4

Statistical Sharing and Traffic Shaping: Any Contradiction?

Yee-Hsiang Chang

Hewlett-Packard Laboratories
1501 Page Mill Road, Palo Alto, CA 94304-1126, USA

Statistical sharing to achieve a high network utilization is a major motivation behind the packet-switched network. This idea takes advantage of the bursty nature of traffic sources to achieve a better network utilization. In recent years, various control mechanisms have been proposed for future high-speed networks to support the network traffic management. Some of the control schemes advocate traffic shaping to reduce the burstiness of the traffic, and others insist on maintaining the bursty nature for better statistical sharing. This paper looks into the true meaning of statistical sharing, and tries to shed light on the design of better control mechanisms by using a simple queueing model to show the relationship between the congestion and the network utilization.

1. MEANING OF STATISTICAL SHARING

The theoretical foundation for statistical sharing is based on the law of large numbers [1-2], which is described by Kleinrock as, *"the collective demand of a large population of random users is very well approximated by the sum of the average demands required by that population"*. That is, the stable state in the network utilization is achieved when the number of users is large, in which case each individual traffic balance its burstiness traffic with others. This stable state is the key to provide good sharing, which is known as *statistical sharing*. On the other hand, if the population is small and each traffic is bursty, an unstable condition is produced and results in bad sharing. One example is in Figure 1, the multiplexing of traffic A and B reduces the overall variance. If there are more traffic multiplexing together, the variance goes down further. However, when two bursts from traffic A and B arrive at the same time to a point that the network can not sustain, delays and losses are likely to happen. This produces a network congestion, in which case the network is overloaded and loses its ability to provide user-requested services.

G. Neufield et al. (eds.), *Protocols for High Speed Networks IV*

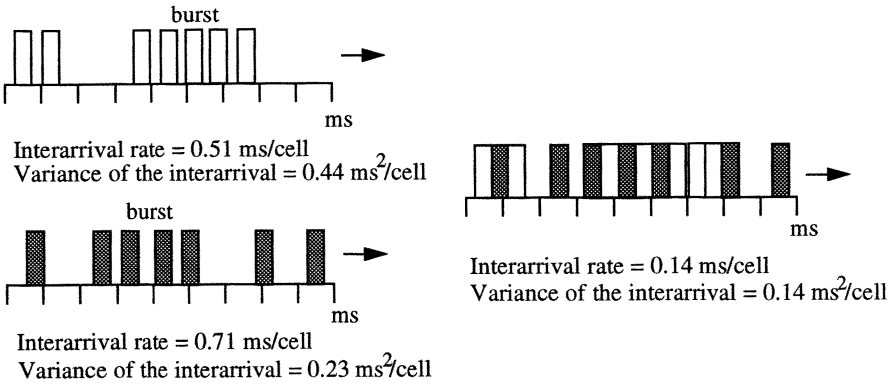


Figure 1. Multiplexing of Traffic Reduces the Overall Variance.

The goal of the network design is to have good sharing (high network utilization) and also maintain guarantees to services. Good sharing can be achieved also by a deterministic way. The deterministic traffic has the potential to obtain even better sharing than the bursty one, since the deterministic traffic is stable in nature. For example, good sharing can happen in the traditional telephone networks with constant-bit-rate circuits (Figure 2)†. When all the channels are fully occupied and multiplexed together, the network utilization is 100%.

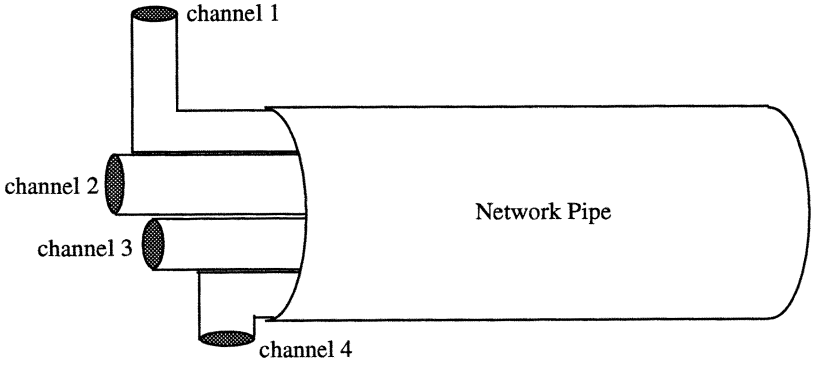


Figure 2. Good Sharing Under the Constant-Bit-Rate Network as Each Channel Fully Utilizes its Capacity.

