Dynamic Shaping for Self-Similar Traffic Using Network Calculus

Halima Elbiaze, Tijani Chahed, Tülin Atmaca, and Gérard Hébuterne

Institut National des Télécommunications
9 rue Charles Fourier 91011 Evry CEDEX - France
{halima.elbiaze, tijani.chahed, tulin.atmaca,
gerard.hebuterne}@int-evry.fr
Phone : +33 1 60 76 47 42 , Fax : +33 1 60 76 47 80

Abstract. The focus of this paper is the shaping of self-similar traffic at the access of an optical node. We propose a novel algorithm that dynamically shapes the incoming traffic, based on service curves equations, in order to meet the optical nodes constraints in terms of buffer size or delay. We first estimate arrival parameters within various time intervals in order to make the incoming traffic fit into a token bucket traffic specification (Tspec) format. We then derive the shaping parameters based on deterministic service curves. Those shaping parameters vary dynamically according to the Tspec of every time window. We eventually set those parameters back into the original model in order to meet some QoS constraints at the optical network level.

1 Preliminaries and Problem Relevance

Optics has been identified as a key technology able to provide a large capacity to transport massive IP flows, and to cope with different Quality of Service (QoS) requirement. The self-similar nature of IP traffic has been demonstrated by several studies and mesurements. Due to the lack of optical memories, QoS could be offered through combined exploitation of electronic memories in the edges and optical ressources in the core of the optical network. The traffic shaping takes place in the edges and has a real impact to maintain the logical performance to its highest level.

Both IETF, and ITU have identified traffic shaping as a way to: 1) allocate a suitable amount of resources (buffer memory, bandwidth) to a connection to achieve its required QoS and 2) police traffic and assure ¨fair¨ access to a shared resource. The problem studied in this paper [1], is motivated by the desire to obtain applicable performance bounds for a very high-speed optical network dealing with self-similar traffic. One may view the problem considered in this paper as a ¨channel capacity¨ issue associated with dynamic shaping at the network edge.

[1] supported by the "Réseau National de Recherche en Télécommunications" under the decision number 99S 0201-0204 and the European IST Project DAVID (Data and Voice Integration over DWDM)

E. Gregori et al. (Eds.): NETWORKING 2002, LNCS 2345, pp. 314–326, 2002.
© Springer-Verlag Berlin Heidelberg 2002

To achieve this aim, a tool for studying end-to-end, bounded performance is needed. Classical queuing analysis studies average performance for aggregate traffic. It focuses on single server environments with attractive traffic models. However, in the packet-switching, integrated services models, bounds on the end-to-end performance need to be studied in a networking environment with traffic dynamics, interactions and burstiness far more complex than in the previous case. Worst-case performance bounds on the packet flow make it possible to derive guaranteed maximum and minimum values rather than averages; which is necessary when dealing with emerging multimedia networking scenarios. In this work, we use the deterministic version of the service curves method [3] and particularly Network Calculus (NC) [1], its Min-Plus algebra formulation. The remainder of this paper is organized as follows. In Section 2, we set the end-to-end system and describe the modeling of source and network in terms of arrival and service curve. In Section 3, we study the performance of the system in the absence of shaping. We also motivate the need for dynamically shaping the traffic in order to meet the network constraints. In section 4, we focus on dynamic shaping between a LAN and the optical network, we show the regions of shaping to meet both buffer and delay constraints and propose a novel algorithm for dynamic shaping based on the equations of the service curves. Some numerical results are presented and discussed in Section 5. Concluding remarks are eventually given in Section 6.

2 End-to-End System

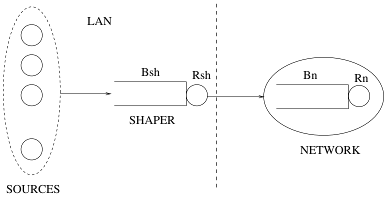


Fig. 1. End-to-end System

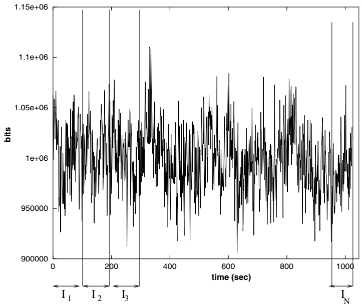


Fig. 2. Arrival process

2.1 Source Modeling and Arrival Curves

In source modeling, packet and connection arrival processes are often assumed to be Poisson owing to the attractive theoretical properties of such models [6]. Numerous studies have shown, however, that for both LAN and WAN networks, the distribution of packet interarrivals clearly differs from the exponential distribution [5]. Recent works argue convincingly that LAN traffic is much better modeled using self similar processes [11], which have very different theoretical properties than Poisson models. A subsequent investigation suggests that the same holds for WAN traffic too [12].

