Issues on Performance Assessment of Optical Burst Switched Networks: Burst Loss Versus Packet Loss Metrics

Nuno M. Garcia[1,2], Przemyslaw Lenkiewicz,
Paulo P. Monteiro[2], Mário M. Freire[1]

[1] Universidade da Beira Interior, Department of Informatics,
6200-001 Covilhã, Portugal
[2] Siemens SA, Information and Communication, RD1, Research
2920-093 Amadora, Portugal
{nuno.mgarcia, paulo.monteiro}@siemens.com
przemek.lenkiewicz@gmail.com, mario@di.ubi.pt

Abstract. With the increasing interest in optical burst switching (OBS) networks, the performance assessment of this kind of networks became of particular concern. Recently, some authors suggested that burst loss was not a reliable performance assessment metric for OBS networks. Refuting this claim, this paper presents simulation results obtained for a ring network, using real tributary IPv4 packets as source for the burst assembly. It is shown that burst loss, packet loss and byte loss lead to similar results over a wide range of burst assembly scenarios and network loads, using different resource reservation schemes. Therefore, burst loss is a reliable metric and can be used for evaluation of performance of optical burst switched networks, when realistic burst assembly algorithms are considered over real traffic.

1 Introduction

Burst Switched networks were initially proposed by Amstutz in 1983 [1] as a way to benefit from the statistical multiplexing effect, or as initially described, benefit from "improved bandwidth efficiencies". This concept was later re-introduced in Optical Networks, contributing to the Optical Burst Switching (OBS) Network paradigm, initially proposed by Qiao and Yoo around 1999 [2]. When referring to Optical Burst Switching, three major assembly algorithms are used: *time constrained*; *size constrained*; both time / size constrained, also termed the *hybrid algorithm*. Bursts are created by aggregating packets into a larger data entity, which, after being transmitted, must be disassembled at the end node, and its constituent packets forwarded to their ultimate destination.

Burst switched networks performance is often measured in terms of burst loss or burst drop ratio. Recently, [3] proposed that burst loss was not equal to packet loss and these values vary within the same range. Research activities described in this paper show different results for several assembly scenarios, and particularly, that there

exists an equivalence relation between burst loss and packet loss, although the latter is of more interest to the end user than the former.

The remainder of this paper is organized as follows: Section 2 discusses basic assumptions and briefly describes the assembly algorithms implemented in the simulator. Section 3 is devoted to the simulation of the burst assembly process. Section 4 discusses the role of burst loss versus packet loss metrics in OBS networks. Section 5 presents main conclusions.

2 Basic Assumptions and Burst Assembly Algorithms

Data packet assembly is a process in which individual data packets are grouped together before the resulting burst is sent into the network structure. These packets may experience re-encapsulation (or not, depending on the network scenario) and typically the nature and origin of the data packets under consideration is not relevant to the assembly principle, as these may be Ethernet frames, ATM cells, IP packets, and so forth. The assembly process requires only the other end of the transmission link to run a complimentary burst disassembly process, retrieving the original constituent packets. In this study, IPv4 packets were used and no encapsulation of the aggregated packets was performed. We can expect IPv6 traffic to output equivalent results, following the research presented in [4]. The disassembly mechanism should thus consider the first 20 bytes of the data burst to be an IPv4 header, and proceed to extract that packet from the aggregated data. This step is repeated until no data is left within the burst. If the network implements burst segmentation techniques, the last readable packet may be corrupt, and if so, it is discarded.

Packet used in the simulation are real IPv4 packets, recorded from NLANR and obtained in [5]. This data is presented in files that record data packet traces in a ***time stamped header*** (*tsh*) format, shown in Fig. 1. The *.tsh* file format stores the payload stripped data packets, time stamped at their acquisition. The typical IPv4 data header is extended by application of the timestamp field (4 bytes for second timestamp and 3 bytes for microsecond timestamp), expressing the timestamp of the captured data packet relative to the 1st of January 1970. The *tsh* record also contains TCP information, comprising Source/Destination ports, Sequence/ Acknowledgment numbers and other TCP specific information. The standard format of the *.tsh* data packet header is shown in Fig. 1.