So, why the trend today moves toward statistical sharing instead of maintaining the traditional circuits? This is because many real-life applications tend to be bursty, which waste bandwidth with the traditional circuit-switched networks. Can we do traffic shaping within the

† Note that Figure 2 uses a conceptual way to show the sharing among different channels; the actual multiplexing (e.g., time division multiplexing) is different.

tolerance of each application to achieve better sharing? The answer is yes because the traffic is more stable especially when fewer bursty users share the same link. We argue that the final solution for the sharing is a combination of both statistical (due to application's nature) and deterministic ways (due to traffic shaping).

In general, statistical multiplexing reduces the long-term average randomness, but increases the potential to have a severe short-term randomness. As the example in Figure 1, although multiplexing reduces the overall variance, the bursts from both streams generate a bigger burst, which potentially causes buffer overruns and generates delays in the down-stream nodes. Traffic shaping helps to smooth out the short-term randomness.

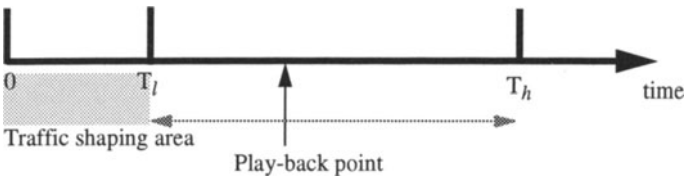
One misconception among some literatures states that maintain the burstiness of the traffic is the key for statistical sharing.[†] This is obviously not true. The key for statistical sharing to work is to reach the stable state with large population. Maintain the burstiness of the traffic does not generate a stable condition. On the other hand, more deterministic the traffic is stabilizes the overall traffic even with a small population, and achieve better sharing.

2. TRAFFIC CHARACTERISTICS AND TO WHAT DEGREE WE CAN CHANGE IT

There is a limit for traffic shaping due to applications' traffic characteristics. What is this limit? To answer, we need to classify communication requirements for different applications. There are three different types of real-time communications. The first one is called *hard real-time or hard guaranteed.* For this type of the application, a maximum time limit is set and all the communications are required to finish within this limit. One example is the communications for real-time control signals on embedded systems (such as the space shuttle). The second one is called *soft real-time or statistical guaranteed.* For this type of the application, the time limit is the same as the first case but can be achieved statistically (not 100% guarantee). One example of this type is the communication for meta computing. Meta computing requires fast communications to work on a wider area. However, the statistical fluctuation of the message transmission is not fatal to the application. The third type is *play-back applications* defined by Clark ed. al. [5]. This type has even less real-time requirements from the network than the two previous cases - it relies on the end systems to adjust. The best examples of play-back applications are voice and video communications. At the transmission source, the voice or video signal is first packetized, and then transmitted over the network. The receiver buffers the incoming messages to remove jitter, and play the voice or video back at the designated play-back points. The receiver can adjust the play-back point within a range according to the network condition. The play-back applications will be the vast majority in the future [5]. In this paper, we mainly look at the traffic shaping issue on this type of the real-time communications.

The play-back point is adjusted between two limits. At one end is the time that the application has a zero performance gain if the message arrives earlier than this point. For example, the audio communication requires the round-trip delay within 400 ms [6]. If the network provides a lesser time, there is no performance gain to users. This time limit is marked as T_l in Figure 3. At the other end is the time that the application is intolerable to the delay if

the message arrives later than this limit (see Figure 3). For example, if an audio delay is more than 5 seconds†, the interactive communication is completely unacceptable. We use T_h in Figure 3 to represent this time.



T_l The time limit that the application does not achieve any better performance if the time used is less than this number.
T_h The time limit that the performance does not be tolerable if the time used is more than this number.

Figure 3. Timing Requirements and the Adjustable Area for a Play-Back Application.