The strength of self similar models is that they are able to incorporate Long-Range Dependence (LRD), which informally means significant correlations across arbitrarily large time scales. For many networking issues, the presence or absence of LRD plays a critical role in the behavior predicted by analytical models. For example, the presence of LRD can completely alter the waiting times at the tail of a queue [4]. Self-similar processes are very difficult to tackle and render traffic characterization cumbersome in this case. One way to circumvent this is to bound the traffic rather than exactly characterizing it as suggested by recent models based on the service curves approch [7], [2], [8], [9] and [10]. Explicitly, in [7], a traffic stream is said to satisfy the $(X_{min}, X_{ave}, I, S_{max})$ model if the inter-arrival time between any two packets in the stream is larger than X_{min} during any interval I of length l, the average packet inter-arrival in I is larger than X_{ave}, and the maximum packet size is smaller than S_{max}. Alternatively, referring to [2], a traffic stream satisfies the (σ, ρ) model if, in I, the number of bits is less than $\sigma + \rho u$, $u \in I$. In the (σ, ρ) model, σ and ρ can be viewed as the maximum burst size and the long term bounding rate of the source, respectively. A similar argument is used in [8] and [9]. Rather than using the bounding rate, the Deterministic Bounding Interval-Dependent (D-BIND) model, found in [10], uses a family of rate-interval pairs where the rate is a bounding rate over the corresponding interval length. The model captures the intuitive property that over longer interval lengths, a source may be bounded by a rate lower than its peak rate and closer to its long-term average rate.

Traffic Specification (Tspec), introduced by the IETF for IP, is a description of the allowed traffic pattern a source can emit and not the actual one. A pair of token buckets is used by the traffic sender to describe the traffic it expects to generate and by the QoS control services to describe the parameters of traffic for which the reservation should apply. A token bucket specification is not a characterization parameter but a data structure definition. It takes the form of a token bucket, with rate r and depth b, plus a peak rate p and maximum packet size M. Units are bytes and bytes per second. We now show how self-similar traffic can be made to fit into a token bucket Tspec format.

Figure 2 depicts a self-similar process. As stated earlier, it is characterized by a similar irregular behavior at different time scales. To put such a process in a Tspec formulation, we partition the whole process into N equal, non-overlapping blocks, corresponding to time intervals $(I_i)_{i=1,\ldots,N}$, of length l_N, and approximate the traffic volume within each interval I_i by a corresponding set of Tspec parameters. In doing so, we obtain a piece-wise Tspec formulation of the global process which approximates the actual process. Let us note that the piece-wise decomposition of the entire process shall yet reflect its self-similar nature. The obtained Tspec in each interval seems to be indeed correlated, as will be illustrated in the simulations. Building on the arguments found in [10], and depending on the value of l_N, one can bound the traffic volume within I_i by either of three

ways. One, for small l_N, i.e., N large, a peak rate bound is sufficient. As l_N gets larger, i.e., N smaller, a peak rate only is too conservative and one needs to refine the volume bound by incorporating a mean bound too which gives the second formulation in terms of peak and mean rates. Three, for l_N on the order of the duration of the global process itself, a mean rate bound may be sufficient.
Using the service curve approach, an arrival function x is said to be bounded by an arrival curve α for all t if and only if for all $s < t$, $x(t) - x(s) < \alpha(t - s)$. Graphically, Figure 3 depicts the arrival curve corresponding to the Tspec given in terms of p, M, r, b. $\alpha(t)$ equals in this case $\min(pt + M, rt + b)$. Thus, in each interval I_i of length l_N, α_i^N is an arrival curve which takes one of those forms:

$$\alpha_i^N(t) = \begin{cases} r_i^N t & \text{for } t \in I_i \text{ , } l_N = L/N \text{ and } N \text{ small} \\ min(p_i^N t, r_i^N t) & \text{for } t \in I_i \text{ , } l_N = L/N \text{ and } N \text{ medium} \\ p_i^N t & \text{for } t \in I_i \text{ , } l_N = L/N \text{ and } N \text{ large} \end{cases}$$

The very values of ¨large¨, ¨medium¨ and ¨small¨ depend on the arrival process itself and the desired accuracy of our bounding approximation. For a highly self-similar process, given by a high value for the Hurst parameter, the intervals I_i should have small lengths l_N. In this case, the peak rate bound applies better as an approximation. If on the contrary, the process is not highly correlated, given by a small Hurst parameter, a mean rate bound offers a good approximation. For the other cases, a peak plus mean formulation applies.