In order to assure IP address security, the Source and Destination Addresses disclosed in the IP *.tsh* packet header section are hashed to preserve the anonymity of the original machines. However, the IP hashing algorithm [6] is designed in such a manner that it preserves the IP address space density, thus class A servers shall always have lower hashed IP number than class D machines. The source code for the IP address hashing procedure is available from the NLANR website. Packet payload is not recorded. Issues on addresses are important because burst assembly is primarily performed in a “by destination” basis. The simulation handled the computation of the destination addresses for the bursts based on the destination address in the packet *tsh* data as follows: when an address was extracted from the packet, it was looked up in an

address table. This address table contains two entries – the first is the IP address itself, the second is the pseudo-address of the destination machine, which is to perform the final disassembly of the burst. If the extracted IP address is not yet present in the address table, then a random pseudo-address is assigned to it as its destination, and this pair was added to the table. This way, the full initial address space was homogeneous and randomly distributed over the available pseudo-addresses of the destination machines. This task is repeated in each node, as a way to closely mimic the hash of the initial IP address space. As an example, while hashed address *12345* processed in node *A* refers to destination machine *X*, it may refer to destination machine *Y* when the same file is processed by node *B*.

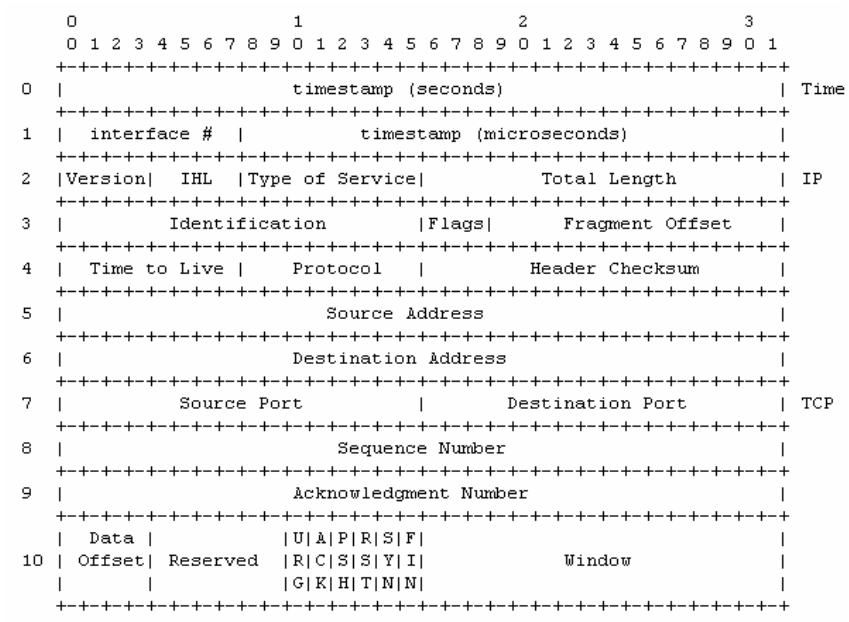

Fig. 1 - Internal format of the .tsh data packet format from NLANR.

The network topology simulated was a four node ring, of nodal degree 2. Shortest path routing was used and full wavelength conversion was assumed for the OBS simulation, using JIT [7] and also the JET [7] signaling protocols. JIT is an immediate reservation protocol and does not perform void filling, and thus every burst is treated independently of its size. On the other hand, JET is burst size sensitive as it performs delayed reservation and attempts void filling, so burst size is important to maximize the efficiency of network resource reservation.

The topology and the remaining default simulation parameters are not relevant for the focus of this research, as a change in these would only alter the performance of the network in terms of burst loss ratios. The simulation was performed with a large set of parameters to allow a wide range of loss ratio values, and thus test the possible correlations of burst and packet loss over the whole counter-domain.

The assembly of packets follows a specific assembly algorithm. Assembly algorithms are constraint driven, and fall into three categories:

1) Maximum Burst Size (MBS)
2) Maximum Time Delay (MTD)
3) Hybrid Assembly (HA)

Other assembly algorithms, like the ones considering classes of services, build upon one of the aforementioned basic types. In this study no CoS (Class of Service) was considered, mainly because the ToS (Type of Service) field in IPv4 packets does not bear reliable information. This limitation could have been overcome by assigning a given packet to a CoS, according to a pre-defined random distribution, but this would not add to the expected conclusions of this research, so no action was taken.

