the UAA and the conventional assembly algorithm only setting $L_{\max}$[4] are used respectively. In both figures, the assembly time is set to be 20, 40, and 60 μs, respectively, and the Hurst parameter of the incoming traffic is 0.8. Comparing Fig. 4

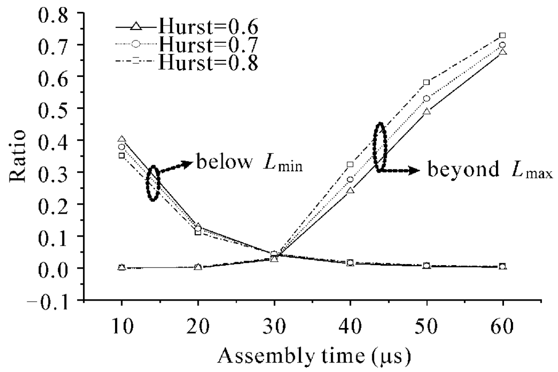

Fig. 1. The interaction of the assembly time and burst length limitations.

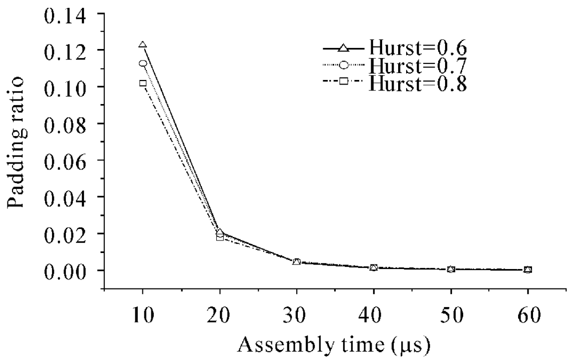

Fig. 2. Padding ratio versus assembly time.

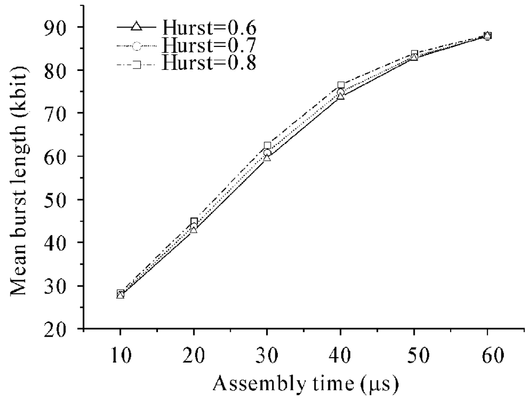

Fig. 3. The assembly time versus the average burst length.

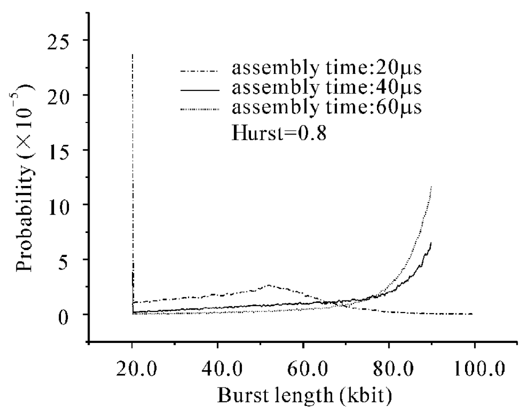

Fig. 4. Distribution of burst length using the UAA.

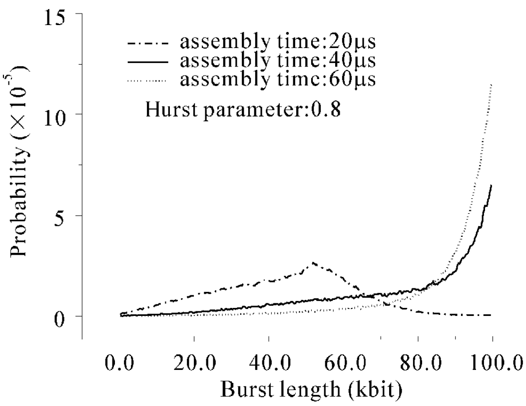

Fig. 5. Distribution of burst length using the conventional algorithm only setting $L_{\max}$.

with Fig. 5, we find a difference when the burst length varies from 0 to 20 kbits. To the UAA, length of all bursts is beyond 20 kbits because of $L_{\min}$. Whereas, seen from Fig. 5, there are some bursts whose length is less than 20 kbits to the considered conventional algorithm, and number of these bursts increases when the smaller assembly time is used. This difference results from the lower bound for burst length in the UAA.

In this paper, we propose a new burst assembly algorithm, UAA, for OBS. The algorithm sets upper and lower bounds for burst length, and can adjust burst characteristics by the assembly time. It more completely reflects requirements of OBS networks than the conventional algorithms. The performance of the UAA is studied. The results show that the interaction of the assembly time and burst length limitations decide the burst characteristics in UAA.

J. Yu's e-mail address is jinhui_yu@sjtu.edu.cn.

References

1. L. Xu, H. G. Perros, and G. Rouskas, IEEE Communications Magazine **39,** 136 (2001).
2. C. Qiao and M. Yoo, J. High Speed Networks **8,** 69 (1999).
3. A. Ge, F. Callegati, and L. S. Tamil, IEEE Communications Letters **4,** 98 (2000).
4. Y. Xiong, M. Vandenhoute, and H. C. Cankaya, IEEE J. Selected Areas in Communications **18,** 1838 (2000).
5. S. Verma, H. Chaskar, and R. Ravikanth, IEEE Network **14,** 48 (2000).
6. D. Wischik, http://www.statslab.cam.ac.uk /~djw1005/Stats/Talks/obscap.pdf, Mar. 2001.