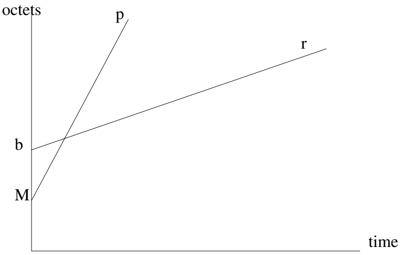


Fig. 3. Arrival curve - Tspec

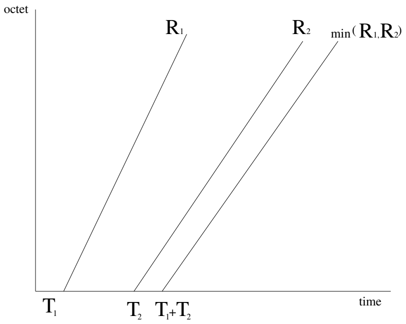


Fig. 4. Concatenation

2.2 Network Modeling and Service Curves

Using the service curve approach, a service curve is defined as the minimal service offered by a single server to our arrival curve. Analytically, for an input function x and output function y, β is a service curve if and only if for all $t \geq 0$, there exists some $t_0 \leq t$ such that $y(t) - x(t_0) \leq \beta(t - t_0)$. The IETF service curve has the form $\beta(t) = R(t - T_0)$ where R is the service rate and T_0 is the time at which the server starts serving our arriving traffic. A nonzero T_0 reflects the presence of a background traffic being served prior to our arriving traffic.

A network is not more than a cascade of successive network elements or servers each offering a service curve β_i. To ease the modeling and analysis of the network, the latter, i.e. the successive nodes that the traffic shall traverse, may be replaced by a single server reproducing their individual services as a concatenation of the individual servers. For n network elements in tandem, each one with service rate R_i and starting service at time T_i, $i = 1, .., n$, one possible concatenation scenario is a network element with a service curve $c(t) = R_n(t - T_n)^+$ where $R_n = \min(R_1, R_2, .., R_n)$ and $T_n = \sum_i^n T_i$, as shown in Figure 4.

3 Performance without Shaping

Let us recall that, for each interval I_i of length l_N, the input function is bounded by an arrival curve of the form $\alpha_i(t) = \min(p_i t + M_i, r_i t + b_i)$. The service curve at the network level is given as $c(t) = R_n(t - T_n)^+$, with service rate R_n and starting service at T_n, as the minimal service curve guaranteed by the server. Taking $c(t)$ as our actual service curve, the output curve is: $\alpha^*(t) = \min(R_n(t - T_n), rt + b)$ for $t > T_n$, as shown in Figure 5. According to the fundamental bounds of the service curve theory (maximum delay d_{max} and backlog B_{max}) in general, and Network Calculus [1] in particular, for $\theta = \frac{b-M}{p-r}$,

$$d_{max} = \frac{p - R_n}{R_n}\theta + \frac{M}{R_n} + T_n \text{ at } t = \theta \text{ if } p > R_n > r \tag{1}$$

We now determine the maximum backlog B_{max}. For $T_n < \theta$:

$$B_{max} = (p - R_n)\theta + M + R_n T_n \text{ at } t = \theta \text{ if } p > R_n > r \tag{2}$$

However, in real settings, two cases may arise. One, it may be the case that the maximum network buffer size B_c is smaller than the above mentioned bound B_{max}, in which case, if nothing is done, some traffic may be lost. Moreover, it may also be the case that the above mentioned bound on delay d_{max} is unacceptable for a real-time user who is not prepared to accept a delay, at the network level, larger than a delay constraint d_c. Again, traffic in excess of d_c may be useless to the user and hence lost. Our objective is then to act on the traffic in such a way so as to not exceed the maximum offered buffer size B_c and /or the maximum tolerated delay bound d_c while guaranteeing a loss free performance. This is achieved by the use of a shaper. A shaper shall be introduced between the source and the network (see Figure 6). It has a size B_{sh} which we try to keep as minimal as possible. It has a shaping rate R_{sh} at which the traffic is shaped and sent into the network. A larger value of R_{sh} means a less affected traffic. This is a good feature as the traffic should be minimally altered. An optimal shaper is thus a shaper with minimal buffer size B_{sh} and maximal shaping rate R_{sh}.

