

On Optical Burst Switching and Self-Similar Traffic

An Ge, Franco Callegati, and Lakshman S. Tamil, *Member, IEEE*

Abstract—In this letter we consider burst switching for very high speed routing in the next generation Internet backbone. In this scenario, Internet Protocol (IP) packets to a given destination are collected in *bursts* at the network edges. We propose a burst assembly mechanism that can reduce the traffic autocorrelation or degree of self-similarity, and at the same time keep the delay due to burst formation limited at the network edges.

Index Terms—Burst switching, high-speed networks, optical routing, self-similar traffic.

I. INTRODUCTION

THE EVER increasing demand for transmission bandwidth, driven by the growth of the Internet and related services, is stretching the capabilities of the existing Internet backbones. The advent of wavelength-division multiplexing (WDM) systems provides a platform to exploit the potential huge capacity of the optical fibers and satisfy this bandwidth demand. The challenge now is to combine the advantages of WDM with emerging all-optical packet switching capabilities to yield optical routers capable of guaranteeing a fully optical data path and throughputs in the hundreds of terabits/second range [1], [2].

The basic idea underlying the development of an optical router is to decouple completely the data path from the control information (header) path. The data path will be fully optical throughout the network, with no conversion and storage of the data in any of the nodes. This should offer very high speed, direct interfacing with the fiber and good network scalability (the network subblocks are not dependent on the bit rate of the data path).

To make the decoupling possible and effective, it is necessary to develop a new protocol for the data transfer with the following characteristics:

- control information should travel on a different time window or preferably on a different wavelength channel and be managed independently from the data; that is not easily feasible with normal Internet Protocol (IP) packet format;
- packets should be at least several kilobytes long in order to be comparable with the time needed to process the header and perform the control functions (including forwarding

decision, switching etc.); that is not the case in the present Internet.

The former issue requires definition of a new packetization protocol as well as a burst control packet (BCP) format. The latter issue requires that several short IP packets be gathered into a single data unit, to be switched as a whole. To date there are two approaches discussed in the literature that address these issues:

- fixed length packets (tailored to the time requirement of the system) with synchronous, “ATM like” node operation, as proposed in the optical transparent packet network studied in the ACTS Project KEOPS [2], [3];
- variable length “super packet,s” also named *bursts*, with asynchronous node operation, as proposed in the “burst switching” approach described in [4]–[6].

In both cases, at the edges of the core network, or at the boundaries of each router, it is necessary to implement a translation function from normal IP packets to the new format and vice versa. In this letter we propose an algorithm to implement this function for the case of burst switching and show its effects on the traffic shape and on the burst assembly delay at the network edges.

II. BUILDING BURSTS FOR OPTICAL BURST SWITCHING

The network architecture considered is a cloud of *optical core* routers, organized as a flat mesh, with *edge routers* at the edges of the cloud that are responsible for the burst assembly/disassembly function. The core routers perform the function of forwarding the burst within the cloud to the proper destination edge router.

The core router inlets and outlets are optical fibers operated in WDM mode with 16 or more wavelengths in order to guarantee a high degree of statistical multiplexing gain [4]. The transmission rate per wavelength is assumed to be ranging from OC48 (STM16) to OC192 (STM64) and beyond.

At the edges of the burst switch cloud, the edge routers assemble bursts by merging IP packets and label each burst with a control packet (BCP) containing information regarding source, destination, quality of service, etc. In the architecture considered, the BCP is transmitted at a different data rate than the burst and on a separate wavelength channel.

To keep the load on the control channel that carries the BCP's under control, the bursts must be bigger than a given minimum b . On the other hand, gathering enough packets to make a burst of length b may take a very long time if the load on the input line is temporarily very low. Therefore a limit has to be placed on the waiting time of the packets in the burst assembly. The algorithm proposed here to perform this function works as follows:

Manuscript received April 16, 1999. The associate editor coordinating the review of this letter and approving it for publication was Prof. C. Douligeris.

A. Ge is with Alcatel Corporate Research Center, Richardson, TX 75801-1936 USA (e-mail: andrew.ge@aud.alcatel.com).

F. Callegati is with D.E.I.S., University of Bologna, 40136 Bologna, Italy.

L. S. Tamil was with Alcatel Corporate Research Center, Richardson, TX 75801-1936 USA. He is now with Yotta Networks, Inc., Richardson, TX 75081 USA (e-mail: ltamil@gt.net).

Publisher Item Identifier S 1089-7798(00)02951-3.

- 1) a logical queue is associated with each destination at the edges of the burst switch cloud (let us number the possible destinations using index i , $i \in [1 : M]$);
- 2) a time counter T_i is started any time a packet arrives directed to destination i and queue i is empty;
- 3) when T_i reaches the window threshold W_i , a burst is created and queued for transmission on the data channel: if the length of the burst is less than b , then it is padded to b ;
- 4) T_i is reset to 0 and it remains so until the next packet arrival to queue i .

This algorithm is very simple; nevertheless:

- fulfills the minimum size requirements on bursts;
- puts an upper bound on the delay experienced by packets due to burst assembly.