In the MBS assembly algorithm, the incoming data packets are aggregated consecutively into a burst, until its size exceeds the defined threshold. When this occurs, the last data packet overflowing the current burst will start a new one, while the current burst is transmitted into the network structure.

The MTD assembly algorithm was devised to prevent situations where, while using the MBS algorithm, the rate of incoming packets is so low or the arriving packets are so small, that it takes an unacceptable amount of time to fill up a single burst, resulting in excessive transmission delay for the aggregated packets. The MTD algorithm checks for the time difference between the head packet in the burst and the current local time. The burst is sent into the network as soon as that time difference exceeds the maximum delay time defined, independently of the size of the burst and of the number of packets it contains.

If the traffic flow rate is too high or the incoming packets are big, the MTD algorithm may end up aggregating bursts that are too big. In order to prevent such a situation, a HA algorithm was devised. In this assembly scheme, both thresholds – time and size – are considered simultaneously. Incoming packets are aggregated into the burst until either one of the threshold conditions is met. If an incoming packet overflows the burst size threshold, then the burst is close and this packet start a new burst.

3 Burst Assembly Simulation

The algorithm used for burst assembly in this research was HA, with several different thresholds. Thresholds were varied to allow HA to emulate MBS, with time threshold set too high, and MTD, with size threshold is set too low, for current network load. Thresholds used for burst size were set to 64KB and 9 KB, and assembly time varied from 100 μs to 2000 μs for 64, 16, 12, 8, 4 and 1 user in each node. Time thresholds and user load were combined to assure that burst loss really occurred in the network – burst loss ranged from 1.445% to 98.966%.

Burst assembly algorithms using real IP traffic were studied in [8]. Fig. 2 shows how different sets of thresholds change the inter-arrival time between bursts, and consequently define the optimum zone for burst assembly algorithms, defined as

corresponding to the minimum interarrival time between bursts with the maximum burst size.

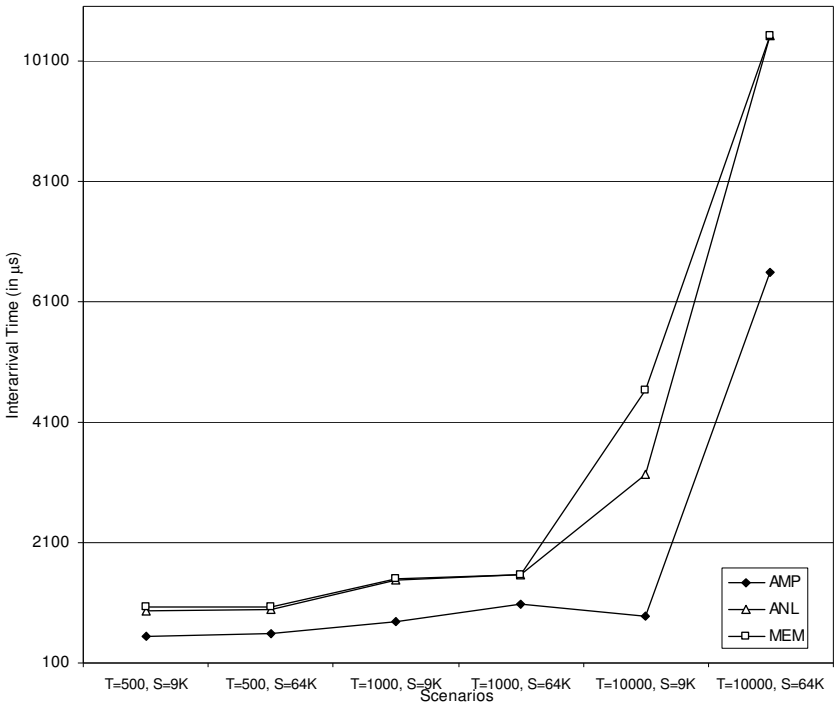

Fig. 2 – Burst Inter-arrival time for different threshold scenarios considering three network collection points (AMP = AMPATH, Miami, Florida, USA, ANL = Argonne National Laboratory to STARTAP, MEM = University of Memphis)

Since burst assembly thresholds are network point dependent [8], HA was used with a wide set of thresholds as to obtain a large range of burst characteristics. The result was the creation of bursts very differentiated in terms of Size (in Bytes) and Size (in number of Packets), results that are clearly visible in Fig. 3. The values ranging from 0.905% to 85.043% show the ratio of standard deviation calculated over the averaged Burst Size (in Bytes) and Burst Size (in Packets).