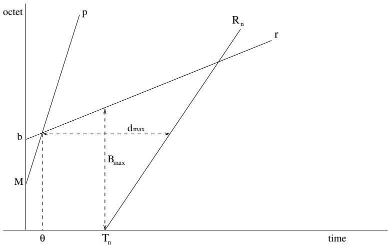


Fig. 5. Arrival Tspec and service curve

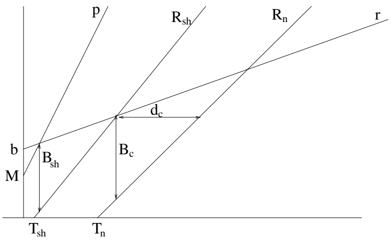


Fig. 6. Shaper between LAN and WAN

4 Shaping

4.1 Regions of Shaping

Adding a shaper to the arriving traffic prior to its entrance to the network is done as follows. A new service curve, corresponding to the actions of the shaper, with parameters $(R_{sh}$,$T_{sh})$ is set between the arrival curve and the network service curve. This causes the arrival traffic to be first shaped by the newly introduced shaping service curve, the output of which is then sent to the network and served by the network service curve. In what follows, we assume, without loss of generality, that the buffer and server at the network level are fully dedicated to our incoming traffic. Any background traffic shall not interfere with our incoming traffic and shall thus be not explicitly shown, i.e., $T_n = 0$.

Shaping to meet buffer requirement. Let us suppose that the maximum buffer size, B_c, at the network level is smaller than the maximum backlog bound, B_{max}, caused by the non-shaped arriving traffic. The point of introducing a shaper in this case is to assure that the incoming traffic does not exceed B_c for a loss-free network performance. For $\theta' = \frac{b}{R_{sh}-r}$,$B_c = (R_{sh} - R_n)\theta'$ at $t = \theta'$ if $R_{sh} > R_n > r$.
Schematically, and considering the setting of Figure 6, the idea is to vary the shaping curve through the segment indicating B_c. In this case, the shaded region, given in Figure 7, shows the region of shaping. It is wise to note the extreme in this case. It is the shaping curve with shaping rate $h_b < R_n$. This corresponds to a maximal buffer size $B_{sh} > B_{max}$ for the network without shaping. However, R_{sh} is not maximal. Let us note that for R_{sh} more than h_b, the buffer constraint B_c is outperformed uselessly for an even higher shaping rate. The optimal case is given by the shaping curve with shaping rate R_{sh} starting from $T_{sh} = 0$ and the intersection point with $rt + b$ is (θ', y) . This corresponds to maximal R_{sh} or equivalently minimal shaping.

$$R_{sh} = \frac{B_c r - R_n b}{B_c - b} \tag{3}$$

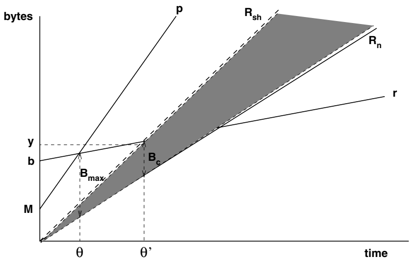


Fig. 7. Region of shaping to meet buffer requirement B_c

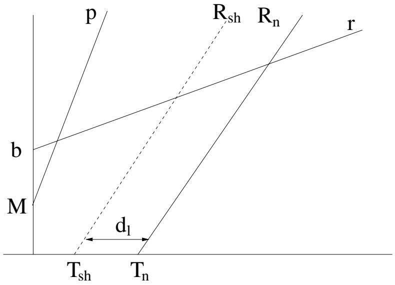


Fig. 8. Region of shaping to meet delay requirement d_c

Again, smaller values of R_{sh} will yield an even smaller B_c uselessly at the cost of higher shaping. Those two cases correspond thus to two feasible shaping parameters depending on the cost of the resources. The first case operates at network buffer less than the target B_c but a high shaping action whereas the second is optimal in view of the shaping action, i.e., large R_{sh}, and network buffer size constraint B_c met.