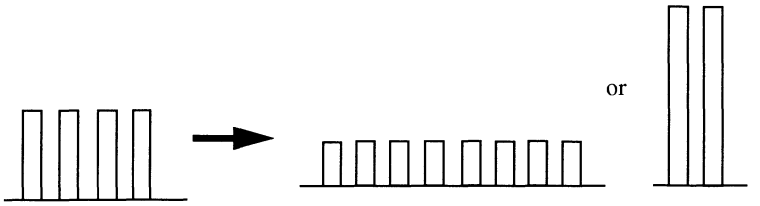
Between T_l and T_h is the range that the play-back point can adjust. If the network is temporarily congested, the play-back point moves toward T_h to recover late messages. When the network is less congested, the play-back point moves back to T_l to achieve a faster response. Traffic shaping should introduce no further delay than its T_l for a message. In general, traffic shaping adjusts the delay for each message (within its delay budget) in order to achieve less delay and better sharing for the overall traffic.

3. THE STYLES OF TRAFFIC SHAPING AND THE PERFORMANCE MODEL

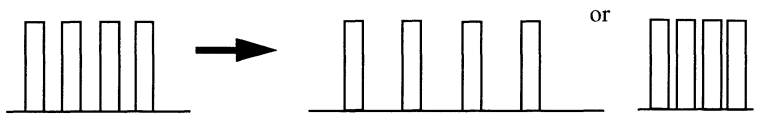
Traffic shaping can be exercised in two places for a packet network. One place is to reduce or increase the peak packet size, and the other one is to un-smooth or smooth the packet/cell inter-arrival time (Figure 4). For ATM with fixed-size cells, traffic shaping is for the latter case (Figure 4b).

---

† More specifically, the paper from [3] makes the comments about Leaky-Bucket from [4] as "A simple model, described in [4], works in the following way: each switch at the network entrance puts packets from each data flow into a corresponding bucket which has a fixed size. The bucket opens periodically to emit packet for transmission. When the bucket is full, incoming packets are discarded. ... The first version of Leaky-Bucket reduces statistical multiplexing because packets are transmitted at a constant rate rather than whenever the channel is available. ... The VirtualClock algorithm avoids those drawbacks by merely ordering packet service without reducing statistical sharing." The point is that a constant bit rate (CBR) channel, which is the ultimate shaped traffic, does not hurt sharing or statistical sharing with other channels, but help sharing instead. The point from [3] about the message should be sent "whenever the channel is available" is tough to achieve because the local resource availability does not guarantee the remote network resource availability. Also note that Leaky-Bucket from [4] does not generate a CBR channel, but a shaped traffic.



(a) Traffic shaping: height adjustment



(b) Traffic shaping: interarrival time adjustment

Figure 4. Different Ways of Traffic Shaping

If we look at the interarrival time of packets, it is a renewal process, which is best modeled by a general arrival. Following, a simple queueing model is employed to prove the significance of traffic shaping, and demonstrate a way to maintain a high network utilization while eliminating the congestion at the same time.

3.1 The performance model (G/D/1)

Here, a simple model is used to show the performance of a network with traffic shaping. Figure 5 presents a network configuration with several switches and three connections across the network. These connections go through one of the links together, where is the potential bottleneck. If the traffic control and management of the network allow more connections across this very link at the same time, the network achieves a higher utilization. This link is modeled by a simple queue (Figure 6). Its characteristics represents the whole network under congestion. The link can actually be any link in the network under a heavy load.

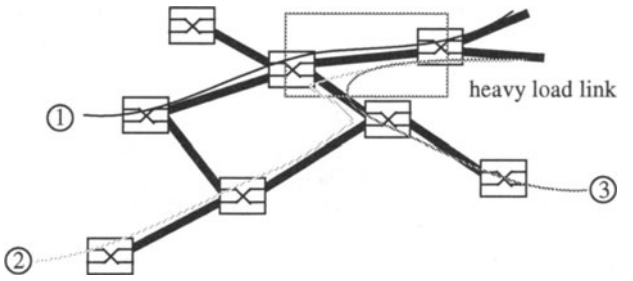
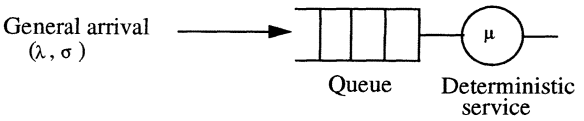


Figure 5. Network Model that Uses a Heavily Load Link to Represent the Whole Network.