The research relevant results the simulator provided were: Number of bursts, size in bytes for each burst, size in packets for each burst, for both bursts created and bursts dropped. The ratio of – {burst, packets in bursts, bytes in burst} created over dropped was calculated and averaged for several simulations with different simulation time lengths.

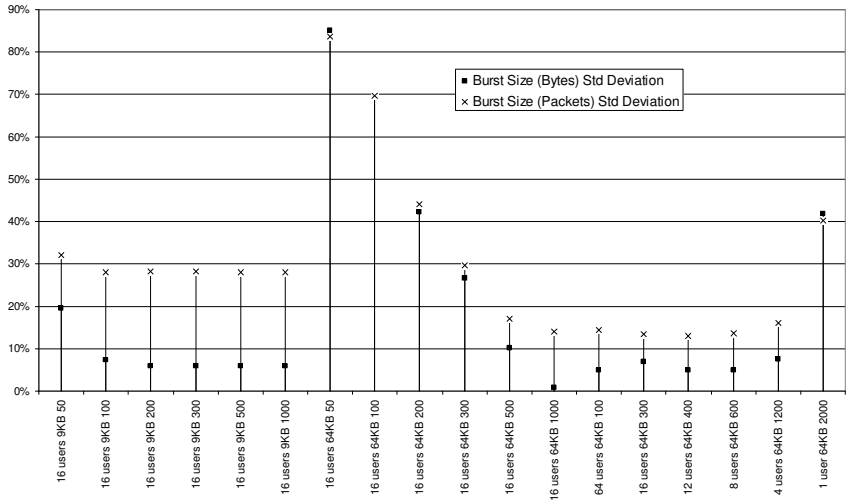

Fig. 3 – Standard deviation ratio of average Burst Size (measured in Bytes) and average Burst Size (measured in number of Packets)

4 Burst Loss Versus Packet Loss

The primary metric used for performance assessment of burst switched networks has been burst loss. There are a number of underlying assumptions in this statement that can be expressed in a simplified form, as follows:

1. all bursts are made of independent smaller data entities, which may be called packets (without loss of generalization);
2. all bursts are equally sized;
3. all bursts contain an equal number of packets.

If these three assumptions are hold true, then there is no doubt that burst loss metric is an adequate performance assessment measurement, and what's more, Burst Loss, Packet Loss and Byte Loss ratios are equal. But if bursts are not equally sized, what does it mean that a network lost a burst – exactly how many bytes were in this burst, and what's more, how many packets were lost? That is to say, the Burst Loss metric may not be relevant to real networks, who are know to exhibit self-similar bursty traffic [9-12].

Also, to the end users – machines and humans using the network – burst loss may not be meaningful. The expected network performance and the perceived quality of the service it's supposed to deliver, is measured in terms of "how long and how well is this content taking to travel from machine *A* to machine *B*", and this often means "how many packets were lost" and "how delayed the packet were". This also points out to conclusions already known from the study of burst assembly algorithms using real IP

packets: minimum packet delay and maximum burst size, i.e. optimization of burst assembly process, is achieved for the HA algorithm using time and size thresholds that are function of the network load on the burst assembly machine, and thus, are network point dependent [8]. As a result of the optimization of the burst assembly process, it has to be assumed that realistic burst switching deals with bursts that are not homogeneously sized, and of course do not contain a fixed number of packets [8].

If the three above mentioned assumptions can not be held true, as in the case where very heterogeneous burst traffic is generated (the case simulated and presented here), only two alternatives remain: either burst loss is not adequate as a performance assessment metric because it is not equal neither to byte loss neither to packet loss, and the latter would be more "user meaningful", or with real traffic the simulation proves the Law of Big Numbers, and so, the final results on the network can be assumed as if all the bursts have the same number of packets, and these in turn are equally sized, to the average number of packets per burst the first, and the average number of bytes per packet (and per burst) the latter.

