A Study of Traffic Statistics of Assembled Burst Traffic in Optical Burst Switched Networks

Xiang Yu, Yang Chen and Chunming Qiao
Department of Computer Science and Engineering
State University of New York at Buffalo
{xiangyu,yangchen,qiao}@cse.buffalo.edu

ABSTRACT

Optical Burst Switching (OBS) is considered as a promising switching technique for building the next generation optical Internet. In OBS networks, one important issue is how the performance will be affected by bursts assembled from packets, which is the basic transmission unit in OBS. In this paper, we study the fundamental statistic properties such as the burst length distribution, inter-arrival time distribution, as well as correlation structure of assembled burst traffic from burst assembly algorithms. From both theoretical and empirical results, it is demonstrated that after the assembly, the traffic will in general approach the Gaussian distribution. In particular, the variance of assembled traffic decreases with the increase in the assembly window size and the traffic load. However, the long range dependence in the input traffic will not change after assembly.

1. INTRODUCTION

Explosive growth of the Internet traffic makes all-optical networks a promising solution. Among the several optical switching technologies, burst switching [1–3] has gained a lot of attention recently because it combines the advantages of both circuit switching and packet switching .

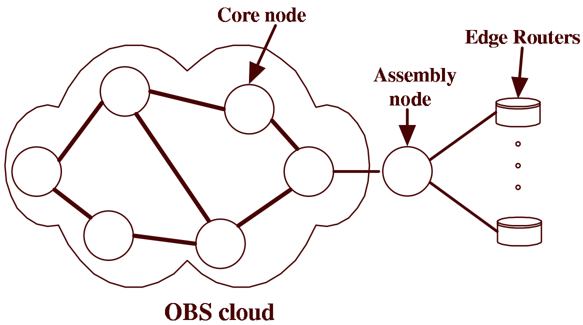


Figure 1. Optical burst switching network

There are two kinds of node in an optical burst switching (OBS) network as illustrated in Figure 1: assembly node and core node. The assembly node connects the edge routers (e.g. IP routers) to the core node inside the OBS cloud. Packets with the same destinations are usually assembled together using an assembly algorithm to reduce electronic processing load in the core nodes and increase switching efficiency [4–7].

Today's Internet traffic has been demonstrated to affect network performance by two kinds of bursty property: short range dependent (SRD) burstiness and long range dependent (LRD) burstiness [8–10]. An interesting phenomenon is that burst assembly can enhance the loss performance and throughput in OBS core networks by smoothing the input packet traffic. Previous work based on simulation in [5] claimed that in the assembly mechanism reduces the long range dependence in the input packet traffic. Nevertheless, no prior work has analyzed the statistic property of assembled traffic. Before applying popular traffic models to the assembled traffic in future studies in OBS networks, the above mentioned open issue of statistic property of assembled burst traffic needs to be addressed first.

In this paper, we first analyze the changes in the statistical property of the assembled traffic from input packet traffic. Starting with the most popular Poisson traffic model as input which is used to approximate the aggregated SRD traffic, we look into the burst size distribution, burst inter-arrival time distribution in the assembled traffic output from given burst assembly algorithms. In addition, we also analyze the effect of burst assembly on LRD input traffic. Our analysis in general proves the smoothness of the assembled traffic. On the other hand, it also reveals that the long range dependence property of the input traffic remains unchanged in light load condition, as verified by simulation results.

The rest of the paper is organized as follows. In Section II, two basic burst assembly algorithms are presented, which will be used to study the assembled traffic statistics throughout this paper. Section III provides mathematical studies on the statistical properties of the assembled traffic, which are considered with different assembly algorithms under both light and heavy load scenarios. In Section IV, simulation results are given to verify the theoretical results. Finally, Section V concludes this paper.

2. BURST ASSEMBLY ALGORITHMS

This section introduces algorithms which assemble input packet traffic into bursts. Two important factors: the assembly time and the assembled burst length are considered. Both of the factors can affect packet delay. The two burst assembly algorithms we study in this paper are, Algorithm I: Fixed-Time-Min-Length burst assembly and Algorithm II: Max-Time-Min-Max-Length burst assembly respectively, which are described below:

2.1. Algorithm I: Fixed-Time-Min-Length Burst Assembly

The Fixed-Time-Min-Length burst assembly algorithm [5] uses a fixed assembly time as the primary criteria, and based on this primary criteria, it requires each burst size to be larger than a minimum length. The following is the detail of this burst assembly algorithm:

Set a fixed assembly time window T and a minimum burst length b. Normally, $b < T * \lambda$, where λ is the average traffic arrival rate. Assume that the assembly queue starts to collect data for the ith burst at time 0, and denote the data arrived in time t as $p_i(t)$. (note that in this algorithm, $p_i(0) = 0$.) Initially, the burst index i is set to 0.