† There is no empirical data for this. 5 seconds is just an example here.

The network is modeled with a deterministic service rate and a general arrival rate (G/D/1) (Figure 6). The deterministic service rate is assumed because the future switches will be fast and take a constant time to process each fixed-size cell. The arrival process consists of the traffic from various incoming sources and can be occasionally faster than the output link speed, which results in a queueing effect. We also assume no peak rate allocation. If the network employs the peak rate allocation, the total aggregated speed is lower than the link speed at any instance, and no queue is formed. The simple queueing model is used to derive a generalized close-form solution that can demonstrate the basic relationship of various parameters. Note that this model is a steady-state solution. More complicated transient analysis is left for further study.



λ Average arrival rate.
σ Standard deviation of the arrival process.
μ Average service rate.

Figure 6. Network Model with General Arrival and Deterministic Service (G/D/1).

Other assumptions are used. First, the effect of admission control is neglected. We assume a perfect admission control to calculate the performance upperbound. Second, the policing function is assumed enforced at the boundary of the network, which no unexpected data can flow in. Third, the scheduling discipline is FIFO and no priority. FIFO is employed because it is the simplest scheduling discipline to implement and exists many ATM switches today. Also, FIFO provides the best sharing among the scheduling disciplines [5]. No priority is used because the goal is to see the performance under only one class. In other words, we are looking at the performance of traffic at the highest priority.

The network utilization is obtained by observing the number of packets or cells in the queue. If there is always one packet or cell being serviced by the network all the time, the network utilization is 100%, which means a constant flow of packets or cells moving at the link speed. So, the

Network utilization = 1 - P_0.

P_0 is the probability of zero packet or cell in the network service point. P_0 can be obtained by the following derivation because of the steady state condition in the queueing network.

E(number of arrivals in time T) = E(number of departure in time T)

$E(x)$ is the expected value of random variable x.
T is a long period of time.

Since T is a long time, the number of the arrival packets/cells (λT) during this period should be equal to the number of departure packets/cells ($\mu T(1\text{-}P_0)$ to maintain the steady state. λ and μ represent the average arrival and service rate for the queueing system.

$$\lambda T = \mu T(1 - P_0)$$

$$P_0 = 1 - \frac{\lambda}{\mu} = 1 - \rho$$

Network utilization = $1 - P_0 = \rho$ (1)

Note that the network utilization is determined only by the average arrival and service rate. By solving the G/D/1 queue, we get the average queue length ($E(Q)$) [7-8], which is

$$E(Q) = \sigma^2\lambda^2/2(1-\rho) \quad (2)$$

The waiting time (W) in queue is

$$W = E(Q)/\lambda = \sigma^2\lambda/2(1-\rho) \quad (3)$$

The queue length is an indication of the network congestion, which normally means the server lags behind for packet or cell processing and results in a queueing delay. The waiting time in the queue contributes to part of the total delay (this time plus the propagation delay and the processing delay constitute the overall delay for a message.) The propagation delay is fixed once the distance is determined. The part that can be controlled is the queueing delay. Today, this number is not trivial. The propagation delay across the US continent (3000 miles) via Internet is about 22 ms. The measured minimum one-way delay is around 50 ms (using *ping* program), and the average delay is around 80 ms.† The queueing delay represents the difference between the average delay and the minimum delay, which constitutes 60% over the minimum delay. Furthermore, 27% of the packets are lost due to the congestion.

We can see in (2) and (3) there are three parameters that affect the network queue length and waiting time. They are σ, λ, and μ. σ is the standard deviation of the arrival process. The value of σ indicates the burstiness of the arrival process. If the value is big, the inter-arrival time varies very much. If the value is small, most of the interarrival time is similar. It is clear that reducing the burstiness of the traffic reduces the queue length and avoid congestion. Reducing the λ and increasing the μ also reduce the queue length. However, the latter case also reduces the network utilization.

So, to keep a good network utilization, the design should rather reduce the variance (or bursty) of the traffic than reducing the arrival rate or increasing the service rate in a long-term sense. This suggests that traffic shaping is desirable.

---

† This number is measured between California and North Carolina at 1pm Pacific Time Zone June 13, 1994.