Shaping to meet delay constraint. In this case, the point of shaping is to reduce the maximum delay to be experienced at the network region from the original d_{max} to a new delay constraint d_c. Let us note that introducing a shaper does not add to the end-to-end delay. The latter shall be just partitioned between the shaper and the network element. This type of partitioning may be useful in an optical context where it is better to hold the packets at the electronic side and not at the optical side where the signal is more prone to being distorted and attenuated. For $\theta' = \frac{b}{R_{sh}-r}$,$d_c = \frac{R_{sh}-R_n}{R_n}\theta'$ at $t = \theta'$ if $R_{sh} > R_n > r$.
This is again achieved by setting appropriate values to R_{sh}. Schematically, the idea is to vary the line the shaping curve through the segment indicating d_c as shown in Figure 8.
The extreme in this case occurs when the shaping curve with shaping rate $h_d < R_n$. This corresponds to a maximal buffer size $B_{sh} > B_{max}$ for the network without shaping. However, R_{sh} is not maximal. Let us note that for R_{sh} more than h_d, the delay constraint d_c is outperformed uselessly for an even higher shaping rate. The optimal case is given by the shaping curve with shaping rate R_{sh} starting from $T_{sh} = 0$ and the intersection point with $rt + b$ is (θ', y).

$$R_{sh} = \frac{R_n(b - rd_c)}{b - R_n d_c} \tag{4}$$

This corresponds to maximal R_{sh} or equivalently minimal shaping. Again, smaller values of R_{sh} will yield an even smaller d_c uselessly at the cost of higher shaping.

4.2 Equation-Based Dynamic Shaping Algorithm

So far, we considered shaping within every interval I_i of length l_N. Our ultimate aim is however to shape the global incoming traffic. Parallel to the idea of partitioning the arrival process so as to locally bound each interval, the shaping scheme introduced in the previous section shall apply to each interval.

It is clear that the shaping rate R_{sh} depends on the arrival curve parameters throughout the whole process. The task in this case is to find optimal, i.e. maximal, shaping rate R_{sh} for each interval I_i such that the buffer constraint and/or delay constraint are satisfied. This is achieved by dynamically changing the shaping rate from one interval to the next. The dynamic shaping algorithm is then as follows.

1. Set observation window size equal to l_N
2. Determine corresponding Tspec in interval $(I_i)_{i=1,\ldots,N}$
3. Apply Equations 3 and 4. to set shaping parameters such that
   i. shaping is minimal in the sense of minimal buffer size B_{sh} and maximal shaping rate R_{sh}
   ii. requirements are met, i.e., buffer or delay constraint at network level
   iii. no loss at shaper, i.e. B_{sh} not exceeded.

5 Numerical Results

5.1 Model

We consider the end-to-end system shown in Figure 1. Let the self similar traffic resulting from the LAN sources have the following characteristics: mean = 100 Mbit/s, variance = 10^8, and Hurst parameter H = 0.7.
Let the packets be of maximal size M equal to 1540 bytes. At the network level, let R_n = 227 Mbit/s be the rate of the server, with buffer capacity B_n equal to 100 packets. We assume without loss of generality that a fixed portion of the server at the network level, with service rate R_n and buffer space B_n, is entirely dedicated to our incoming traffic; any background traffic will not be modeled explicitly. This assumption simplifies the analysis and simulation as T_n is equal to zero. In real setting, this amounts to considering a dedicated share of buffer space and service rate.

5.2 Estimation of Arrival Parameters

The first step of our equation-based dynamic shaping algorithm is to estimate the parameters of the incoming traffic into a Tspec format, i.e., peak rate p, mean rate r and maximum burst size b for different observation windows of size l_N. In interval $(I_i)_{i=1,\ldots,N}$ of length l_N, as stated in Section 2.1, the peak rate p is equal to the reciprocal of the minimum interarrival time X_{min} and the mean rate r is equal to the reciprocal of the average interarrival times X_{ave}. The complexity lies in the estimation of the maximum burst size b, an essential parameter for a well-defined arrival envelope, performance bounds and shaping issues.

By definition, b corresponds to consecutive arrivals with interarrival times tending to zero as the traffic is observed in a mean phase. In our present work, we observe the consecutive interarrivals of size equal to X_{min} for every interval $(I_i)_{i=1,...,N}$ of length l_N and store the largest value of the corresponding packets in b.