1. When a packet arrives in an empty burst assembly queue, start the time counter at $t = 0$, which increases with time;
2. When $t = T$
   if $p_i(t) \geq b$ **then**
   send all the collected data $p_i(t)$ as Burst i immediately;
   else
   increase the data size to b with padding and send the data out as Burst i immediately;
   end if
3. Increase the burst index i and go to step 1;

2.2. Algorithm II: Max-Time-Min-Max-Length Burst Assembly

The Max-Time-Min-Max-Length burst assembly algorithm uses the maximum assembly time as the primary criteria. In order to decrease the delay of an individual packet, it also allows a burst to be sent out as soon as the burst length reaches or exceeds a given maximum burst length. The detail of this algorithm is given as follows.

Set a maximum burst length B and a minimum burst length b as well as a maximum assembly time window T. Normally, $b < T * \lambda < B$). We also denote the data accumulated in the ith burst at time t as $p_i(t)$. Note that unlike in Algorithm I, $p_i(0)$ in Algorithm II may not equal to zero because of the possible leftover from the previous burst $i - 1$ if it is longer than B.

1. If the buffer is nonempty or when a new packet arrives, initiate timer t as 0 which increases with time;

    if $p_i(t) \geq B$ **then**
      go step 2;
    end if
    if $t \geq T$ **then**
      if $p_i(t) < b$ **then**
        increase the data size to b with padding and send the data out as Burst i immediately;
      else if $p_i(t) \leq B$ **then**
        send out the data as burst i immediately;
      end if
      increase i;
    end if

2. **while** total data size in the assembly buffer is larger than B **do**
      subtract a burst of length B from the buffer and send it out as burst i immediately, increase i;
    end while

3. go to step 1;

Note that if the maximum assembly time T in Algorithm II is set to be very small relative to the maximum burst size B, the assembled burst's length will never reach B before the it will be sent out as timer expires. In such a case, Algorithm II will function in the same way as Algorithm I with a fixed time T. In other words, Algorithm I can be treated as a special case of Algorithm II if $T * \lambda << B$. However, B may be small relative to T and hence we still treat Algorithm I and II separately.

3. ANALYSIS OF THE ASSEMBLED TRAFFIC

In this section, we study the assembled traffic theoretically, assuming either SRD traffic or LRD traffic as the input to the two burst assembly algorithms introduced in the previous section. We consider two traffic load scenarios: one with heavy traffic load (or small link capacity and the other with light traffic load (or infinite link capacity).

3.1. The Light Traffic Load (Infinite Link Capacity) Scenario

This section studies the two assembly algorithms under the light traffic load scenario, which means the traffic arrival rate is very small comparing to the link capacity, or in other words, the link capacity can be treated as infinite.

We begin with the case where the input traffic is Poisson.

3.1.1. Poisson Traffic Input

Packets arrive at an OBS assembly node in the form of multiplexed traffic from many independent sources. Previous studies have shown that such packets arriving in a short time period will become independent as the number of sources increases, and in fact, such multiplexed traffic will approach Poisson traffic [11–13]. Normally the assembly time period can be treated as short time period where Poisson traffic is used to model the input packet traffic.

For an assembly node with infinite link capacity, the transmission time of a packet is negligibly small and accordingly each arrival packet can be treated as a point in the time axis. In other words, *Simple Poisson Point* process [14] can be used to model the input traffic in the infinite link speed scenario, which assumes that: (1) no packet arrives at exactly the same time; (2) all packet arrivals are independent.

For simplicity, we assume fixed packet size first.

1. Poisson Traffic with Fixed Packet Size

Suppose all the packets have a size equal to a constant q, and the inter-arrival time x_m of these packets follows an exponential distribution:

$$f(x_m) = \frac{1}{\mu} e^{-\frac{x_m}{\mu}} \tag{1}$$

where μ is the expected value as well as the standard variance of x_m.

For Algorithm I (Fixed-Time-Min-Length burst assembly), the burst inter-arrival time t_m of the assembled traffic is equal to the time window T, i.e a fixed constant and thus we will focus on the burst size distribution for now.

The burst size denoted by variable p_i depends on the number of packets variable n_i arrived in the fixed time window T. The probability that there are n packet arrivals within time T is [15]:

$$\begin{aligned} & P\{n_i = n\} \\ & = P\{n \text{ packets arrive at time interval } T\} \\ & = \frac{(\mu T)^{n-1}}{(n-1)!} e^{-\mu T} \end{aligned} \tag{2}$$

The distribution of burst size p_i is the distribution of the sum of n_i packets' sizes, which arrive to build the Burst p_i, and thus the distribution of p_i can be calculated as

$$\begin{aligned} P\{p_i = p\} &= P\{n_i q = p\} \\ &= P\{n_i = p/q\} \\ &= \frac{(\mu T)^{p/q-1}}{(p/q-1)!} e^{-\mu T} \end{aligned} \tag{3}$$

For the Algorithm II (Max-Time-Min-Max-Length assembly), the assembled traffic will have a burst size between a maximum value B and a minimum value b. If the time window T is chosen to be much larger than B/q or if b is chosen to be close to B, the burst size will approach the constant B. Thus, the burst size has a very limited distribution range. However, the inter-arrival time t_i of the bursts is variable of interest in such a case.