4. THE TRAFFIC MANAGEMENT DESIGN AND THE PERFORMANCE UPPERBOUND

With the basic model in mind, let us see the fundamental principals for the network control mechanism. Here, we only consider the sharing among all real-time traffic (no priority used). One way to achieve a good network utilization is to multiplex non-real-time traffic with the real-time traffic to fill the unused network capacity. This is a good idea if there always exists enough non-real-time traffic. By only dealing with the real-time traffic to achieve a high network utilization, the design also does well with or without multiplexing non-real-time traffic.

The design goal of the control mechanism: *To achieve a high network utilization and congestion-free at the same time.*

The congestion is directly related to the queue length. The control mechanism should be designed to reach the balance point between the highest possible network utilization and low network congestion. This balance point is not fixed and has very much to do with the traffic pattern of the sources. There is a dynamics among the following parameters. As mentioned, we assume to have the FIFO as the scheduling mechanism and a perfect admission control to simplify the model.

- network utilization,
- scheduling,
- congestion (queueing delay, queue length),
- traffic pattern (λ, μ, and σ),
- admission control, and
- control mechanisms (control burstiness, or rate).

Since the parameters λ and μ contradict the requirements of a shorter queue length and a higher network utilization, it does not make sense to control them at the first place. We argue that the control mechanism should be in two levels. At the first level, the network should exercise traffic shaping all the time to reduce the possible burstiness. This is a traffic management function, which can result in a shorter queue length/delay time and a higher network utilization in the long term. If the first level traffic management can not eliminate the congestion condition and the congestion becomes a more serious situation, the second level control mechanism should be employed. At the second level, by sacrificing the network utilization, the network or the end systems should reduce λ or increase μ to reduce the queue length. This is generally a flow control function [8-9].

The G/D/1 example is a simple model to show the relationship among several major parameters. A more complicated model is desirable to demonstrate more detail information about how to reach this balance point. Following, a performance upperbound analysis is done using the simple G/D/1 model. A simple shaping scheme is introduced for the purpose of demonstrating the shaping effect and finding the performance upperbound, and not for the purpose of proposing such a scheme. We also employ the JPEG data as a generic video traffic for the same purpose.

4.1 Upperbound analysis

If the nature of a real-time class limits the maximum waiting time (for example, the voice can not tolerate too much delay, T_h), and the first level control mechanism (traffic shaping) can only function up to T_l, we can obtain a performance upperbound for the network utilization, which represents the best value that the network utilization can reach.

For example, using the JPEG compressed Star War movie† as a generic video traffic pattern for input. The movie stream has the following statistical values:

Average rate = 5.34 Mbps (13895.46 cell/sec)
Peak rate = 15.06 Mbps

The internal speed is assumed to be 200 Mbps.†† The internal speed in the sender is very likely to be faster than the rate passing through the network, because the former rate depends on the cpu, memory, and the internal bus speed, which is normally a order of the magnitude faster than the network speed. The internal switch speed is also several times faster than the switch port rate to avoid blocking within the switch.

As JPEG frames generated in the source, they are packetized (or segmented) into cells. The interarrival time among cells is based on the internal speed in the sender (Figure 7). In the case of a 200 Mbps internal speed, the variance of the JPEG stream is

Variance = 2.890781 ms^2/cell

This value is also used to represent the switch internal speed in this paper.

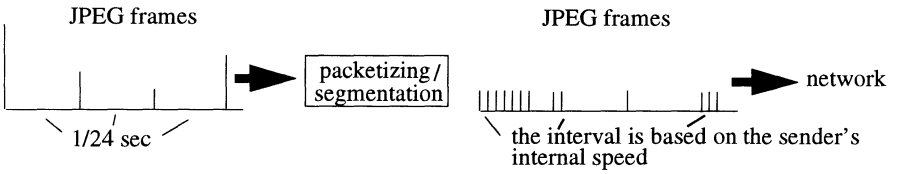


Figure 7. The Internal Rate is Faster than the Delivery Rate

With above numbers, the network utilization upperbound is calculated in two conditions: with and without shaping. For the case without shaping, the performance upperbound for the utilization against the delay (using the equation (3)) is shown in Figure 8. If an application has a 200ms delay budget (e.g., voice communications), and the propagation delay is assumed insignificant in this case, the maximum network utilization is around 0.86 after subtracting the reassembly delay (one interval - 41.67ms†††) at the receiver. This one interval reassembly delay is unavoidable. However, depending on the variation of the traffic (σ), this reassembly delay might need to increase to recover the late cells or just drop the frame.