The simulated network was a four-node ring with nodal degree of 2. The network was loaded with bursts assembled from real IPv4 packets, and simulated network data channels were defined as to allow for burst loss.

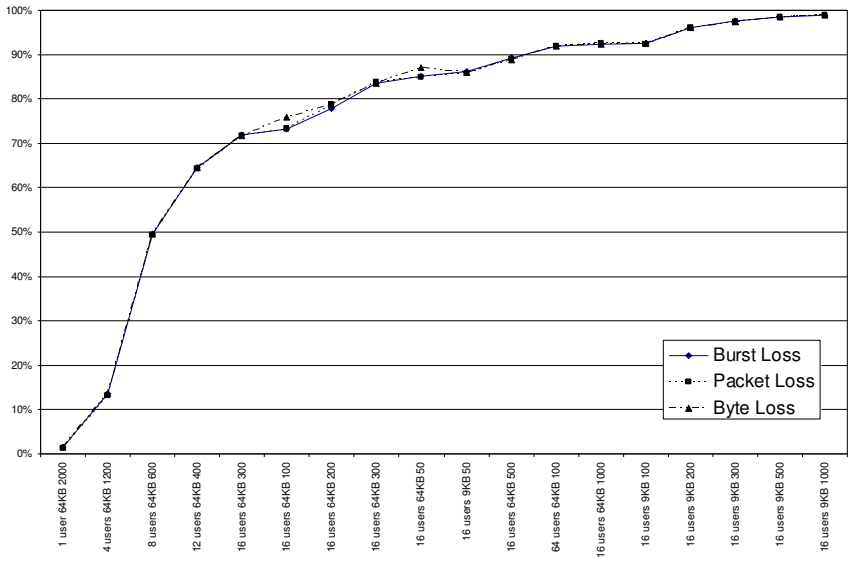

Fig. 4 – Burst, Packet and Byte loss for different burst assembly scenarios in an OBS JIT 4-node ring network (time thresholds in x-axis are μs)

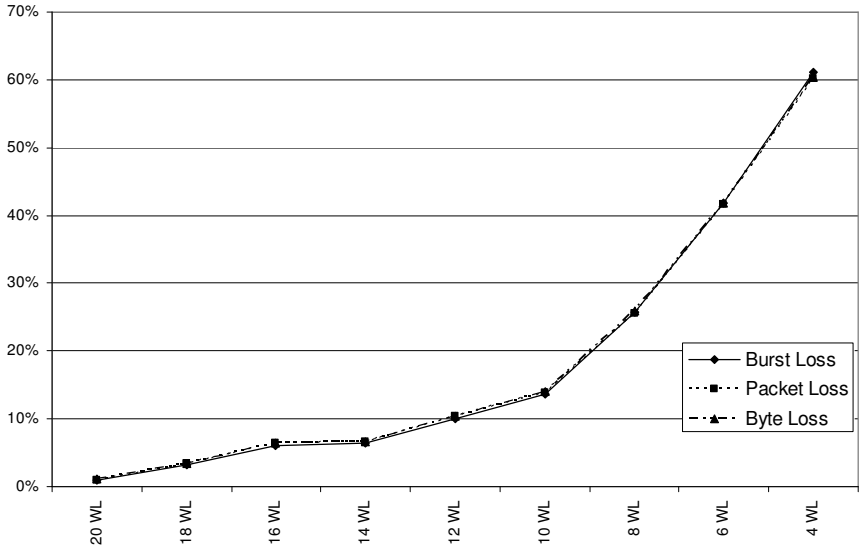

Fig. 5 – Burst, Packet and Byte loss for different burst assembly scenarios in an OBS JET 4-node ring network with 64 users, 64 KB burst size threshold, 40 μs time threshold and variable number of data channels (in x-axis of the graph)

A set of three ratios was devised and implemented in the simulator:

1. Burst Loss Ratio = number of bursts dropped / number of bursts created at edge nodes;
2. Packet Loss Ratio = sum of the packets in the bursts that were dropped / number of packets assembled in bursts at the edge nodes;
3. Byte Loss Ratio = sum of sizes (in bytes) of bursts that were dropped / sum of sizes (of bytes) of created bursts at the edge nodes.