Estimation of mean rate r and peak rate p. Figures 9 and 10 shows the average rate r_i for intervals I_i of lengths equal to 300ms and 1s respectively. We notice that for a given window size l_N, r_i^N varies from one interval to the next keeping the same behavior as the original traffic, i.e., incorporating correlation. On the other hand, the family of $(r_i^N)_{i=1,...,N}$ behaves in the same way in different lengths l_N of intervals $(I_i)_{i=1,...,N}$, i.e., in many time scales. That means the presence of self-similarity property in the sequence $(r_i^N)_{i=1,...,N}$. Figures 9 and 10 shows two times scales $(r_i^N)_{i=1,...,N}$ behaviors : 300ms and 1s. The same remarks remain valid for p.

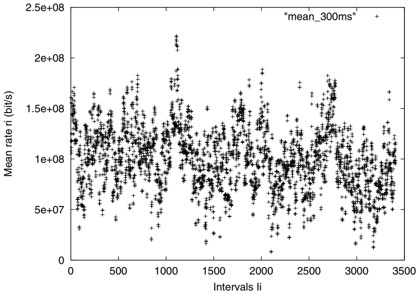


Fig. 9. Average rate r_i during the Interval I_i : $l_N = 300ms$

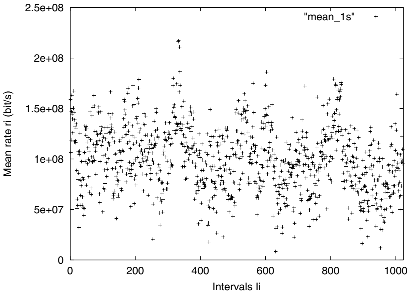


Fig. 10. Average rate r_i during the Interval I_i : $l_N = 1s$

Estimation of burst size b. For each window size l_N, we have observed the interarrival packets during the smallest mean rate r_i. The sum of consecutive interarrivals smaller or equal to the interarrival time within the corresponding peak rate p_i corresponds to the burst size. The obtained values for b vary from 85 packets for small window size l_N to 60 for larger ones.

5.3 Non-constrained Performance

If no shaping is used at the access of the network, Figure 11 shows the probability density function of the queue at the network level. For a no-loss performance, this figure indicates to us that a buffer size of 55 packets is needed at the network server. The maximum delay in the network level in this case is equal to 0.00268 sec.

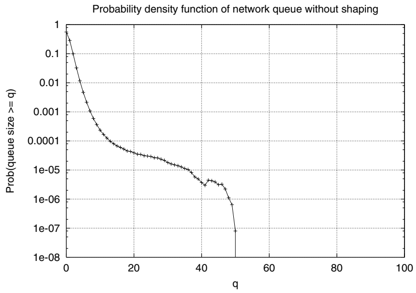


Fig. 11. Network queue PDF without shaping

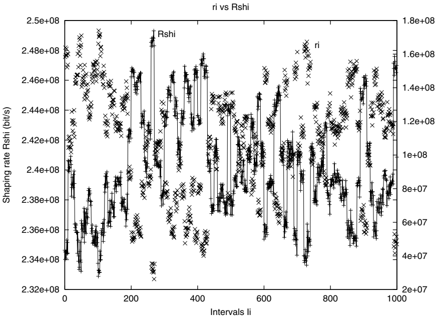


Fig. 12. Average rate r_i vs Shaping rate R_{sh} during I_i

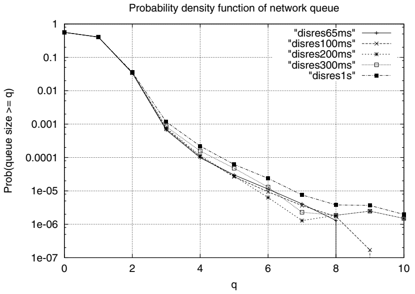
5.4 Buffer-Constrained Performance

In this case, let us assume that in fact, the buffer size B_c at the network level cannot be as large as to hold 55 packets, i.e., if no shaping is used, there will be some loss. Let B_c be large enough to hold 10 packets , a typical buffer size in optical switches. To keep up with a non-loss performance, we need to operate some shaping at the access of the network in order to meet the buffer constraint $B_c = 10$. This brings us to the second step of our algorithm. Based on the arrival Tspec and the service curve equations, we derive the shaping parameters for every interval $(I_i)_{i=1,...,N}$ of length l_N. For intervals of length $l_N = 100$ ms, Figure 12 shows the mean arrival rate r_i versus shaping rate R_{sh} throughout the duration of the connection (100 seconds).