Similarly to Equation (3), the distribution of inter-arrival time t_i is the distribution of the sum of $n_i = B/q$ packets' inter-arrival times, and can be calculated as follows if we use B as an approximation of the burst size:

$$\begin{aligned} P\{t_i = T_i\} &= P\{\sum_{m=1}^{n_i} x_m = T_i\} \\ &= P\{\sum_{m=1}^{B/q} x_m = T_i\} \\ &= \frac{(\mu T_i)^{B/q-1}}{(B/q-1)!} e^{-\mu T_i} \end{aligned} \tag{4}$$

In short, Burst Assembly Algorithm I shapes the Poisson traffic to an assembled burst traffic with a constant inter-arrival time (T) and a burst size distribution satisfying (3). On the other hand, Algorithm II shapes the Poisson traffic to an assembled traffic with an approximately constant burst size (B) and an inter-arrival time whose distribution satisfies (4).

From the details of (3) and (4), it is noticed that both of these two distributions are the distribution of the sum of many independent and identically distributed random variables. According to Central Limit Theorem (CLT) [15], both distributions in (3) and (4) will approach Gaussian distribution as the number of packet arrivals goes up or the assembled burst size goes up. In addition, as more and more packets are aggregated in a burst, the burst length in Algorithm I will approach its mean value:

$$p_i \rightarrow \bar{p} = q\frac{T}{\mu} \tag{5}$$

and the inter-arrival time in Algorithm II will also approach its mean value:

$$t_i \rightarrow \bar{t} = \mu\frac{B}{q} \tag{6}$$

In other words, the assembled burst traffic from both Algorithm I and II will become Gaussian distribution of zero variance, i.e., constant rate traffic.

2. Poisson Traffic with Variable Packet Size

If the input traffic is Poisson traffic with the variable packet size, which has for example, exponential distribution, the results are quite similar to those obtained when the input is Poisson traffic with fixed packet size. Below, we only focus on Algorithm II as Algorithm I can be treated similarly.

Suppose the Poisson traffic has the same packet inter-arrival time distribution as in (1), and a packet size distribution as follows:

$$f(q_m) = \frac{1}{a}e^{-\frac{q_m}{a}} \tag{7}$$

where by definition a is the expected value as well as the standard variance of packet size q_m.

The distribution of burst inter-arrival time from Algorithm II can be calculated as follows:

$$\begin{aligned} P\{t_i = T_i\} &= P\{\sum_{m=1}^{n_i} x_m = T_i\} \\ &= \sum_{n=1}^{\infty} P\{\sum_{m=1}^{n_i} x_m = T_i | n_i = n\} P\{n_i = n\} \\ &= \sum_{n=1}^{\infty} \frac{(\mu T_i)^{n-1}}{(n-1)!} e^{-\mu T_i} \times \frac{(aB)^{n-1}}{(n-1)!} e^{-aB} \\ &= [e^{-aB} \sum_{n=0}^{\infty} \frac{(aB\mu)^n T_i^n}{n!}] e^{-\mu T_i} \end{aligned} \tag{8}$$

To analyze (8), we first give a lower bound and an upper bound (derived in Appendix) as follows:

$$\frac{1}{2}e^{-aB}e^{-\mu T_i + \sqrt{aB\mu T_i}} < P\{t_i = T_i\} < Ce^{-aB}e^{-\mu T_i + \sqrt{aB\mu T_i}} \tag{9}$$

where $C > 1$ is a constant.

From (9), we can see that when T_i is small, the distribution will be larger than exponential because of the additional term $\sqrt{aB\mu T_i}$. However, when T_i becomes large, the distribution will be dominated by $-\mu T_i$ and thus become the same as exponential tail distribution. Generally speaking, for SRD traffic which can be treated as Poisson traffic in short time period, the assembled traffic will approach Gaussian. As the ratio of variance over mean decreases as a result of increase in the number of packets aggregated in a burst, the assembled traffic will become more smooth.

We can also look at this burst inter-arrival time distribution from a different angle. If we fix the number of packets in a burst, then the burst inter-arrival time t_i can be viewed as the aggregation of a fixed number of packet inter-arrival time, that is,

$$t_i \sim \sum_{m=1}^{n} x_m \tag{10}$$

and also, the number of packet is decided by the distribution of packet size q_m. Since we have $n \sim B/q$ where B is fixed, the distribution of n is decided by distribution of q_m and should also converge to Gaussian. And the distribution of t_i is then the result from the multi-effect of aggregation of packet sizes q_m (or n) and inter-arrival times x_m. As B goes up, from CLT, n will approach a constant from Gaussian distribution. And if we fix n in the extreme case, the distribution of t_i will be decided by the summation random variables $\sum_{m=1}^{n} x_m$. Use CLT again, the distribution of t_i will approach Gaussian as n goes large. So we conclude that t_i will finally converge to a constant from the Gaussian distribution.

The only difference between having a variable packet size input Poisson traffic and having a fixed packet size input Poisson traffic is with the latter, the assembled traffic converges to Gaussian more quickly, since it only needs to use CLT once and only one variable need to be converged, while the former converges to Gaussian more slowly because the converge process uses CLT twice and needs two variables converge at the same time.