Furthermore, as we shall show in the following, the burst assembly algorithm shapes the traffic, making it more random than the input IP packets that make up the burst. This is a very interesting and useful feature and will be addressed in the remainder of this letter.

Let us start from the characteristics of existing Internet traffic. Considerable effort has been devoted in the last few years to the analysis of real traffic traces from the Internet. As the early works in [7] showed, the Internet traffic has fractal characteristics and can be described in terms of self-similar stochastic processes.

This feature is a significant drawback in terms of queuing performance. Simulations and analysis of queuing systems loaded with self-similar traffic have shown that the queuing performance is in general much worse than that obtained with short range dependent (random) traffic [8]. For this reason it is widely recognized that traffic self-similarity is a significant problem as far as network engineering is concerned. Hence, any reduction in the degree of self-similarity will be greatly beneficial.

This was one of the main motivations behind the present study. The basic idea is: may the burst assembly function be used to reduce the traffic self-similarity, and, if yes, to what extent? A positive answer to this question would mean that the burst switch backbone network could rely on limited congestion and small buffers could be enough to guarantee good quality of service [8].

III. SIMULATION MODEL

We simulated the burst assembly mechanism when fed by self-similar traffic and tested the traffic behavior at the output. The input traffic was generated according to [9], by multiplexing a large set of ON/OFF sources with ON and OFF periods independently and identically distributed according to a Pareto distribution. The Pareto distribution has a cumulative distribution of the form $F(x) = 1 - 1/(x^\alpha)$, where α is a parameter that decides the heaviness of the distribution tail. For an infinite number of sources, this kind of traffic results in a self-similar process with the Hurst parameter $H = (3-\alpha)/2$. When a high but not infinite number of sources is used in simulations, the resulting traffic is an approximation of a purely self-similar process.

We also assumed that the final destination for each packet was randomly chosen among M edge routers or possible destinations within the burst switch cloud. Each arrival from the

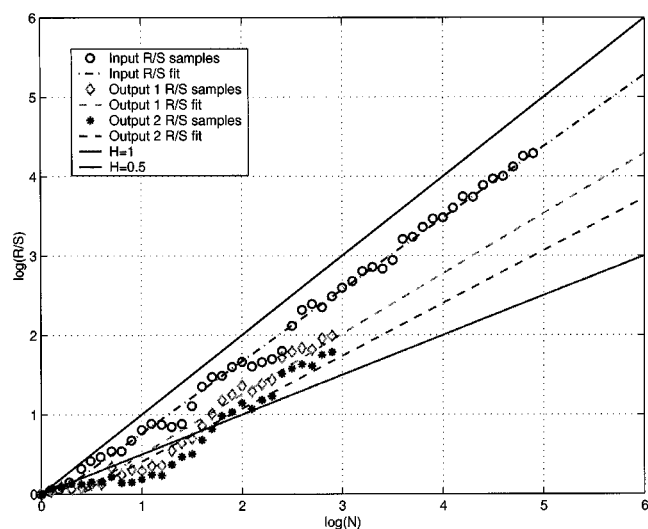


Fig. 1. R/S log / log plot diagram for input and output traffic from the burst assembly function: \diamond for a window threshold of 25 times the average packet length and $*$ for a window threshold of 50 times the average packet length.

merged traffic was given a destination drawn randomly, and it was then queued in the queue associated with that destination in the edge router. Bursts were then assembled according to the previously explained algorithm and queued in a common FIFO queue for transmission on the data link.

The aim of the simulations is to show that the burst assembly function can reduce the degree of self-similarity of the traffic. This has been accomplished by estimating the Hurst parameter of the input traffic H_{in} and of the output traffic H_{out} by means of the logarithmic R/S plots. Reference [7] gives details on the properties and characteristics of the R/S plot. Here R is the rescaled sample variability and S^2 is the sample variance. For a self-similar process, $\log(R/S) \sim H \log(N)$ for large N , where N is the number of samples. When plotting $\log(R/S)$ versus $\log(N)$, the slope of the least-square fitted line will give us an estimate of the Hurst parameter (H). The choice of the R/S plot is motivated by its easy visual interpretation. Even though this is not the most accurate tool to estimate the Hurst parameter (H), it is easily readable and distinguishes clearly the different traffic traces.

IV. RESULTS

Fig. 1 shows the R/S logarithmic plots for two different simulations. We assume that the output data rate is 16 times the input data rate and the utilization is 0.8. Each ON/OFF source follows the Pareto distribution with $\alpha = 1.2$ for both ON and OFF periods. This gives a load for each source equal to 0.5 and $H_{in} = 0.9$. From the R/S plots, we measure the aggregated input traffic to have a Hurst parameter equal to 0.904, which is very close to the theory.