We notice that r_i and R_{sh} are inversely proportional; for every large values of r_i, i.e., high input rate, the value of R_{sh} is small, i.e., a severe shaping is needed to accommodate the buffer constraint and the loss-free performance. The inverse case is also true. The third step of the algorithm is to plug the equation-based shaping parameters back into the simulation model. Figures 13 and 14 show the probability density function of the buffer occupancy at the network level and shaper, respectively, for different observation window lengths l_N. The shaper size is also derived from the equations and the largest value over all intervals is used. This conservatism explains the fact that no loss is observed at the shaper. The independence between the shaping queue PDF's and the interval sizes l_Ns, can be explained by the fact that the shaping rate R_{sh} is adaptive with respect to the incoming traffic in order to meet the non-loss performance. Thus, for each l_N, R_{sh} varies in an inversely proportional way with the mean rate r_i, keeping the shaping queue behavior more or less the same.

The above figures and observation may actually suggest that self-similar traffic, variable at different time scales, may exhibit the same type of variability at those very time scales. If this turns out to be true, it may suggest that observing and monitoring the traffic at small time intervals may be sufficient in constructing and extrapolating its behaviour over larger time scales.

As of the network, we notice that the smallest interval lengths $l_N =$ 65 and 100 ms yield a non-loss performance. This is explained by the fact



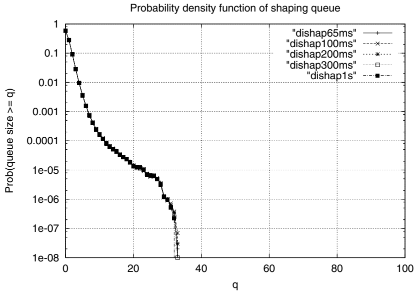


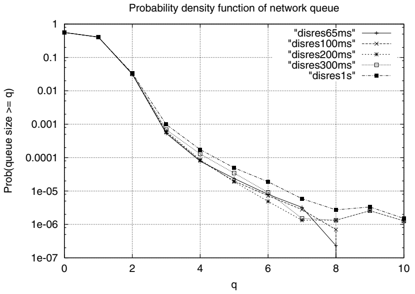
Fig. 13. Network queue PDFs for different lengths Intervals I_i: 65ms, 100ms, 200ms,300ms, 1s

Fig. 14. Shaping buffer PDFs for different lengths Intervals I_i: 65ms, 100ms, 200ms,300ms, 1s

that at those interval lengths, we obtain higher precision for estimation of arrival parameters and hence shaping parameters. For larger interval lengths, $l_N = 200$ and 300 and 1000 ms, some loss, on the order of 2.4 10^{-7}, is observed. This is explained by the fact that for small precision, the arrival parameters are under-estimated. Put in the equations, they yield high shaping rates, or equivalently, soft shaping. This in turn results in loss at the network level.

5.5 Delay-Constrained Performance

Let us assume that in fact, the tolerated maximum delay d_c at the network level cannot be as large as 0.0005 sec , i.e., if no shaping is used, there will be some loss due to the delay being exceeded. Again, to ensure this performance, we need to operate some shaping at the access of the network in order to meet the delay constraint d_c =0.0005 sec. We apply then the three steps of our equation-based algorithm , as done for the buffer-constrained case.



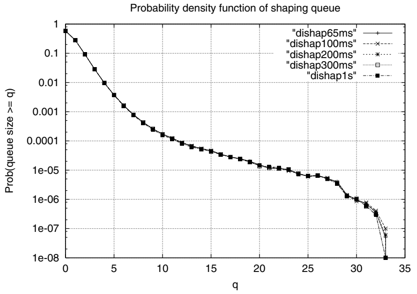


Fig. 15. Network queue PDFs for different lengths Intervals I_i: 65ms, 100ms, 200ms,300ms, 1s

Fig. 16. Shaping buffer PDFs for different lengths Intervals I_i: 65ms, 100ms, 200ms,300ms, 1s

Figures 15 and 16 show the probability density function of the size of the buffer at the network level and shaper, respectively, for different observation window lengths l_N. Table 1 illustrates the maximum delay values obtained by simulation for different window lengths l_N: 65ms, 100ms, 200ms, 300ms and 1s. Again, we notice that the smallest interval lengths $l_N = 65$ and 100 ms yield the target maximum delay.