Given that Gaussian model can be used to analyze the assembled according to the above discussion, with more packets aggregated in a burst, the inter-arrival time distribution t_i will approach its mean value:

$$t_i \rightarrow \bar{t} \rightarrow \mu \frac{B}{a} \tag{11}$$

Similarly, the burst size distribution of the assembled traffic from Burst Assembly Algorithm I will also has the same distribution as (8). And the assembled burst traffic will also approach Gaussian and become smooth.

3.1.2. Long Range Dependent Traffic

According to the CLT, the assembled traffic for the long range dependent input traffic will also approach Gaussian because of the packet independence property in a short time period. To determine the long range dependence property, we can examine the correlation structure [8, 16] of the assembled traffic.

If we divide the time axis into K blocks and each block contains m sub-blocks. Suppose the traffic load arriving in the ith sub-block of kth block is $W_{(k-1)m+i}$, and $W_k^{(m)} = \frac{1}{m}(W_{(k-1)m} + ... + W_{km-1})$ is called local average workload. Let $r^{(m)}(k)$ denote the correlation with lag k of the original long range dependent input packet traffic, and $r^{(m)}(k)$ satisfies:

$$r^{(m)}(k) \sim k^{2(H-1)} \text{as } m \rightarrow \infty. \tag{12}$$

Suppose there are n packets contained in the time block k, then the local average workload can be rewritten as $W_k^{(m)} = \frac{1}{m}\{q_{N_{k-1}+1} + ... + q_{N_{k-1}+n}\}$, where N_{k-1} is the total number of packets contained in the previous $k-1$ blocks and $q_{N_{k-1}+i}$ denotes the $(N_{k-1}+i)$th packet in the arrived traffic load. So, when $m \rightarrow \infty$ we also have $n \rightarrow \infty$, and (12) is rewritten as

$$r^{(m)}(k) \sim k^{2(H-1)} \text{as } n \rightarrow \infty. \tag{13}$$

If the sub-block time is large enough compared with assembly window size, we can approximate $W_k^{'(m)}$, the local average workload in assembled burst traffic in time block k, with the average summation of burst. Suppose there are "equivalently" (Here "equivalently" means that $W_k^{'(m)}$ may contain partial burst in head and tail, but these two partial bursts will not affect the total $W_k^{'(m)}$ as long as sub-block time window is large enough) l bursts in the workload $W_k^{'(m)}$:

$$W_k^{'(m)} = \frac{1}{m}\{p_{L_{k-1}+1} + ... + p_{L_{k-1}+l}\} \tag{14}$$

where L_{k-1} denotes the total number of bursts contained in previous k-1 blocks and $p_{L_{k-1}+i}$ denotes the $(L_{k-1}+i)$th burst in the arrived traffic load.

In light traffic load scenario, if the sub-block time period is large enough compared to the assembly window size, the total size of packets contained in a sub-block time period should be "equivalent" to the total size of bursts contained in the same sub-block time period. So, $W_k^{'(m)}$ will approach $W_k^{(m)}$, and $r^{'(m)}(k)$, which is the correlation of $W_k^{'(m)}$ will also approach $r^{(m)}(k)$: since $n \to \infty$ in (14) immediately follows $l \to \infty$ and also $m \to \infty$, so we have

$$r^{'(m)}(k) \sim k^{2(H-1)} \text{as } m \to \infty. \tag{15}$$

From (15) we conclude that the correlation structure of the assembled traffic does not change or in other words, the long range dependence property in the input traffic is kept unchanged in the assembled traffic, which is verified by our simulation results though it disagrees with the results in [5]. However, the assembled traffic will get smoothed in short range, i.e., the assembly period range, because of the central limit theorem.

3.2. The Heavy Traffic Load (Low Link Capacity) Scenario

The results for the heavy traffic load scenario are expected to be different because when the traffic arrival rate is comparable to the link capacity, queuing will happen in the assembly buffer. We will discuss the assembled traffic statistics using two burst assembly algorithms with either SRD (Poisson) or LRD traffic as an input.

3.2.1. Poisson Traffic

We first examine Algorithm I where all the packets in the assembly buffer will be sent out as a burst when the timer reaches the window size T. Accordingly, there will be no packets left in the buffer at the time. However, because the link capacity is finite, it takes some (non-negligible) time to transmit each burst. To avoid the loss, the next burst has to wait a minimum time T^{min} before it can be sent out, where T^{min} equals to the transmission time of the previous burst p_{i-1}, i.e., $T_i^{min} = p_{i-1}/C$, where C is the link capacity. Note that when the local arrival rate in a given time period is higher than the link capacity, the previous burst may be very long and requires a long time for transmission, making $T_i^{min} > T$ a possibility. In such a case, the inter-arrival time distribution will approach the distribution of T^{min}, which is related to the burst length distribution. And after the heavy load period is over, the inter-arrival time will then go back to the fixed T.

Next, let's look at the burst length distribution. As mentioned earlier, the burst length could be very long with a fixed packet size Poisson traffic input. Based on the CLT, the number of packets arrived in any fixed time interval will approach a constant independent of the length of the interval. The burst length T will also approach a constant and so will T^{min}. Thus, if the time window T is large, for Poisson traffic which is bursty only in short time period, T^{min} can not be much larger than T and burst size distribution has a limited range. In such a case, the assembled traffic will approximately have both constant burst size and inter-arrival time, i.e., become a constant rate traffic. However, if the time window is not large enough, and T^{min} can be larger than the fixed time window T, then the assembled traffic will have variable burst size as well as variable burst inter-arrival time. But in such a case, large burst usually follows a proportionally large inter-arrival time. And the traffic rate (B_i/T_i) is still constant.