The three values were calculated for all the simulated scenarios. Fig. 4 and Fig. 5 show the obtained measurements for JIT and JET signalling protocols respectively, with several burst assembly scenarios. As expected, despite such a wide range of burst characteristics in terms of size in bytes and number of constituent packets, and also, despite of the difference in the way the network signaling protocols accepts or drops the bursts, the Burst, Packet and Byte Loss ratios, are almost coincident.

5 Conclusion

Kantarci, Oktug and Atmaca [3] have evaluated the issue of burst loss versus packet loss using Pareto distributed traffic generation. When they measured it against Packet Loss for different burst assembly algorithms, their conclusion was that Burst Loss is not a reliable metric for performance assessment of OBS networks, since Packet Loss probability was lower than Burst Loss. On the contrary, results presented in this paper, obtained through simulation using real tributary IP data packets and realistic burst

assembly algorithms, show that Burst Loss is a reliable metric for assessment of Burst Switching networks, and that Burst Loss ranges very closely to Packet Loss and to Byte Loss, even when bursts are very heterogeneous in size both packet and byte wise. Also, this study proves that Burst Loss, Packet Loss and Byte Loss are equivalent performance assessment metrics for Burst Switched networks even when the signaling and resource reservation protocols are burst size sensitive, e.g. when void filling is performed (e.g. the JET protocol). Furthermore, it must also be noted that simulation using real tributary data associated with algorithms that are efficiency concerned, produce results that do not always agree with the ones obtained by statistically generated data.

References

[1] S. R. Amstutz, "Burst Switching - An Introduction," in IEEE Communications Magazine, vol., pp. 36-42, 1983.

[2] C. Qiao and M. Yoo, "Optical burst switching (OBS) - A new paradigm for an optical Internet," *Journal of High Speed Networks*, vol. 8, pp. 69-84, 1999.

[3] B. Kantarci, S. Oktug, and T. Atmaca, "Analyzing the Effects of Burst Assembly in Optical Burst Switching under Self-Similar Traffic," in Proc. Advanced Industrial Conference on Telecommunications, Lisbon, Portugal, 2005, IEEE Computer Society Press, pp. 109-114.

[4] N. M. Garcia, M. Hajduczenia, P. Monteiro, H. Silva, and M. Freire, "Modeling and Simulation of IPv6 Traffic," in 7th Internet Global Congress, Global IPv6 Summit, Barcelona, 2005.

[5] National Laboratory for Applied Network Research, "NLANR PMA: Special Traces Archive," in http://pma.nlanr.net/Special/, 2005, accessed at 2005-01-13.

[6] National Laboratory for Applied Network Research, "IPv4 hashing function source code (tsh file format)," in ftp://pma.nlanr.net/pub/dagtools-0.9.6.tar.gz, 2005, accessed at 2005-01-13.

[7] J. Teng and G. N. Rouskas, "A Comparison of the JIT, JET, and Horizon Wavelength Reservation Schemes on A Single OBS Node," in WOBS 2003, Dallas, Texas, 2003.

[8] N. M. Garcia, P. P. Monteiro, and M. M. Freire, "Assessment of Burst Assembly Algorithms using real IPv4 Data Traces," in (submitted) IEEE International Conference on Communications, ICC'06, Istabul, Turkey, 2006.

[9] W. T. Willinger and R. M. S. Sherman, "Self-similarity through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the source level," *IEEE / ACM Transactions on Networking*, pp. 71-86, 1997.

[10] K. Park, "How does TCP generate Pseudo-self-similarity?" in Winter Simulation Conference, 1997, pp. 215-223.

[11] M. S. Borella, S. Uludag, G. B. Brewster, and I. Sidhu, "Self-similarity of Internet Packet Delay," in IEEE ICC '97, 1997, pp. 513-517.

[12] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (extended version)," *IEEE Transactions on Networking*, vol. 2, pp. 1-15, 1994.