Table 1. Maximum delay at the network level for different window lengths l_N: 65ms, 100ms, 200ms,300ms, 1s

window lengths l_N	**65ms**	**100ms**	**200ms**	**300ms**	**1s**
maximum delay in the network	0.0004102 s	0.0004219 s	0.0005413 s	0.0005414 s	0.0005414 s

For larger interval lengths, $l_N = 200$ and 300 and 1000 ms, the maximum values of the observed delay exceed the constraint. This is explained by the fact that for small precision, the arrival parameters are under-estimated.

6 Conclusion

In this paper, we focused on self-similar traffic at the input of an optical node. If this traffic is left as is, it cannot satisfy the buffer and/or delay constraint at the network level, which may be very stringent in the optical case. In order to meet those requirements, shaping is essential. In this work, we proposed an equation-based dynamic shaping with three key steps: 1) estimation and fitting of interval-wise incoming traffic into arrival curves, 2) solving into the service curve approach for the shaping parameters in an adaptive manner and 3) fitting the later back into the original model.

As of the first step of our algorithm, we notice that the input estimate reproduces the same self-similar, correlated nature of the original traffic. The shaping parameters derived in step 2 are typically conservative owing to the deterministic nature of the service curve. However, when put back into the original model, i.e., step 3, they are shown to be numerically not very conservative. This may be explained by the correlated nature of the original self-similar traffic.

Future work perspectives shall focus on the following issues. First, the conservatism of the deterministic version of the service curve approach seem to be less apparent in the presence of self-similar, LRD traffic, as shown by the small loss at the network level. It may be wise to quantify to which extent self-similar traffic reduces this conservatism. Second, optimal shaping relies on the trade-off between buffer sizes at the shaper versus network. We intend to tackle this issue by releasing the loss-free determinism at the shaper level where we can in effect tolerate some loss. This can be achieved by more severe shaping action, by decreasing the shaping rate R_{sh}, and hence reaching the buffer size limit. This limit can actually be violated in controlled manner in order to tolerate a loss performance similar to that encountered at subsequent network elements. This feature is desirable and more pragmatic as it is useless to operate a shaping performance too perfect with respect to that of the optical network; after all, what really counts to the user view is the end-to-end performance.

References

1. J-Y. Le Boudec *Application of Network Calculus To Guaranteed Service Networks.* IEEE Trans on Information theory, (44) 3, May 1998.
2. R. L. Cruz *A calculus on network delay, part I: Network elements in isolation.* IEEE Transaction of Information Theory, 37(1):114-121,1991.
3. R. L. Cruz *Quality of Service Guarantees in Virtual Circuit Switched Networks.* IEEE JSAC, 1995.
4. A. Erramili, O. Narayan, and W. Willinger. *Experimental Queuing Analysis with Long-Range Dependence Packet Traffic* IEEE/ACM Transactions on Networking, 4(2), pp. 209-223, Apr. 1996.
5. H. Flower and W. Leland. *Local Area Network Traffic Characteristic, with Implications for Broadband Network Congestion Management* IEEE JSAC, 9(7), pp. 1139-1149, September, 1991.
6. V. Frost and B. Melamed. *Traffic Modeling for Telecommunications Networks* IEEE Communications Magazine, 32(3), pp. 70-80, March, 1994.
7. D. Ferrari and D. Verma. *AA scheme for real-time channel establishement in wide-area networks* IEEE Journal on Selected Areas in Communictions, 8(3):368-379 April, 1990.
8. S. Golestani *Congestion-free transmission of real-time traffic in packet networks* In Proceedings of IEEE INFOCOM'90, pp. 527-142,San Francisco, California, June 1990.
9. C. Kamanek, H. Kanakia and S. Keshav. *Rate controlled servers for very high-speed networks* In Proceedings of IEEE Global Telecommunications Conference, pp. 300.3.1-300.3.9,San Diego, California, December 1990.
10. E. Knightly and H. Zhang. *Traffic characterization and switch utilization usinf deterministic bounding interval dependent traffic models* In Proceedings of IEEE INFOCOM'95,Boston, MA, April 1995.
11. 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(1), pp. 1-15, February 1994.
12. V. Paxon and S. Floyd. *Wide-Area Traffic: The failure of Poisson Modeling* IEEE/ACM Transactions on Networking, 3(3), pp; 226-244, June 1995.