For Burst Assembly Algorithm II, when there are enough packets queued in the buffer, the assembly node will subtract the next burst of size B directly from its buffer and without having to wait for additional packets to arrive. This burst will be immediately sent out as soon as the previous burst is sent out, and the assembled traffic will also become constant rate traffic with a fixed burst length B and a fixed inter-arrival time $T = B/C$.

In short, as the traffic load goes up, the assembled traffic using either Algorithm I or II with SRD traffic input from both will become constant rate traffic because of their smoothing effect during the short time period.

3.2.2. Long Range Dependent Traffic Input

The results for the LRD traffic scenario will be somewhat different from those for SRD input. This is because under the heavy load scenario, the input LRD traffic may be heavy in a very long time period, and then followed by a long silent period, whereas the input SRD traffic will always be heavy. In addition, Algorithms I and II will also have different effects on the assembled burst traffic when the input traffic is LRD and heavy.

First, for Burst Assembly Algorithm I, the long busy period with LRD traffic input can be treated in the same way as the heavy traffic load scenario, while the long silent time period with LRD traffic input can be treated the same way as light traffic load scenario with SRD traffic input. Since in Algorithm I, all the packets as a burst in the assembly buffer is sent out at the end of each fixed time period, the assembly process won't affect much of the long range dependent traffic pattern as the traffic correlation structure changes little in the long run (even though long bursts may be formed during long busy period). When the busy period comes, the assembled traffic will soon become the constant rate traffic as discussed for the case with Poisson traffic input, and when the silent period comes, the assembled traffic will become the Gaussian traffic immediately. In other words, traffic will get smoothed in a short time period (where the length depends on the assembly time), but not in a long time period. In fact, the assembled traffic will still keep its LRD property when using Algorithm I, which again disagree with the conclusion in [5].

However, for Burst Assembly Algorithm II, the case will be different because it utilizes the buffer to smooth the traffic. Given very low link rate and infinite buffer, there will always be more data than the given maximum burst length B in a assembly queue, and thus the assembled traffic is heavy for both busy and silent periods. The effect of the buffer is to shift the traffic load accumulated in a busy period to the following silent period, and the assembled traffic will still approach the constant rate traffic. However, the price paid for such a removal of the long range dependence in the assembled traffic is the long delay of individual packets, which arrive in a busy period but have to wait in the buffer until they get served.

4. SIMULATION RESULTS AND DISCUSSION

In our simulation study, we use Poisson traffic and long range dependent traffic with traffic rate 100 unit per second (e.g., average packet size is 10 unit and average inter-arrival time is 0.1 second) to verify our theoretical analysis. And we use the load of 1 percentile to represent the light traffic load scenario and 100 percentile to present the heavy traffic load scenario.

In our simulation, the LRD traffic is generated by multiplexing independent sources with Pareto-distributed ON/OFF periods [17]. In the ON period, traffic source send packets out back to back while in the OFF periods, no packet comes out from the source. The probability density function of the Pareto distribution are:

$$f(x) = \frac{\alpha \cdot b^{\alpha}}{x^{\alpha+1}} \tag{16}$$

For LRD traffic, α should be chosen between 1 and 2. The Hurst parameter [16] $H = (3 - \alpha)/2$. In this paper we choose $\alpha = 1.5$, i.e., $H = 0.75$.

4.1. Light Traffic Load Scenario

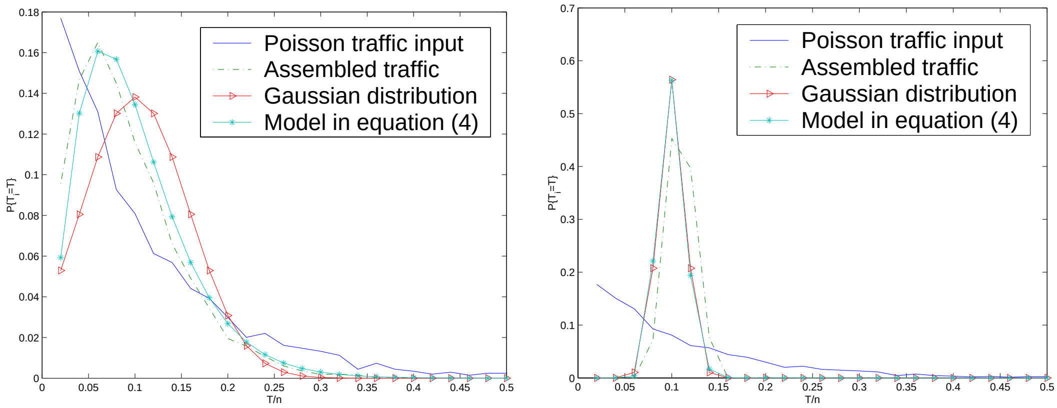
When the input is Poisson traffic with a fixed packet size q, Figure 2 shows the burst inter-arrival time distribution using Algorithm II for two maximum values of burst size ($B = nq$), namely $n = 3$ and $n = 50$ respectively.