† The JPEG compressed starwar movie data is from Bellcore.

†† The internal speed is the data rate inside the source and the data rate inside the switch.

††† In our case with 200 Mbps internal speed, the cells from the same frame remain inside the 1/24 sec range after packetizing.

For the case with shaping, there are two solid curves in Figure 8 that present the result. In both cases, the traffic shaper has the knowledge that a frame arrives in every 1/24 sec (41.67ms), and uses 41.67ms or 83.34ms as a window to shape the traffic. For the example of using 41.67ms as the shaping window, the traffic shaper buffers the data every 41.67ms, then sends the cells out in a constant rate in the next frame interval (Figure 9). Note that the numerical result shows that the traffic shaper does not need to align on the frame boundary (Table 1) if the traffic is random enough†, which makes the shaping function easier to implement.

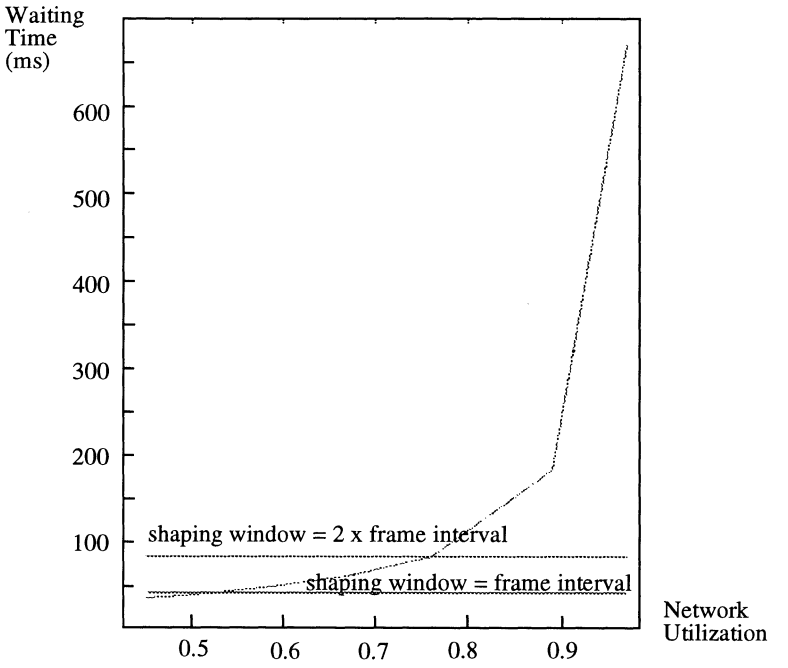


Figure 8. Queueing Delay vs. Network Utilization with and without Shaping.

---

† If the shaping window size is fixed, the average and the variance of the interarrival time are the following.:

$$average = \sum P_i x_i = \frac{x_1}{w}\cdot\frac{x_1}{x} + \frac{x_2}{w}\cdot\frac{x_2}{x} + \ldots = \frac{x_1^2 + x_2^2 + x_3^2 + \ldots}{wx}$$

$$variance = \frac{x_1^3 + x_2^3 + x_3^3 + \ldots}{w^2 x}$$

where

w: the frame interval

x: the sum of $x_1, x_2, \ldots$

$x_1, x_2, \ldots$: the cell number within the shaping window

Shifting the window changes the distribution of cells in each interval. However, as long as the distribution is random enough, and the number of the interval is large, the overall values are similar.

The disadvantage of traffic shaping is that a constant delay is generated, 41.67ms or 83.34ms in the examples; however, due to the reduction of the variance, the waiting time reaches a fixed value even under the heavy load.