Due to the nature of the burst assembly mechanism, the minimum burst length is longer than b , and the spacing related to the window threshold W_i . Therefore, the R/S plot for the output traffic would be shifted to the right relative to the one for the input traffic (whose packet sizes and interarrival times have a smaller granularity). In order to provide graphical comparison,

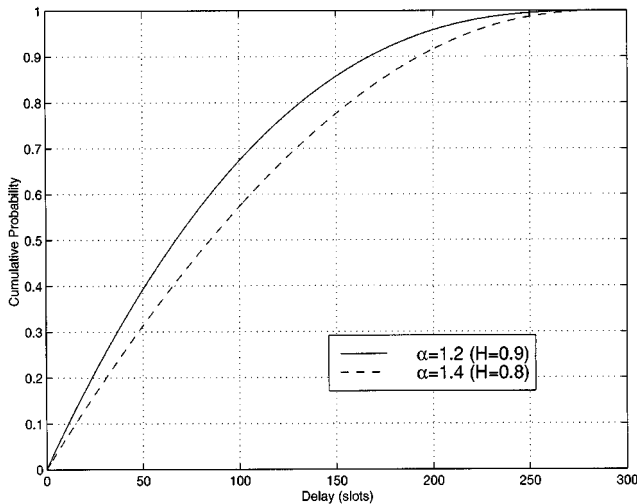


Fig. 2. Cumulative distribution function of the packet delay at the edge router due to the burst assembly function, for two different values of the Hurst parameter (solid line $H_{in} = 0.9$ and dashed line $H_{in} = 0.8$).

in the same plot we have renormalized the output plot to have the same starting point as in the input plot. This is just a translation that does not affect the slope of the curve and hence the estimate of H_{out} .

The two output samples in the figure use the same parameters, except for the window thresholds that take different values. The values of W_i for these two samples are equal to 25 and 50 times the average packet length, respectively. The results suggest that the burst assembly mechanism reduces the Hurst parameter of the traffic. This can be easily seen from the output samples of the graph. An estimate of H_{out} gives the values 0.76 for the first case and 0.65 for the second case. In both cases the reduction is significant when compared to $H_{in} = 0.904$, the Hurst parameter of the input traffic. Also the results indicate that the longer the window threshold, the greater the reduction of self-similarity. Of course the price we pay is the additional delay due to the burst assembly function.

The delay characteristics of different self-similar traffic have also been evaluated. We expect that the degree of correlation in the input traffic trace will affect the delay performance of the assembly process, so that highly correlated traffic will experience smaller delays. Intuitively this can be understood by noting that a higher correlation means a higher traffic burstiness, that

would eventually lead to a smaller delay in the burst assembly function. Fig. 2 shows the cumulative distribution of the delay in the burst assembly for input traffic with $H_{in} = 0.8$ and 0.9 . The curves behave as expected. This is another important feature of the burst assembly process and allows us to conclude that the burst assembly could be dimensioned using a simple random traffic model, as far as delay is concerned.

V. CONCLUSIONS

In this letter, we presented a simple burst assembly algorithm which reduces the self-similarity of traffic in an IP backbone network. It is shown that by properly setting the starting point and the duration of the time window, the traffic can be shaped. From the simulations, we found that the average delay due to the burst assembly algorithm is bounded by the minimum length of the burst and the window threshold and depends on the burstiness of the traffic.

ACKNOWLEDGMENT

The authors thank L. Tančevski for providing the traffic simulator.

REFERENCES

- [1] B. S. Arnaud, "Architectural and engineering issues for building an optical internet," in *Proc. SPIE Conf. on All-Optical Networking*, Boston, MA, Nov. 1998, pp. 358–377.
- [2] F. Callegati, M. Casoni, C. Raffaelli, and B. Bostica, "Packet optical networks for high speed TCP-IP backbones," *IEEE Commun. Mag.*, pp. 124–129, Jan. 1999.
- [3] P. Gambini *et al.*, "Transparent optical packet switching: Network architecture and demonstrators in the KEOPS project," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1245–1259, Sept. 1998.
- [4] J. S. Turner, "Terabit burst switching," *J. High Speed Networks*, vol. 8, no. 1, pp. 3–16, 1999.
- [5] M. Yoo and C. Qiao, "Just-enough-time: A high-speed protocol for bursty traffic in optical networks," in *Proc. IEEE Summer Topic Meeting*, Aug. 1997, pp. 26–27.
- [6] C. Qiao and M. Yoo, "Optical burst switching (OBS) - A new paradigm for an optical internet," *J. High Speed Networks*, vol. 8, no. 1, pp. 68–84, 1999.
- [7] W. Leland, M. Taqqu, and D. Wilson, "On the self-similar nature of ethernet traffic (extended version)," *IEEE/ACM Trans. Networking*, vol. 2, pp. 1–15, Feb 1994.
- [8] L. Tančevski, A. Ge, G. Castanon, and L. S. Tamil, "A new scheduling algorithm for asynchronous, variable length IP traffic incorporating void filling," in *Proc. OFC '99*, San Diego, CA, Feb. 23–25, 1999, ThM7.
- [9] M. S. Taqqu, W. Willinger, and R. Sherman, "Proof of a fundamental result in self-similar traffic," *Computer Commun. Rev.*, vol. 27, no. 2, pp. 5–23, 1997.