Figure 2(a) illustrates that when the maximum burst length is set to be small ($n = 3$), Equation (4) models the burst inter-arrival time distribution more precisely than Gaussian. That is, for such a small burst length, the convergence of the assembled traffic to Gaussian distribution is not obvious. However, when the maximum burst length is set to be large enough (i.e., $n = 50$), the convergence to Gaussian distribution in assembled traffic is obvious. In such a case, the distribution in equation (4) also converges to Gaussian, which overlaps with Gaussian distribution as shown in Figure 2(b).

Figure 3(a) and 3(b) compare the two cases where the input Poisson traffic has a fixed packet size and a variable packet size. The inter-arrival time (normalized with respect to the average number of packets n in a burst) distribution of the traffic assembled from input Poisson traffic with variable packet sizes also converges to Gaussian as shown in Figure 3(b) as the average burst length increases, but with a slower converging rate than that shown in Figure 3(a).

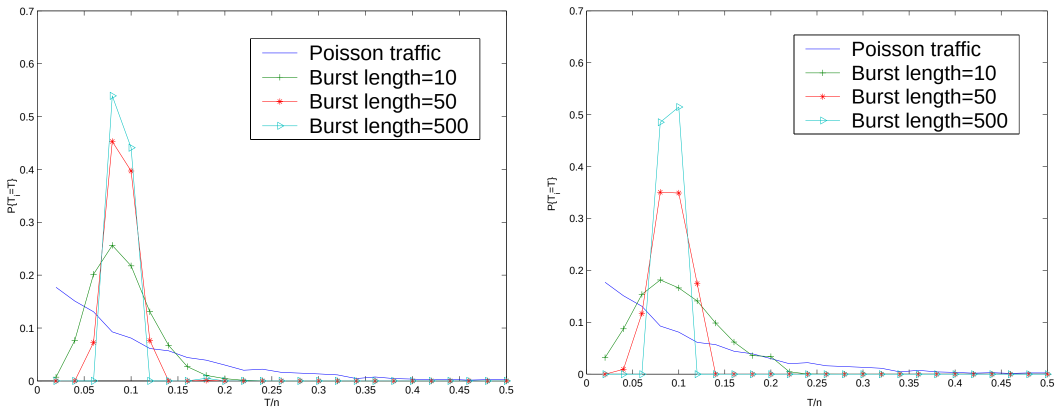
Our results (though not shown) also verified the assumption made in our theoretical study in that the burst size in the light traffic load scenario will approach a constant B if the assembly time window T in Algorithm II is set to be much larger than the time needed to form a burst B. The burst size will also approach a constant when the max burst length is large. (keep the ratio of the burst length and time window unchanged).

Figure 4 shows the burst size distribution when Algorithm I is used, which is similar to the burst inter-arrival time distribution in Algorithm II in Figure 3. In the following discussion, we will omit the results for Algorithm I.



(a) burst length = 3   (b) burst length = 50

Figure 2. Inter-arrival distribution from Algorithm II with Poisson traffic(fixed packet size).



(a) fixed packet size   (b) variable packet size

Figure 3. Comparison of convergence of Algorithm II with different Poisson traffic input.

4.2. Heavy Traffic Load Scenario

Figure 5 illustrates the burst inter-arrival time distribution when using Algorithm II under different input traffic loads. In the simulation, the maximum time window T is chosen to be slightly larger than the time needed to assemble a maximum burst length (i.e. $T = 1.1B/\lambda$). As can be seen, for the heavy traffic load (100 percentile) scenario, 99.97% inter-arrival times are centered around the mean value of $T/n = 0.1$ within the 0.01 interval, and thus such a distribution can almost be treated as a constant as 0.1. In the light traffic load scenario (1 percentile), on the other hand, the inter-arrival time has a wide range distribution, which cannot be treated as a constant. In addition, our results (though not illustrated here) indicate the following burst length distribution in the heavy traffic load scenario: 99.99% of bursts have the average burst length, while in the light traffic load scenario, 76.9% bursts have the average burst length and the remaining 23.1% of the bursts have a wide range of distribution. In short, in the heavy traffic load scenario, the assembled burst traffic is approaching a constant rate traffic with a fixed burst length as well as a fixed inter-arrival time.