Table 1
Window Shifting Effects

	Align on frame boundary	Shift 1/4 window size	Shift 1/2 window size	Shift 3/4 window size
Average of cell interarrival (ms)	0.073295	0.073294	0.073294	0.073294
Variance of cell interarrival (ms^2)	0.000288	0.000283	0.000283	0.000283

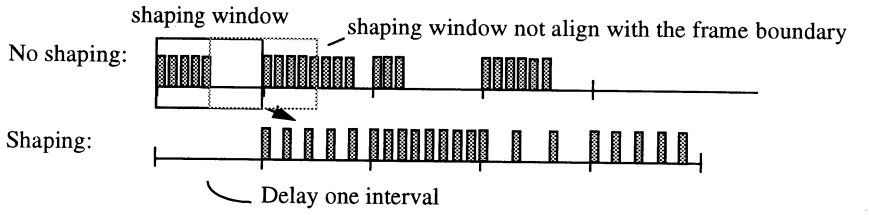


Figure 9. The Traffic Shaping Scheme and the Associated Delay.

The shaping window can be reduced or increased. The biggest case covers all the data and generates a constant-bit-rate stream (CBR). The smallest case results in no shaping at all. When the window is small, the buffering delay is small, but the variance is big. On the other hand, when the window is big, the buffering delay is big, but the variance is smaller.

5. WHERE ARE THE POSSIBLE PLACES TO EXERCISE TRAFFIC SHAPING

The ability to do traffic shaping lies on the knowledge of the application traffic pattern and required guarantees. This fits well with the basic concept of the ATM traffic management that requires a negotiation of the traffic contract. In the current standard, a signaling message carries this information across networks for admission. Having this knowledge throughout networks, the traffic shaping function can be exercised in various places.

The most convenient place for traffic shaping is to associate this function with the network policing mechanism. The policing mechanism makes sure the application sends data according to promised. For each packet, a delay budget can be specified by the application to the policing

mechanism to inform the shaping range.

The traffic shaping function can also be put into the intermediate nodes [10-11]. This idea has better chance to perform more effective traffic shaping due to using all the nodes. The most complicated case is the per-VC traffic shaping in every node. However, the hardware complexity and processing overhead must be taken into consideration.

6. CONCLUSION

In this paper, the meaning of sharing is addressed. We conclude that by using traffic shaping to introduce more deterministic traffic behavior is desirable to stabilize the network condition and achieve good sharing. We also propose that the future network should use a combination of both statistical (due to application's nature) and deterministic sharing (due to traffic shaping). Then, we look at several types of real-time applications, especially the play-back one, to see how much shaping is allowed to do. An analytical model is used to show the relationship between the network congestion and the network utilization. From the model, it is shown that more deterministic the traffic is, less congestion the network is, and the high network utilization is achieved. From this analytical model, we then do the performance upperbound analysis, discuss the design of control mechanisms by proposing a two-level control scheme, and where can we exercise these control mechanisms in high-speed networks.

REFERENCES

1. Kleinrock, L., Queueing Systems Vol II: Computer Application, John Wiley & Sons, New York, 1976.
2. Wolff, R.W., Stochastic Modeling and the Theory of Queues, Prentice-Hall, 1989.
3. Zhang, L., "VirtualClock: A New Traffic Control Algorithm for Packet-Switched Networks," ACM TOCS, 1990.
4. Turner, J., "New Directions in Communications (or Which Way to the Information Age?)," IEEE Communications Magazine, October 1986.
5. Clark, D.D., Shenker, S., and Zhang, L., "Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism," SIGCOMM 1992.
6. Emling, J.W. and Mitchell, D., "The Effects of Time Delay and Echos on Telephone Conversations," Bell System Technical Journal, November 1963.
7. Kleinrock, L., Queueing Systems Vol I, John Wiley & Sons, New York, 1975.
8. Mukherjee, A., Landweber, L.H., and Faber, T., "Dynamic Time Windows and Generalized Virtual Clock: Combined Closed-Loop/Open-Loop Congestion Control," INFOCOM '92, 1992.
9. Makrucki, B.A., "On the Performance of Submitting Excess Traffic on ATM Networks," IEEE GLOBCOM '91, December 1991.
10. Boyer, P.E., Guillemin, F.M., Servel, M.J., and Coudreuse, J.-P., "Spacing Cells Protects and Enhanced Utilization of ATM Network Links," IEEE Network, September 1992.
11. Verma, D., Zhang, H., and Ferrari, D., "Delay Jitter Control for Real-Time Communication in a Packet Switching Network," in Proceedings of TriComm '91, 1991.