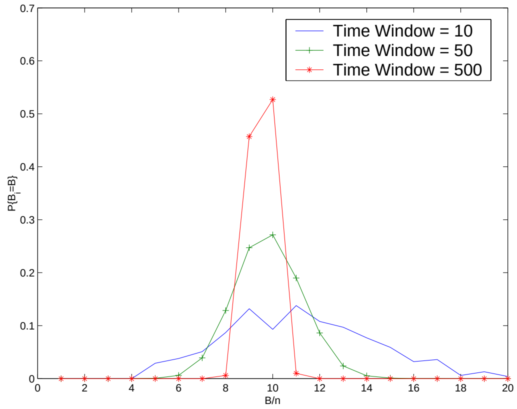


Figure 4. Burst length distribution with different assembly time windows in Algorithm I

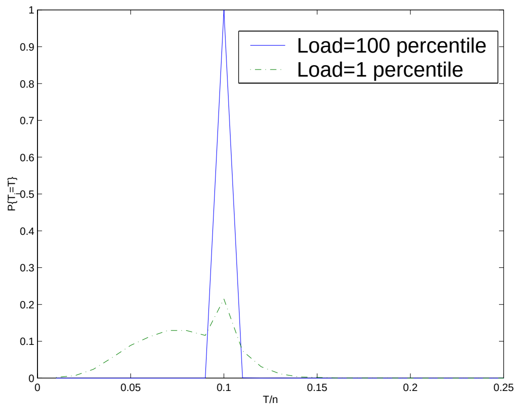


Figure 5. Inter-arrival time distribution at different traffic load

4.3. Long Range Dependence

To study the long range dependence in the assembled traffic, we use an input LRD traffic with its Hurst parameter equals to 0.75. Figure 6 illustrates the R/S plots [18,19] for the input packet traffic and the assembled traffic respectively. The slope of the R/S plot represents the Hurst parameter associated with the corresponding traffic. It can be seen that the Hurst parameter for the assembled traffic in the light traffic load scenario (e.g. 1 percentile) and normal traffic load scenario (e.g., 50 percentile) are both identical with the input LRD traffic. This implies that burst assembly did not reduce the Long Range Dependence in both light and normal traffic load scenarios. However, in the extreme heavy traffic scenario (e.g. 100 percentile), the assembled traffic has been smoothed in both short range and long range as the Hurst parameter becomes 0 indicated by horizontal line in the R/S plot.

4.4. OBS Core Node Performance

In order to show the smoothing effect of the assembled traffic on the performance enhancement in OBS networks, we provide simulation result for a single OBS core node with 8 output wavelengths, first-fit scheduling with void filling [7]. The input to the scheduling algorithm is either unassembled (raw) Poisson packet traffic or burst traffic assembled from the same Poisson packet traffic using burst assembly Algorithm II, both under the same load scenarios in the OBS code node. Figure 7 compares the loss rates and bandwidth utilization. It can be seen that the smoothing effect of assembly algorithm enhances both performance metrics to some extent.

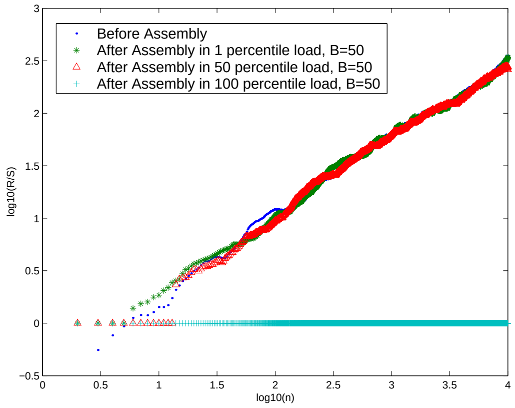


Figure 6. R/S plot

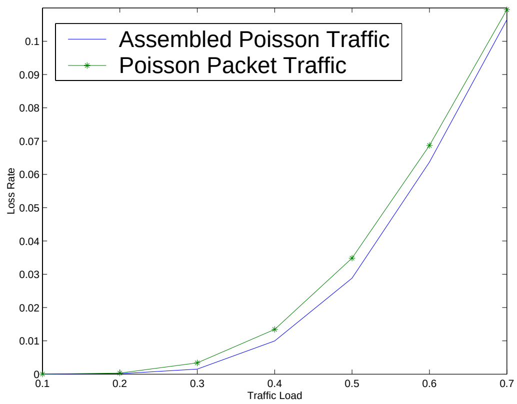


Figure 7. Performance comparison in core node

5. CONCLUSION

In this paper, we have studied the statistical property of burst traffic assembled from either Poisson or LRD traffic both theoretically and empirically. Since the input packet traffic is multiplexed from many sources, the packets' inter-arrival time and packet size will be independent in short time period as the multiplexing degree goes up. Thus we can in general apply the central limit theory to any such multiplexed traffic in the burst assembly window and conclude that the burst length or the inter-arrival time of the assembled burst traffic will approach Gaussian distribution.

The assembled traffic from Poisson traffic input, with either a fixed or variable packet size, may eventually converge to Gaussian distribution. But the long range dependence in the input traffic will remain unchanged in the assembled traffic. Furthermore, the Gaussian distribution of the assembled traffic will approach to the constant mean value under an extremely heavy load and the assembled traffic will becomes constant rate traffic. Our results in the statistical study of the assembled traffic are useful for OBS scheduling. The assembled burst traffic is smoother than the input packet traffic in terms of the short range burstiness, i.e., the statistical variance. The smoothed assembled traffic enhances the performance of OBS core node as demonstrated in our paper.

REFERENCES

1. M. Yoo, M. Jeong, and C. Qiao, "A high speed protocol for bursty traffic in optical networks," in *SPIE's All-Optical Communication Systems: Architecture, Control and Protocol Issues*, **3230**, pp. 79–90, 1997.

2. C. Qiao and M. Yoo, "Optical burst switching(OBS)–a new paradigm for an optical Internet," *Journal of High Speed Networks* **8**(1), pp. 69–84, 1999.
3. J. Turner, "Terabit burst switching," *Journal of High Speed Networks* **8**(1), pp. 3–16, 1999.
4. C. Xin and C. Qiao, "A comparative study of OBS and OFS," in *Proceeding of Optical Fiber Communication Conference*, 2001.
5. A. Ge, F. Callegati, and L. Tamil, "On optical burst switching and self-similar traffic," *IEEE Communications Letters* **4**, pp. 98–100, March 2000.
6. K. Dolzer and C. Gauger, "On burst assembly in optical burst switching networks - a performance evaluation of just-enough-time," in *Proceedings of the 17th International Teletraffic Congress (ITC 17)*, pp. 149–160, 2001.
7. Y. Xiong, M. Vandenhoute, and H. Cankaya, "Control architecture in optical burst-switched WDM networks," *IEEE Journal onSelected Areas in Communications* **18**, pp. 1838–1851, October 2000.
8. W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self-similar nature of ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking* **2**, pp. 1–15, February 1994.
9. V. Paxson and S. Floyd, "Wide-area traffic: The failure of poisson modeling," *IEEE/ACM Transactions on Networking* **3**, pp. 226–244, June 1995.
10. M. Crovella and A. Bestavros, "Self-similarity in world wide web traffic: Evidence and possible causes," in *proceeding of ACM SIGMETRICS*, 1996.
11. J. Roberts, "Traffic theory and the internet," in *IEEE Communication Magzine*, pp. 94–99, 2001.
12. J. Cao, W. Cleveland, D. Lin, and D. Sun, "The effect of statistical multiplexing on long-range dependence of internet packet traffic," tech. rep., Bell labs, 2002.
13. J. Cao, W. Cleveland, D. Lin, and D. Sun, "On the nonstationarity of Internet traffic," in *proceeding of ACM SIGMETRICS*, pp. 102–112, 2001.
14. D. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Process*, Springer-Verlag, 1998.
15. A. Papoulis, *Probability, Random Vaiables, and Stochastic Processes*, McGraw-Hill, 1991.
16. R. Addie, M. Zukerman, and T. Neame, "Broadband traffic modeling: Simple solutions to hard problems," *IEEE Communications Magazine* **36**(8), pp. 88–95, 1998.
17. W. Willinger, V. Paxson, and M. S. Taqqu, "Self-similarity and Heavy Tails: Structural Modeling of Network Traffic," *A Practical Guide to Heavy Tails: Statistical Techniques and Applications* , 1998.
18. M. Taqqu, V. Teverovsky, W. Willinger, and D. Wilson, "Estimators for long-range dependence: an empirical study," in *Fractals*, **3**, pp. 785–788, 1995.
19. B. Mandelbrot and M. Taqqu, "Robust R/S analysis of long run serial correlation," in *Proceedings of 42nd Session ISI*, pp. 69–99, 1979.

Appendix

This appendix provides the proof for the lower and upper bound for Equation(8).

To obtain a lower bound, we look at the following inequality:

$$\frac{(aB\mu)^n T_i^n \cdot}{(n!)^2} \geq \frac{(\sqrt{aB\mu T_i})^{2n}}{(2n)!}$$

then, we can get

$$\begin{aligned} \frac{(aB\mu)^n T_i^n}{(n!)^2} &\geq \frac{(\sqrt{aB\mu T_i})^{2n}}{(2n)!} \\ &> \frac{1}{2}\{\sum_{n=1}^{\infty}[\frac{(\sqrt{aB\mu T_i})^{2n-1}}{(2n-1)!} + \\ &\frac{(\sqrt{aB\mu T_i})^{2n}}{(2n)!}] + \frac{(\sqrt{aB\mu T_i})^0}{(0)!}\} \end{aligned}$$

$$= \frac{1}{2}\sum_{m=0}^{\infty}[\frac{(\sqrt{aB\mu T_i})^m}{m!}]$$
$$= \frac{1}{2}e^{\sqrt{aB\mu T_i}} \quad (17)$$

And the lower bound can be obtained as follows:

$$P\{t_i = T_i\} > \frac{1}{2}e^{-aB}e^{-\mu T+\sqrt{aB\mu T_i}} \quad (18)$$

To obtain an upper bound, we look at the following inequality:

$$\frac{(aB\mu T_i)^n T_i^n}{(n!)^2} < C\frac{(aB\mu)^{\frac{n}{2}}T_i^{\frac{n}{2}}}{n!}$$

from which we have

$$\sum_{n=0}^{\infty}\frac{(aB\mu T_i)^n}{(n!)^2} \quad < \quad \sum_{n=0}^{\infty}C\frac{(aB\mu)^{\frac{n}{2}}T_i^{\frac{n}{2}}}{n!}$$
$$= \quad e^{\sqrt{aB\mu T}} \quad (19)$$

where $C > 1$ is a constant. And we get the upper bound as follows:

$$P\{t_i = T_i\} < Ce^{-aB}e^{-\mu T_i+\sqrt{aB\mu T}} \quad (20)$$